



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

Trabajo de Fin de Grado

Desarrollo de una plataforma Fullstack con Chatbot de IA

Development of a Fullstack platform with AI Chatbot

Roxana Mihaela Baba

La Laguna, 23 de mayo de 2024

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

D. **Alejandro Pérez Navas**, profesor Asociado de Universidad adscrito al Departamento de Ingeniería Informática y de Sistemas en el área de 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:

"Desarrollo de una Plataforma Fullstack con Chatbot de IA"

ha sido realizada bajo su dirección por D. **Roxana Mihaela Baba**.

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

Quiero agradecer a mi familia, pareja, amigos y compañeros de universidad por su apoyo incondicional. También a mi tutor, Javier, por ser un valioso mentor en este proyecto y por su orientación fundamental.

Licencia



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

Resumen

En un mundo cada vez más conectado y digital, el desarrollo de aplicaciones web y móviles ha adquirido un papel fundamental en la vida cotidiana de las personas. La creación de soluciones tecnológicas versátiles y eficientes se ha convertido en un desafío apasionante y necesario. Este proyecto se enfoca en el desarrollo de una aplicación full-stack que comprende la creación de una página web totalmente funcional y un prototipo de una aplicación móvil. El proyecto está destinado a ofrecer una experiencia de interacción a través de varios chat de inteligencia artificial con la finalidad de que el usuario pueda comparar las distintas respuestas y seleccionar la que mejor se adapte a su consulta además de poder valorarlas y etiquetarlas.

La importancia de este proyecto radica en su capacidad para abordar diversas áreas clave de la informática, incluyendo el desarrollo web (frontend haciendo uso de Angular con Typescript y backend haciendo uso de Flask con Python), el desarrollo de aplicaciones móviles, manejo de una base de datos (Supabase) y la implementación de sistemas de inteligencia artificial (a partir de los modelos proporcionados de la plataforma Fireworks).

Además, se ha realizado un estudio de la viabilidad económica de la aplicación, la duración estimada del proyecto es de 8 meses y 3 semanas. Por otro lado, el retorno de la inversión se obtendría aproximadamente a los 16 meses de su comercialización.

Este trabajo promete no solo un desafío técnico, sino también la oportunidad de marcar una diferencia en la forma en que las personas interactúan con la tecnología y gestionan su información.

Palabras clave: Inteligencia artificial, aplicación, chat múltiple, planificación

Abstract

In an increasingly connected and digital world, the development of web and mobile applications has acquired a crucial role in people's daily lives. Creating versatile and efficient technological solutions has become an exciting and necessary challenge. This project focuses on the development of a full-stack application, including the creation of a fully functional website and a prototype of a mobile application. The project aims to provide an interactive experience through various artificial intelligence chat systems, allowing users to compare different responses, select the one that best suits their query, and provide ratings and tags.

The significance of this project lies in its ability to address various key areas of computer science, including web development (frontend using Angular with TypeScript and backend using Flask with Python), mobile application development, database management (Supabase), and the implementation of artificial intelligence systems (based on the models provided by the Fireworks platform).

In addition, an economic feasibility study of the application has been conducted, estimating a project duration of 8 months and 3 weeks. Moreover, the return on investment is expected around the 16th month of its commercialization.

This work not only promises a technical challenge but also the opportunity to make a difference in how people interact with technology and manage their information.

Keywords: Artificial intelligence, application, multiple chat, planning

Índice general

1. Introducción	1
1.1. Definición del problema	1
1.2. Justificación	3
1.3. Estado actual del mercado	5
1.4. Tendencia de mercado	8
1.5. Objetivos	9
2. Estudio Previo	11
2.1. Lenguajes de programación y frameworks	11
2.1.1. Frontend	11
2.1.2. Backend	12
2.1.3. Base de datos	14
2.2. Entorno de desarrollo	15
2.2.1. Editor de Código - Visual Studio Code	15
2.2.2. Control de Versiones - Git y GitHub	16
2.2.3. Otras tecnologías y herramientas	17
2.3. Plataformas para realizar el despliegue	18
2.3.1. Servicios en la Nube	18
2.3.2. Plataformas de alojamiento web compartido	19
2.4. Dificultades encontradas en las tecnologías	20
2.4.1. Compatibilidad entre navegadores	20
2.4.2. Optimización para dispositivos móviles	20
2.4.3. Seguridad	21
2.4.4. Rendimiento y Tiempo de Carga	21
2.5. Aplicaciones similares	21
3. Desarrollo de arquitecturas	23
3.1. Elección de tecnologías y herramientas	23
3.2. Arquitectura	24
3.3. Frontend	25
3.3.1. src/app	25
3.3.2. src/app/quire-quest	26
3.3.3. Páginas	26
3.4. Backend	29
3.4.1. Models	29
3.4.2. Server	30
3.4.3. requirements.txt	30
3.4.4. app.py	30

3.5. Base de datos	31
3.5.1. Chats	31
3.5.2. Conversations	31
3.5.3. Messages	32
3.5.4. Models	32
3.5.5. Users	32
3.6. Despliegue de la aplicación	32
3.6.1. Funcionamiento	33
3.6.2. Ventajas	33
3.7. Resultados	33
4. Estudio de viabilidad económica	34
4.1. Funcionalidades adicionales	34
4.2. Desarrollo del proyecto	36
4.3. Coste y duración del proyecto	39
4.4. Modelo de comercialización	40
4.5. Estimación del retorno de la inversión [63]	41
5. Conclusiones y líneas futuras	46
6. Summary and Conclusions	47
7. Presupuesto	48

Índice de Figuras

1.1. Casos de uso de IA más comúnmente adoptados [10]	2
1.2. Mercado de IA generativa. Concentraciones y características [11]	6
1.3. Mercado de la IA generativa en Estados Unidos [11]	9
1.4. Mercado de Chatbots en Norte América [12]	9
2.1. Logos frameworks de frontend	12
2.2. Logos de lenguajes de programación	13
2.3. Logos frameworks de backend	13
2.4. Logos base de datos	15
2.5. Logo del editor Visual Studio Code	16
2.6. Logos de Git y GitHub	16
2.7. Logos de Figma, Project Libre y PWA	18
2.8. Logos de los servicios en la Nube	19
2.9. Logos de plataformas de alojamiento web compartido	20
2.10. Plataforma de Vercel AI Playground	22
3.1. Página principal de la aplicación en la versión web y móvil	26
3.2. Página de inicio de sesión en la versión web y móvil	27
3.3. Página de registro de usuarios en la versión web y móvil	27
3.4. Página de chat en la versión web y móvil	28
3.5. Relación y estructura de las tablas de la base de datos	31
3.6. QR con acceso a la aplicación web	33
4.1. Funcionalidades de la aplicación profesional	35
4.2. Tareas definidas para el desarrollo del proyecto	37
4.3. Diagrama de Gantt (1)	38
4.4. Diagrama de Gantt (2)	38
4.5. Diagrama de Gantt (3)	38
4.6. Recursos necesarios para llevar a cabo la aplicación profesional	40
4.7. Usuarios potenciales y los que llegarían a usarla y/o pagarla	42
4.8. Promedio de usuarios en cada plan y las ganancias mensuales	43
4.9. Gastos a tener en cuenta	44
4.10. Evolución de los gastos frente a los ingresos	45
4.11. Punto en el que se empieza a tener beneficios	45

Índice de Tablas

4.1. Características de cada uno de los planes de suscripción	41
7.1. Presupuesto del proyecto de Trabajo de Fin de Grado	48

Capítulo 1

Introducción

1.1. Definición del problema

La inteligencia artificial está experimentando una rápida expansión en diversos sectores de nuestra sociedad, evidenciando un cambio paradigmático que parece imparable. Esta tecnología está redefiniendo las dinámicas empresariales, al posibilitar la toma de decisiones fundamentadas en datos, así como la automatización y mejora de la experiencia del cliente.

La notable popularidad de ChatGPT¹ [1] ha marcado un hito en la adopción pública de la inteligencia artificial, permitiendo a individuos de todas partes experimentar el potencial de esta tecnología de manera tangible. Los grandes modelos lingüísticos (LLM)² y los modelos subyacentes han representado un avance significativo en la inteligencia artificial generativa. Estos, no solo han logrado descifrar la complejidad del lenguaje, posibilitando que las máquinas comprendan el contexto, deduzcan intenciones y sean creativas de forma autónoma, sino que también pueden adaptarse rápidamente a una amplia variedad de tareas.

Esta tecnología está diseñada para una transformación radical en diversos campos, desde la ciencia y los negocios, hasta la atención médica y la sociedad en su conjunto. El impacto positivo en la creatividad y productividad humanas promete ser considerable.

Las empresas se embarcarán en la adopción de estos modelos, con el propósito de reinventar sus procesos de trabajo. Todas las funciones organizacionales pueden ser objeto de una reinención, ya que la colaboración entre humanos y sistemas de inteligencia artificial se está convirtiendo en la norma, ampliando significativamente las capacidades de los profesionales. La inteligencia artificial generativa influirá en las funciones laborales, no en los profesionales per se, dado que algunas tareas se automatizarán, otras serán asistidas por la inteligencia artificial y otras permanecerán inalteradas.

Tal y como se puede observar en la Figura 1.1, los datos de la encuesta revelan que las empresas están adoptando la inteligencia artificial (IA) en una variedad de funciones empresariales, con un enfoque particular en operaciones de servicio, desarrollo de productos y marketing/ventas.

¹**ChatGPT:** Aplicación de chatbot de inteligencia artificial desarrollado en 2022 por OpenAI que se especializa en el diálogo.

²**LLM:** Modelo de lenguaje que consta de una red neuronal con muchos parámetros, entrenados en grandes cantidades de texto sin etiquetar mediante aprendizaje autosupervisado o aprendizaje semisupervisado.



Figura 1.1: Casos de uso de IA más comúnmente adoptados [10]

En primer lugar, la optimización de las operaciones de servicio, que abarca la automatización de centros de contacto y servicios predictivos, es el caso de uso más común, con un 24 % de los encuestados indicando su adopción. Esto sugiere un fuerte interés en mejorar la eficiencia y la calidad de los servicios prestados a los clientes a través de la IA.

El desarrollo de nuevos productos basados en IA y las mejoras continuas en productos existentes ocupan el segundo lugar, con un 20 % y un 19 %, respectivamente. Esto indica que las empresas están aprovechando la IA para innovar en sus ofertas de productos y servicios, así como para mejorar la experiencia del cliente a través de características más personalizadas y avanzadas.

En el ámbito del marketing y las ventas, el análisis de servicio al cliente, la segmentación de clientes y la adquisición de clientes y generación de leads son funciones clave donde la IA está siendo utilizada, cada una con un 19 % y un 17 % de adopción, respectivamente. Esto refleja el interés en utilizar la IA para comprender mejor las necesidades y comportamientos de los clientes, así como para identificar oportunidades de ventas y marketing más efectivas.

En cuanto al análisis de riesgos, aunque ocupa un lugar inferior en la lista, con un 15 % de adopción, sigue siendo una función importante donde la IA está siendo aplicada para ayudar a las empresas a evaluar y gestionar mejor los riesgos asociados con sus operaciones y decisiones comerciales.

En resumen, los datos muestran que las empresas están adoptando la IA en una variedad de funciones empresariales, con un enfoque particular en mejorar las operaciones

de servicio, desarrollar productos innovadores y optimizar las estrategias de marketing y ventas. Esto refleja el reconocimiento del valor potencial de la IA para impulsar la eficiencia, la innovación y el rendimiento comercial en diversas áreas de la empresa.

En esta etapa de implementación, la mayoría de las empresas están iniciando sus incursiones en el empleo de modelos básicos preconfigurados. No obstante, se reconoce que el valor primordial radicará en la capacidad de ajustar o personalizar estos modelos mediante la incorporación de datos propios que aborden las necesidades específicas de cada entidad.

Las aplicaciones de inteligencia artificial generativa y modelos lingüísticos de gran escala están disponibles de manera accesible a través de interfaces de programación de aplicaciones (API)³. Además, las empresas pueden adaptar estos recursos a sus requisitos mediante procesos de ingeniería básica, tales como ajustes rápidos o la implementación de aprendizaje basado en prefijos. Con el propósito de potenciar el valor de la inteligencia artificial generativa y los modelos básicos en aplicaciones comerciales específicas, las organizaciones están cada vez más inclinadas a personalizar los modelos preentrenados, ajustándolos con sus propios datos y, de esta manera, optimizando su rendimiento.

1.2. Justificación

La inteligencia artificial generativa posibilita la multimodalidad de los modelos, lo que implica su capacidad para procesar simultáneamente diversas modalidades, como imágenes y texto, ampliando así sus áreas de aplicación y aumentando su versatilidad. Además, esta tecnología mejora la interfaz entre humanos y computadoras, permitiendo la comunicación a través de lenguaje natural en lugar de requerir conocimientos en lenguajes de programación.

En términos empresariales, la inteligencia artificial generativa tiene el potencial de transformar las operaciones al abrir nuevas oportunidades en términos de automatización, innovación y personalización. Este enfoque no solo conlleva beneficios económicos al reducir costos, sino que también optimiza la experiencia del cliente. Un ejemplo paradigmático de esta capacidad es el lanzamiento de GrammarlyGo [2] por parte de Grammarly, Inc. en marzo de 2023, que incorpora funciones generativas de inteligencia artificial para facilitar la redacción, edición y personalización de texto.

Es fundamental destacar el concepto de *Mixtures of experts* en el ámbito de la inteligencia artificial y el aprendizaje automático. Este enfoque implica la combinación de varios modelos más simples, denominados *expertos*, para constituir un modelo más complejo y flexible. Es particularmente valioso al enfrentarse a conjuntos de datos complejos y no lineales.

La esencia de este concepto reside en la división del espacio de entrada en regiones más manejables, asignando a cada región un experto específico. Cada experto tiene la responsabilidad de modelar y comprender una porción determinada del espacio de entrada. Posteriormente, un *selector* o *gate* se emplea para determinar qué experto debe

³API: Pieza de código que permite a diferentes aplicaciones comunicarse entre sí y compartir información y funcionalidades.

activarse ante un dato de entrada dado, considerando las características de dicho dato y seleccionando el experto más idóneo para manejar la entrada.

El modelo global se configura mediante la combinación de las salidas de todos los expertos, ponderadas por los resultados del selector. Este enfoque se revela como una técnica poderosa para abordar problemas complejos, permitiendo una adaptación eficiente a la complejidad inherente a los datos al combinar modelos más simples y especializados.

Es relevante resaltar que en la actualidad existen diversas aplicaciones que habilitan a los usuarios para realizar consultas mediante inteligencia artificial. Entre algunas de estas aplicaciones se pueden destacar la siguientes:

- **Murf** [3]: Generador de voz.
- **Jasper** [4]: Asistente de escritura.
- **Syntesia** [5]: Generador de video.
- **Trint** [6]: Transcripción de archivos, convierte archivos de audio y video en texto.
- **Otter.ai** [7]: Transcripción de conversaciones de voz.
- **Speechify** [8]: Conversión de texto en cualquier formato en voz con sonido natural.
- **Hooper** [9]: Utiliza capacidades predictivas para ayudar a los usuarios a obtener información precisa sobre los costos asociados con vuelos, habitaciones de hotel y alquiler de automóviles.

Sin embargo, se observa una carencia en la posibilidad de personalizar completamente estas aplicaciones según las necesidades individuales de los usuarios. En otras palabras, los usuarios se ven obligados a utilizar aplicaciones específicas para consultas particulares, sin contar con una solución integral que se adapte plenamente a sus requisitos.

La disponibilidad de una aplicación que pueda personalizar de manera completa su interacción con la inteligencia artificial, utilizando múltiples modelos simultáneamente, comparando los resultados y seleccionando aquellos más apropiados para cada situación, podría significar una mejora sustancial tanto para los usuarios como para las empresas. Esta capacidad facilitaría la realización de tareas de manera más eficiente, sencilla y adaptada a las necesidades específicas de cada usuario, proporcionando una experiencia más fluida y personalizada.

Por otro lado, es importante comentar que el proyecto se enfoca en desarrollar una plataforma integral que aproveche el potencial de la inteligencia artificial generativa y otras técnicas para crear una capa de conexión entre diversas IA. Esta capa permitirá contrastar respuestas, unificar diferentes tipos de IA en una misma interfaz para evitar que el usuario tenga que acceder a varias aplicaciones por separado.

La idea es crear una experiencia de usuario más fluida y personalizada al proporcionar acceso a múltiples modelos de inteligencia artificial simultáneamente. Esto significa que los usuarios podrán interactuar con la plataforma utilizando lenguaje natural y obtener respuestas o realizar tareas que involucren imágenes, texto, voz, video, entre otros tipos de datos, sin tener que cambiar entre diferentes aplicaciones.

Por ejemplo, un usuario podría utilizar la plataforma para realizar una consulta compleja que involucre la generación de texto, la interpretación de imágenes y la síntesis de voz. La plataforma utilizaría una combinación de modelos de IA especializados, cada uno experto en un tipo específico de tarea, y un selector para determinar qué modelo activar en función de la entrada del usuario.

Además, la plataforma se diseñaría para aprender y adaptarse a las preferencias y necesidades individuales de cada usuario, brindando recomendaciones personalizadas y ajustando su comportamiento en función del historial de interacciones.

En resumen, el objetivo del proyecto es proporcionar una solución integral que permita a los usuarios aprovechar el potencial de la inteligencia artificial de manera eficiente y personalizada, eliminando la necesidad de utilizar múltiples aplicaciones por separado y mejorando la experiencia de usuario en términos de facilidad de uso y personalización.

1.3. Estado actual del mercado

La fase de crecimiento del mercado experimenta una elevada actividad y su ritmo se encuentra en constante aceleración. Este mercado se distingue por la presencia de tecnologías como el procesamiento del lenguaje natural (NLP) ⁴, el aprendizaje automático (ML) ⁵, la visión por computadora y el procesamiento del habla. La expansión de las aplicaciones de tecnologías innovadoras, respaldada por la implementación de modelos de inteligencia artificial multimodal, que facilitan la integración de audio, video e imágenes para la creación de contenido nuevo, constituye un factor determinante en el crecimiento del mercado.

En diciembre de 2023, Google introdujo Gemini, un sistema multimodal de inteligencia artificial generativa capaz de comprender y mantener conversaciones inteligentes sobre diversas entradas, incluyendo imágenes, texto, voz, música, códigos de computadora, entre otros.

La estructura del mercado se caracteriza por su fragmentación y la presencia de múltiples actores a nivel global y regional. Estos participantes en el mercado realizan inversiones significativas en investigación y desarrollo (I+D) ⁶ con el propósito de desarrollar soluciones avanzadas y obtener una ventaja competitiva. Asimismo, se observa una tendencia marcada hacia la formación de asociaciones, fusiones y adquisiciones, ya que el mercado se distingue por su dinamismo, innovación y cambios rápidos.

En noviembre de 2021, IBM anunció la adquisición de SXiQ [13], una empresa australiana especializada en servicios de transformación digital, focalizada en aplicaciones y plataformas en la nube, así como ciberseguridad en la nube. Esta adquisición tuvo como

⁴**NLP**: Campo de las ciencias de la computación, de la inteligencia artificial y de la lingüística que estudia las interacciones entre las computadoras y el lenguaje humano, así como los detalles computacionales de las lenguas naturales.

⁵**ML**: Subapartado de la inteligencia artificial (IA) que se centra en desarrollar sistemas que aprenden, o mejoran el rendimiento, en función de los datos que consumen.

⁶**I+D**: Actividades contraídas por corporaciones o gobiernos, para el desarrollo de nuevos productos y servicios, o mejora de los mismos.

objetivo fortalecer la estrategia de inteligencia artificial y nube híbrida de IBM, posibilitando a las empresas modernizar y mejorar aplicaciones críticas en diversas plataformas y entornos de nube.

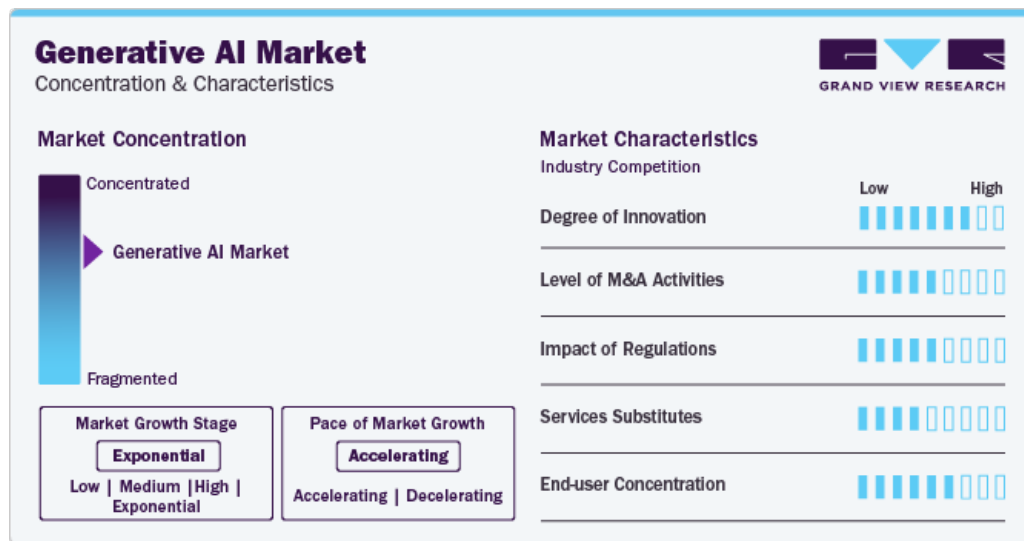


Figura 1.2: Mercado de IA generativa. Concentraciones y características [11]

Tal y como se puede observar en la Figura 1.2, el mercado de la inteligencia artificial generativa exhibe una tendencia hacia la concentración, lo que implica que un número reducido de actores domina una parte significativa del mercado. Esto puede ser indicativo de barreras de entrada sustanciales, un alto nivel de competitividad entre los principales jugadores o la presencia de economías de escala que favorecen a las empresas más grandes.

En cuanto al estado de crecimiento, el mercado de la IA generativa está experimentando un crecimiento exponencial, lo que sugiere un aumento rápido y significativo en la demanda y la adopción de estas tecnologías. Este tipo de crecimiento puede ser atribuido a varios factores, como avances tecnológicos, la creciente conciencia sobre las capacidades de la IA generativa y su aplicación en una variedad de sectores.

El ritmo de crecimiento del mercado también es notable, ya que se encuentra en una fase acelerada. Esto indica que el crecimiento no solo es significativo, sino que también está aumentando en velocidad con el tiempo.

En cuanto a las características específicas del mercado, la IA generativa muestra un alto grado de innovación, con continuos avances y desarrollos que impulsan su evolución. Además, el nivel de actividades de fusiones y adquisiciones en el mercado es notablemente alto, lo que sugiere un entorno dinámico y propenso a la consolidación entre las empresas del sector.

Las regulaciones de impacto también son relevantes en este mercado, aunque no tan restrictivas como en otros sectores, lo que permite cierto grado de flexibilidad y adaptación a medida que la tecnología continúa desarrollándose. Sin embargo, las regulaciones pueden influir en la dirección y el ritmo del crecimiento del mercado, ya sea promoviendo la innovación o estableciendo límites y requisitos que las empresas deben cumplir.

En cuanto a la sustitución de servicios, aunque es relativamente baja en comparación con otras características, indica que la IA generativa está empezando a desplazar o complementar servicios tradicionales en diversos sectores, lo que puede tener implicaciones significativas para las empresas establecidas y los modelos de negocio existentes.

Por último, la concentración "*end-user*" sugiere que un número significativo de usuarios finales concentra la demanda de soluciones basadas en IA generativa, lo que puede influir en la forma en que las empresas desarrollan y comercializan sus productos y servicios para satisfacer las necesidades de este segmento de mercado.

Asimismo, es evidente un crecimiento significativo en el mercado de chatbots en los últimos años. La creciente necesidad de ofrecer servicios al cliente de manera ininterrumpida, junto con la búsqueda de la reducción de costos operativos mediante la implementación de chatbots, ha llevado a un incremento en la demanda. Este fenómeno puede atribuirse a la preferencia creciente de los consumidores por aplicaciones de mensajería en lugar de plataformas de redes sociales.

Se anticipa que los chatbots diseñados para integrarse con aplicaciones de mensajería experimentarán una demanda significativa en el futuro. Además, los chatbots están ganando popularidad a medida que las empresas continúan automatizando sus procesos de ventas y servicios al cliente. La utilización de chatbots permite a las organizaciones ofrecer servicios de manera oportuna a costos reducidos.

En la actualidad, se observa un aumento notorio en el uso de servicios de mensajería, superando la cantidad de usuarios en comparación con las plataformas de redes sociales. Paralelamente, se evidencia un desarrollo significativo en el campo del procesamiento del lenguaje natural, facilitando las interacciones entre lenguajes informáticos y humanos. El ecosistema del chatbot comprende diversas funciones interconectadas que contribuyen a una aplicación óptima.

Los usuarios interactúan con el servicio de atención al cliente a través de la interfaz del chatbot, que se despliega en páginas web, aplicaciones móviles o mediante SMS, entre otros canales. Los chatbots se implementan en diversos entornos, como plataformas de mensajería independientes, sitios web y aplicaciones de terceros. Independientemente, los chatbots se integran con las aplicaciones y software empresariales. Los usuarios emplean los chatbots para elevar solicitudes o presentar quejas al servicio de atención al cliente.

El chatbot, como servicio de mensajería, se desarrolla mediante un conjunto de reglas e inteligencia artificial, permitiendo la interacción a través de una interfaz de chat. Se distinguen dos tipos de chatbots: aquellos desarrollados mediante reglas predefinidas y aquellos que emplean inteligencia artificial y algoritmos de aprendizaje automático. Algunos chatbots incorporan reconocimiento de voz para proporcionar una experiencia mejorada en el servicio al cliente. Los desarrolladores de chatbots, actualmente, incluyen análisis en las aplicaciones de software para obtener una comprensión más profunda del comportamiento y patrones de compra de los clientes.

OpenAI [14], una empresa estadounidense especializada en inteligencia artificial, lanzó ChatGPT, un chatbot de IA que responde a diversas preguntas. Esta última iteración se enmarca en la serie de IA denominada GPT. La estructura del chat permite a la IA abordar

preguntas de seguimiento, reconocer errores, cuestionar suposiciones incorrectas y rechazar solicitudes inapropiadas.

En el contexto de la propagación del coronavirus, diversas agencias de marketing, organizaciones financieras y establecimientos de comestibles han adoptado completamente la implementación de chatbots y herramientas de inteligencia artificial conversacional para mejorar el servicio al cliente durante la crisis.

1.4. Tendencia de mercado

El panorama futuro de la inteligencia artificial (IA) se encuentra en un vertiginoso proceso de aceleración, marcando el inicio de una era transformadora que redefinirá sustancialmente el acceso a la información, la creación de contenido, la atención al cliente y la gestión empresarial.

Al integrar la IA generativa y modelos básicos en el núcleo central de las operaciones empresariales, se prevé una optimización de tareas, mejora de capacidades y la apertura de nuevas vías de crecimiento. Estas tecnologías, en su desarrollo, también darán origen a un lenguaje empresarial completamente innovador. La clave para aprovechar plenamente el potencial de estos avances en IA radica en reformular la manera de trabajar y asistir a las personas en su adaptación a estos cambios tecnológicos. Las empresas deben invertir tanto en la evolución de sus operaciones y la capacitación de su personal como en la adopción de tecnologías.

La creciente necesidad de construir mundos virtuales en el metaverso, la implementación de grandes modelos de lenguaje y la capacidad conversacional de GenAI ⁷ contribuyen aún más al aumento de la demanda en el mercado. La IA generativa, aplicada en la industria de los juegos, transforma la creación de juegos, las experiencias de los jugadores y la producción de contenido.

La pandemia de COVID-19, al impulsar la digitalización en todas las industrias debido al cambio al trabajo en línea, ha tenido un impacto positivo en el mercado de IA. Más del 53% de los profesionales de TI, según el índice global de adopción de IA 2022 de IBM, han acelerado el despliegue de la IA en los últimos 24 meses en respuesta a la pandemia. Durante este período, se desarrollaron herramientas de diagnóstico avanzadas que utilizan IA para analizar datos clínicos, epidemiológicos y genómicos relacionados con la COVID-19, así como diversos sistemas de imágenes.

Por otro lado, el mercado global de chatbots alcanzó una valoración de 5.132,8 millones de dólares en el año 2022, y se anticipa un crecimiento continuado a una tasa de crecimiento anual compuesta (CAGR) del 23,3% desde 2023 hasta 2030. Este aumento se atribuye en gran medida a la creciente adopción de actividades de servicio al cliente por parte de las empresas, orientadas a la reducción de costos operativos.

⁷**GenAI:** Rama de la IA que se centra en desarrollar algoritmos y modelos capaces de crear contenido nuevo y original.

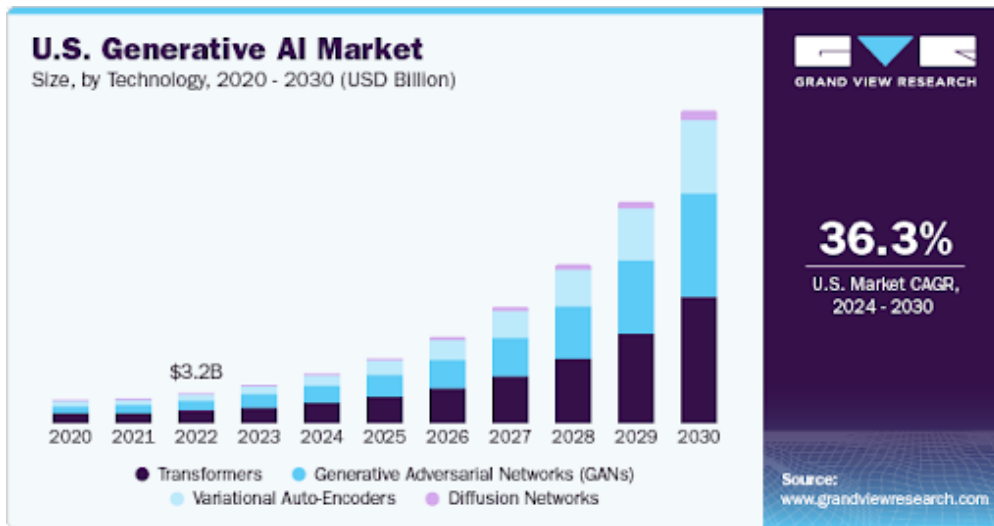


Figura 1.3: Mercado de la IA generativa en Estados Unidos [11]

Un chatbot, una aplicación colaborativa desarrollada mediante tecnología de inteligencia artificial y un conjunto específico de reglas, se diseñó para interactuar con humanos a través de conversaciones textuales. Integrado con diversas funciones de mensajería, proporciona soporte a usuarios en diversos sectores. Se anticipa que las continuas innovaciones en inteligencia artificial y tecnologías de aprendizaje automático mejorarán las capacidades de los chatbots, generando así un aumento en la demanda del mercado.

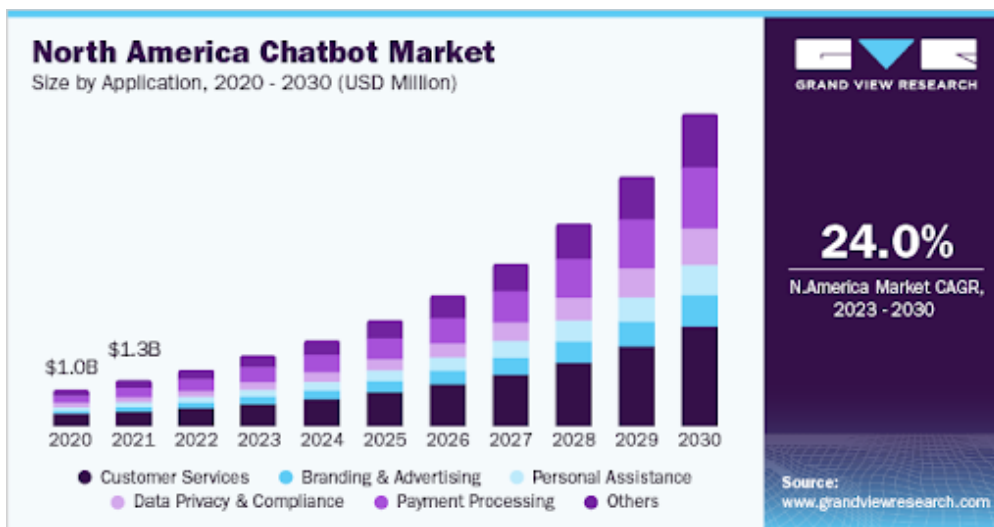


Figura 1.4: Mercado de Chatbots en Norte América [12]

1.5. Objetivos

En el presente estudio se busca determinar si es factible la creación de una aplicación web y móvil que posibilite la interacción simultánea de diversos modelos de inteligencia artificial integrando diversas funcionalidades.

Para dar respuesta a este planteamiento, se han establecido los siguientes objetivos:

- Evaluar las soluciones actualmente disponibles en el mercado en relación con el uso de inteligencia artificial para el chat y la resolución de problemas.
- Investigar las herramientas y tecnologías disponibles para el desarrollo del proyecto, seleccionando aquellas más adecuadas a sus requisitos.
- Desarrollar arquitecturas:
 - Creación de un backend.
 - Implementación de un sistema para el procesamiento de información solicitada por los usuarios.
 - Creación de una API REST ⁸ para facilitar el acceso a distintos modelos de inteligencia artificial.
 - Desarrollar frontend tanto web como móvil que posibiliten la visualización adecuada de los datos generados tras el procesamiento.
 - Interconectar las diferentes arquitecturas para asegurar su funcionamiento conjunto.
 - Verificar el correcto desempeño de la aplicación web y móvil.
- Elaborar un plan de negocio para la comercialización del producto profesional:
 - Creación de un diagrama de Gantt ⁹ con la definición de tareas, duraciones, recursos y costos.
 - Creación de la estimación del costo total del proyecto.
 - Diseñar un modelo de comercialización.
 - Calcular el retorno de inversión (ROI) ¹⁰.

⁸**API REST:** Interfaz de programación de aplicaciones (API o API web) que se ajusta a los límites de la arquitectura REST y permite la interacción con los servicios web de RESTful.

⁹**Diagrama de Gantt:** Herramienta gráfica cuyo objetivo es exponer el tiempo de dedicación previsto para diferentes tareas o actividades a lo largo de un tiempo total determinado.

¹⁰**ROI:** Razón financiera que compara el beneficio o la utilidad obtenida y su relación con la inversión realizada, es decir, «representa una herramienta para analizar el rendimiento que la empresa tiene desde el punto de vista financiero».

Capítulo 2

Estudio Previo

En siguiente apartado, se explorarán diversas tecnologías y herramientas consideradas durante la fase de desarrollo del proyecto. Posteriormente, se discutirán las opciones óptimas, así como los desafíos identificados en el análisis de estas tecnologías.

2.1. Lenguajes de programación y frameworks

2.1.1. Frontend

Entre los distintos lenguajes de programación que existen a día de hoy, los que son esenciales para la creación del frontend de las páginas web son HTML, CSS y JavaScript [15, 16, 17].

- **HTML** (Hypertext Markup Language): Lenguaje fundamental para estructurar el contenido de una página web. Utiliza etiquetas para definir elementos como encabezados, párrafos, enlaces e imágenes.
- **CSS** (Cascading Style Sheets): Se utiliza para dar estilo y formato a los elementos HTML. Define la apariencia visual, como colores, fuentes, márgenes y alineaciones.
 - **Tailwind CSS** [18]: Marco de trabajo (framework¹) de hojas de estilo en cascada (CSS) que se utiliza para diseñar sitios web de manera rápida y eficiente.
 - **Bootstrap** [19]: Marco de diseño (framework) de código abierto desarrollado por Twitter. Se utiliza para facilitar el desarrollo web y la creación de sitios responsivos y móviles.
 - **Materialize** [20]: Marco CSS receptivo moderno basado en Material Design de Google.
- **JavaScript**: Es un lenguaje de programación del lado del cliente que permite la interactividad en las páginas web. Puede manipular el contenido HTML, responder a eventos del usuario y realizar solicitudes a servidores.

Por otro lado, para poder llevar a cabo la creación de una página web con un código más limpio y consistente de una forma más rápida, sencilla y eficaz será necesario hacer uso de algún framework. Entre algunos de los más populares se pueden destacar los siguientes:

¹**Framework**: Conjunto de reglas y convenciones que se usan para desarrollar software de manera más eficiente y rápida.

- **React** [21]: Es una biblioteca de JavaScript desarrollada por Facebook. Se centra en la construcción de interfaces de usuario interactivas mediante la creación de componentes reutilizables. Utiliza un enfoque basado en componentes, lo que facilita el mantenimiento y la escalabilidad del código.
- **Ionic** [22]: Es un marco de desarrollo de código abierto para construir aplicaciones móviles y web utilizando tecnologías web estándar como HTML, CSS y JavaScript. Se centra en la creación de aplicaciones híbridas.
- **Angular** [23]: Es un marco de desarrollo web mantenido por Google. Proporciona un conjunto integral de herramientas y características para el desarrollo, prueba y despliegue de aplicaciones web complejas. Utiliza TypeScript, lo que brinda una mejora en la productividad y la detección de errores durante el desarrollo.
- **Vue.js** [24]: Es un marco progresivo para construir interfaces de usuario. Es apreciado por su simplicidad y flexibilidad. Permite la creación de aplicaciones interactivas de manera incremental, lo que significa que puedes integrar Vue.js en partes específicas de tu aplicación en lugar de adoptarlo por completo desde el principio.

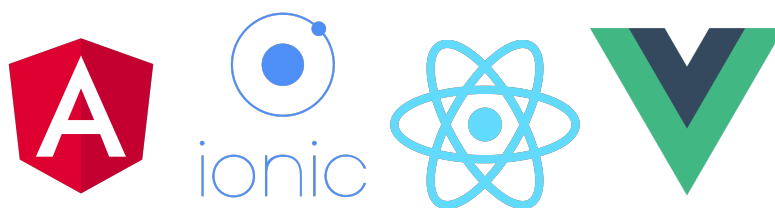


Figura 2.1: Logos frameworks de frontend

2.1.2. Backend

El backend de una aplicación es la parte responsable del procesamiento de datos, lógica y gestión de la base de datos. Existen varios lenguajes de programación y herramientas populares para el desarrollo del backend.

En cuanto a los lenguajes más usados para el backend de una aplicación se pueden destacar algunos de ellos como por ejemplo Python, Java, Ruby, PHP, Go (Golang), etc.

- **Python** [25]: Es conocido por su sintaxis clara y legible. Flask y Django son dos populares frameworks para el desarrollo web con Python.
- **Java** [26]: Es un lenguaje orientado a objetos utilizado ampliamente en el desarrollo backend. Spring es uno de los frameworks más utilizados para aplicaciones empresariales.
- **Ruby** [27]: Es conocido por su sintaxis elegante y simple. Ruby on Rails es un framework web popular para desarrollo rápido.
- **PHP** [28]: Es un lenguaje diseñado para el desarrollo web. Frameworks como Laravel y Symfony son ampliamente utilizados.

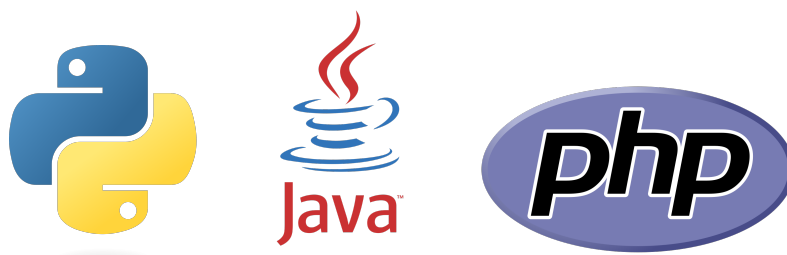


Figura 2.2: Logos de lenguajes de programación

- **Go (Golang)** [29]: Es conocido por su rendimiento y eficiencia. Es particularmente adecuado para sistemas distribuidos y concurrencia.

Por otro lado, también será necesario el uso de herramientas y frameworks para el backend, para así poder asegurar la correcta comunicación entre el frontend, el backend y la base de datos de la aplicación. Entre ellas se destacan las siguientes:

- **Express.js** [30]: Un framework minimalista para Node.js que simplifica el desarrollo de aplicaciones web y API.
- **Django** [31]: Un framework web para Python que sigue el principio de "baterías incluidas", proporcionando muchas herramientas integradas.
- **Spring Boot** [32]: Un framework para Java que simplifica el desarrollo de aplicaciones Java basadas en Spring.
- **Ruby on Rails** [33]: Un framework web para Ruby que sigue el principio de convención sobre configuración para un desarrollo rápido y fácil.
- **Laravel** [34]: Un framework web para PHP que prioriza la elegancia y la simplicidad, facilitando tareas comunes en el desarrollo web.
- **Flask** [35]: Un microframework para Python que es ligero y fácil de usar, adecuado para proyectos pequeños y medianos.



Figura 2.3: Logos frameworks de backend

2.1.3. Base de datos

La elección de la base de datos es crucial en el desarrollo de aplicaciones, ya que influye directamente en la eficiencia del almacenamiento, recuperación y manejo de datos. Las bases de datos se pueden clasificar en dos grandes categorías: Relacionales (SQL²) y No Relacionales (NoSQL³).

Relacionales (SQL)

- **MySQL** [36]: Ampliamente utilizado y soportado, robusto, confiable y bien documentado. Además es excelente para aplicaciones que requieren transacciones complejas. Sin embargo, puede enfrentar problemas de escalabilidad con bases de datos extremadamente grandes o bajo cargas de trabajo muy altas.
- **PostgreSQL** [37]: Soporta un rango más amplio de tipos de datos y operaciones avanzadas y es muy conforme a los estándares SQL y extensible. Por otro lado, hay que tener en cuenta que es más complejo y puede requerir más recursos del sistema que MySQL en algunas configuraciones.
- **Oracle Database** [38]: Altamente escalable, soporta una gran cantidad de operaciones paralelas y transacciones además de ser muy seguro. Sin embargo, no hay que olvidarse de su complejidad y costos de licencia pueden ser prohibitivos para startups y pequeñas empresas.

No Relacionales (NoSQL)

- **MongoDB** [39]: Consta de esquemas de datos flexibles que permiten almacenar documentos JSON además de tener buen rendimiento para grandes volúmenes de datos y consultas simples. Sin embargo, las transacciones son menos robustas en comparación con las bases de datos relacionales. La gestión del tamaño de almacenamiento puede ser desafiante.
- **Redis** [40]: Es una base de datos en memoria, ofrece altísima velocidad para operaciones de lectura/escritura. Es interesante destacar que soporta estructuras de datos complejas. También añador que la limitación de memoria puede ser un problema para conjuntos de datos grandes.

La elección entre SQL y NoSQL depende de la naturaleza de los datos y los requisitos de la aplicación. Las bases de datos relacionales son ideales para datos estructurados y relaciones complejas, mientras que NoSQL se adapta mejor a esquemas de datos flexibles y escalabilidad horizontal.

Por otro lado, las bases de datos NoSQL generalmente ofrecen mejor escalabilidad horizontal que las bases de datos relacionales. Además, hay que tener en cuenta que

²**SQL**: Structured Query Language. Lenguaje específico de dominio, diseñado para administrar, y recuperar información de sistemas de gestión de bases de datos relacionales.

³**NoSQL**: Amplia clase de sistemas de gestión de bases de datos que difieren del modelo clásico de sistema de gestión de bases de datos relacionales en aspectos importantes, siendo el más destacado que no usan SQL como lenguaje principal de consultas.

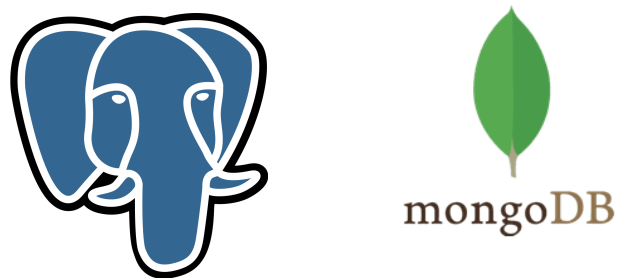


Figura 2.4: Logos base de datos

las bases de datos SQL ofrecen fuertes garantías de consistencia (ACID)⁴, mientras que NoSQL tiende a enfocarse en la disponibilidad y partición de la tolerancia (Teorema CAP)⁵, con algunas excepciones.

La elección de la base de datos debe basarse en los requisitos específicos del proyecto, incluyendo el tipo de datos, volumen, velocidad de acceso, consistencia, y la experiencia del equipo de desarrollo.

2.2. Entorno de desarrollo

El entorno de desarrollo es fundamental en la crección de cualquier proyecto de software, ya que proporciona las herramientas y procesos necesarios para escribir, probar y depurar código de manera eficiente. En el contexto de un proyecto web, el entorno de desarrollo incluye desde el editor de código hasta sistemas de control de versiones, pasando por herramientas específicas para el frontend y el backend, así como plataformas para el despliegue. A continuación, se detalla un entorno de desarrollo ideal para un proyecto web, ampliando la información sobre Visual Studio Code y Git/GitHub, e introduciendo nuevas tecnologías y herramientas.

2.2.1. Editor de Código - Visual Studio Code

Visual Studio Code (VS Code) [42] es un editor de código fuente desarrollado por Microsoft que ha ganado popularidad por su ligereza, eficiencia y capacidad de personalización. A continuación, se profundiza en sus características:

- **Extensiones:** VS Code tiene un amplio repositorio de extensiones disponibles que soportan prácticamente cualquier lenguaje de programación o tecnología, desde linters y formateadores de código hasta extensiones específicas para frameworks como React, Vue.js, y Angular.
- **Depuración Integrada:** Cuenta con potentes herramientas de depuración integradas que soportan numerosos lenguajes de programación, lo que permite a los desarrolladores encontrar y corregir errores directamente desde el editor.

⁴**ACID:** Características de los parámetros que permiten clasificar las transacciones de los sistemas de gestión de bases de datos.

⁵**Teorema CAP:** Enuncia que es imposible para un sistema de cómputo distribuido garantizar simultáneamente: La consistencia, es decir, cualquier lectura recibe como respuesta la escritura más reciente o un error.

- **Soporte de Git Integrado:** Permite a los desarrolladores realizar commits, push, pull y otras operaciones de Git sin salir del editor.
- **Personalización:** Ofrece amplias opciones de personalización, incluyendo temas, configuraciones de teclado y ajustes del entorno de trabajo, lo que permite a los usuarios adaptar el editor a sus preferencias personales.



Figura 2.5: Logo del editor Visual Studio Code

2.2.2. Control de Versiones - Git y GitHub

Git [43] es un sistema de control de versiones distribuido que facilita a los equipos de desarrollo el seguimiento de cambios en el código fuente a lo largo del tiempo.

GitHub [44], por otro lado, es una plataforma de alojamiento basada en Git que proporciona herramientas adicionales para la colaboración en equipo, tales como la revisión de código, gestión de proyectos y CI/CD (Integración Continua/Despliegue Continuo)⁶.

- **Colaboración:** Facilita la colaboración entre desarrolladores, permitiendo la revisión de código a través de pull requests y la gestión de tareas y bugs mediante issues y proyectos.
- **Integraciones:** Ofrece integraciones con numerosas herramientas y servicios, lo que permite automatizar flujos de trabajo, desde pruebas automatizadas hasta despliegues.



Figura 2.6: Logos de Git y GitHub

⁶CI/CD: Práctica de desarrollo de software que es posible gracias a la automatización.

2.2.3. Otras tecnologías y herramientas

A continuación se nombrarán otras tecnologías que podrían ser necesarias para realizar un proyecto de este tipo de una forma más rápida y eficaz:

- **Supabase** [41]: Plataforma BaaS (Backend as a Service)⁷ alojada en la nube que provee a los desarrolladores una amplia gama de herramientas para crear y gestionar servicios backend. Ofrece todos los servicios y herramientas de backend necesarias para crear una aplicación escalable y segura: gestión de base de datos, autenticación, almacenamiento de archivos, generación automática de APIs y actualizaciones en tiempo real, entre otros.
- **Docker** [45]: Plataforma que permite empaquetar aplicaciones y sus dependencias en contenedores, asegurando la consistencia del entorno de desarrollo a través de distintos entornos de despliegue. Esto es especialmente útil en proyectos web para gestionar servicios como bases de datos, servidores web y aplicaciones de back-end de manera aislada y reproducible.
- **Node.js** [46]: Entorno de ejecución que permite ejecutar código JavaScript fuera del navegador.
- **Postman** [47]: Herramientas que permiten diseñar, probar y documentar APIs de manera eficiente. Son esenciales para el desarrollo de backend, ya que facilitan la prueba de endpoints de APIs RESTful sin necesidad de desarrollar un frontend.
- **Figma** [48]: Herramientas de diseño UI/UX⁸ que facilitan la creación de prototipos y mockups de aplicaciones web y móviles. Permiten la colaboración en tiempo real y ayudan a equipos de diseño y desarrollo a mantenerse sincronizados.
- **Project Libre** [49]: Software de gestión de proyectos de código abierto que se utiliza para planificar, dar seguimiento y gestionar proyectos. Es una alternativa de código abierto a Microsoft Project y proporciona herramientas para la creación de gráficos de Gantt, asignación de recursos, seguimiento de tareas y más.
- **Overleaf** [50]: Plataforma en línea para la colaboración y creación de documentos científicos y técnicos en formato LaTeX⁹.
- **PWA (Progressive Web App)** [51]: Son aplicaciones web que utilizan las últimas tecnologías para ofrecer una experiencia de usuario similar a la de una aplicación nativa, pero a través de un navegador web.
- **Fireworks.ai** [52]: Plataforma de inteligencia artificial generativa para la innovación de productos que brinda acceso a LLM y modelos de imágenes de código abierto y de última generación.

⁷**BaaS (Backend as a Service)**: Servicio que automatiza el desarrollo del back-end, mediante la tercerización de estas funciones.

⁸**UI/UX**: Conceptos usados en el diseño web para representar la interfaz de usuario y la experiencia del usuario, respectivamente.

⁹**LaTeX**: Sistema de composición de documentos ampliamente utilizado para la creación de distintos tipos de documentos profesionales.



Figura 2.7: Logos de Figma, Project Libre y PWA

Cada una de estas herramientas y tecnologías desempeña un papel crucial en la creación de un entorno de desarrollo cohesivo y eficiente, facilitando desde la escritura del código hasta el despliegue de la aplicación, asegurando así la calidad y la escalabilidad del proyecto web.

2.3. Plataformas para realizar el despliegue

Para que la página web que se vaya a crear pueda ser accesible para todos los usuarios desde cualquier navegador web, será necesario desplegar la aplicación. Para ello existen varias formas de hacerlo: a través de servicios en la nube o a través de plataformas de alojamiento web compartido.

2.3.1. Servicios en la Nube

Cuando se trata de llevar tus proyectos web al siguiente nivel, los servicios en la nube se presentan como pilares fundamentales para ofrecer escalabilidad, seguridad y una infraestructura confiable. Entre los principales actores en este espacio, se destacan algunas de las opciones líderes que ofrecen una amplia gama de servicios diseñados para satisfacer las demandas de aplicaciones modernas.

- **Amazon Web Services (AWS)** [53]: Es uno de los proveedores de servicios en la nube más grandes y completos. Ofrece una amplia gama de servicios, desde almacenamiento en la nube (Amazon S3) hasta máquinas virtuales escalables (Amazon EC2). AWS permite a los desarrolladores escalar recursos de manera eficiente y proporciona servicios específicos para alojar aplicaciones web, como AWS Elastic Beanstalk y AWS App Runner.
- **Google Cloud Platform (GCP)** [54]: Ofrece servicios en la nube que van desde el almacenamiento de objetos (Google Cloud Storage) hasta plataformas de aplicación totalmente administradas como Google App Engine. GCP se destaca por su enfoque en la inteligencia artificial y el aprendizaje automático, así como por su escalabilidad y servicios de red avanzados.
- **Microsoft Azure** [55]: Azure es la plataforma en la nube de Microsoft y ofrece una amplia gama de servicios, desde alojamiento de máquinas virtuales (Azure Virtual Machines) hasta soluciones de contenedores (Azure App Service). Azure proporciona integración estrecha con tecnologías de Microsoft, como .NET y SQL Server.

- **DigitalOcean** [56]: Se destaca por su simplicidad y precios transparentes. Ofrece máquinas virtuales (droplets) y servicios administrados como Kubernetes para el despliegue de aplicaciones.

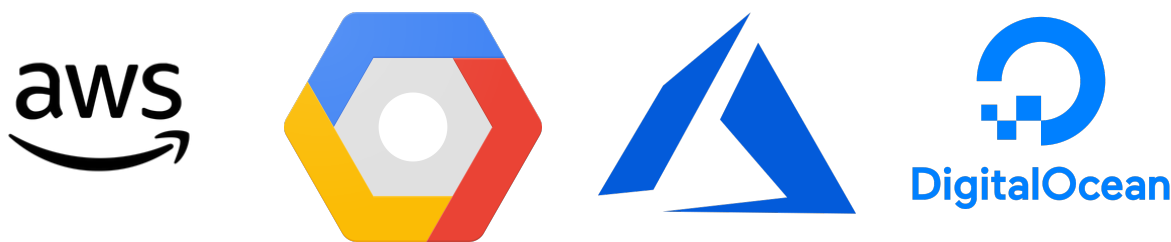


Figura 2.8: Logos de los servicios en la Nube

2.3.2. Plataformas de alojamiento web compartido

En la búsqueda de soluciones de alojamiento web compartido, existen diversas plataformas diseñadas para ofrecer simplicidad, escalabilidad y despliegue eficiente de proyectos, especialmente para aquellos de tamaño mediano o pequeño. A continuación, exploramos algunas opciones destacadas:

- **Netlify** [57]: Es una plataforma de alojamiento web que se especializa en ofrecer despliegue continuo y funciones de CDN (Content Delivery Network)¹⁰. Se integra fácilmente con repositorios de Git y proporciona certificados SSL¹¹ gratuitos, así como funciones de redireccionamiento y reescritura.
- **Vercel** [58]: Similar a Netlify, es una plataforma de alojamiento que se enfoca en proporcionar despliegue automático y funciones avanzadas para aplicaciones web y estáticas. Ofrece un enfoque sin servidor y permite integración continua directa desde repositorios de Git.
- **Heroku** [59]: Es una plataforma de desarrollo que permite a los desarrolladores construir, ejecutar y escalar aplicaciones en varios lenguajes. Ofrece una experiencia sin servidor y es conocida por su facilidad de uso y flexibilidad.
- **Render** [60]: Es una plataforma de la nube que ofrece servicios de alojamiento y despliegue de aplicaciones web, aplicaciones estáticas, despliegue de contenedores (Docker) y funciones serverless¹².

¹⁰**CDN (Content Delivery Network)**: Red de servidores interconectados que acelera la carga de las páginas web para las aplicaciones que tienen un uso intensivo de datos.

¹¹**SSL**: Tecnología estandarizada que permite cifrar el tráfico de datos entre un navegador web y un sitio web (o entre dos servidores web), protegiendo así la conexión.

¹²**Serverless**: Significa sin servidor, es una solución que permite crear y ejecutar aplicaciones con rapidez y menor costo total de propiedad.



Figura 2.9: Logos de plataformas de alojamiento web compartido

2.4. Dificultades encontradas en las tecnologías

2.4.1. Compatibilidad entre navegadores

La diversidad de navegadores web presenta un desafío constante en el desarrollo web. Aunque los estándares web buscan la uniformidad, diferentes interpretaciones y niveles de soporte entre navegadores pueden causar problemas de renderizado y funcionalidad. Para abordar esto se pueden llevar a cabo distintas estrategias:

- **Pruebas exhaustivas:** Implementar pruebas en diferentes navegadores y versiones para identificar y corregir problemas específicos.
- **Uso de prefijos:** Aplicar prefijos específicos del navegador para propiedades CSS que aún no estén completamente estandarizadas.
- **Adopción de Polyfills:** Incorporar polyfills ¹³ para funciones no admitidas en navegadores más antiguos.

2.4.2. Optimización para dispositivos móviles

El aumento del tráfico desde dispositivos móviles destaca la importancia de la optimización para proporcionar una experiencia de usuario fluida. Para asegurar esto se tendrá que tener en cuenta los siguientes aspectos:

- **Diseño responsivo:** Utilizar técnicas de diseño responsivo para adaptar la interfaz de usuario a diferentes tamaños de pantalla.
- **Optimización de imágenes:** Comprimir y entregar imágenes optimizadas para reducir el tiempo de carga en dispositivos móviles.
- **Pruebas en dispositivos reales:** Realizar pruebas en dispositivos móviles reales para identificar y solucionar problemas específicos.

¹³**Polyfills:** Fragmento de código (generalmente JavaScript en la Web) que se utiliza para proporcionar una funcionalidad moderna en navegadores antiguos que no lo admiten de forma nativa.

2.4.3. Seguridad

La seguridad es una preocupación constante, y las aplicaciones web están expuestas a diversas amenazas. Algunas medidas fundamentales son las siguientes:

- **Uso de HTTPS¹⁴:** Implementar siempre conexiones seguras para proteger la comunicación entre el navegador y el servidor.
- **Validación de entrada:** Aplicar validación de entrada en el lado del servidor para prevenir ataques como la inyección de SQL.
- **Protección contra ataques comunes:** Incorporar prácticas de desarrollo seguro para prevenir vulnerabilidades, como la protección contra la inyección de código y la manipulación de sesiones.

2.4.4. Rendimiento y Tiempo de Carga

El rendimiento es clave para la experiencia del usuario y el posicionamiento en motores de búsqueda. A continuación se enumerarán algunos de los desafíos relacionados con el tiempo de carga:

- **Optimización de código:** Minificar y combinar archivos CSS y JavaScript para reducir el tamaño de las descargas.
- **Caché eficiente:** Utilizar técnicas de almacenamiento en caché para minimizar las solicitudes al servidor y mejorar el tiempo de carga.

2.5. Aplicaciones similares

Para contextualizar la propuesta de desarrollo web, es vital explorar aplicaciones similares en el mercado. Esta investigación permitirá identificar características únicas, identificar brechas en la oferta existente y afinar la propuesta para destacar entre la competencia. Algunas aplicaciones similares que merecen análisis incluyen Vercel AI Playground, ChatGPT y Gemini (Google Bard).

- **ChatGPT:** Modelo de lenguaje desarrollado por OpenAI, basado en la arquitectura GPT-3.5. Es parte de la serie de modelos Generative Pre-trained Transformers (GPT)¹⁵ que utiliza aprendizaje profundo para comprender y generar texto de manera contextual.
- **Vercel AI Playground [61]:** Es una plataforma que permite experimentar y trabajar con varios modelos de inteligencia artificial a la vez directamente en el navegador. Permite probar y ejecutar modelos de machine learning sin necesidad de configurar un entorno local.

¹⁴**HTTPS:** Protocolo de transferencia de hipertexto seguro. Es la versión segura de HTTP, que es el principal protocolo utilizado para enviar datos entre un navegador web y un sitio web.

¹⁵**Generative Pre-trained Transformers (GPT):** Arquitectura de red neuronal profunda que se utiliza para el procesamiento del lenguaje natural (NLP) y la generación de texto. Utiliza una técnica de aprendizaje automático conocida como "transformer", que se basa en la atención y el procesamiento paralelo.

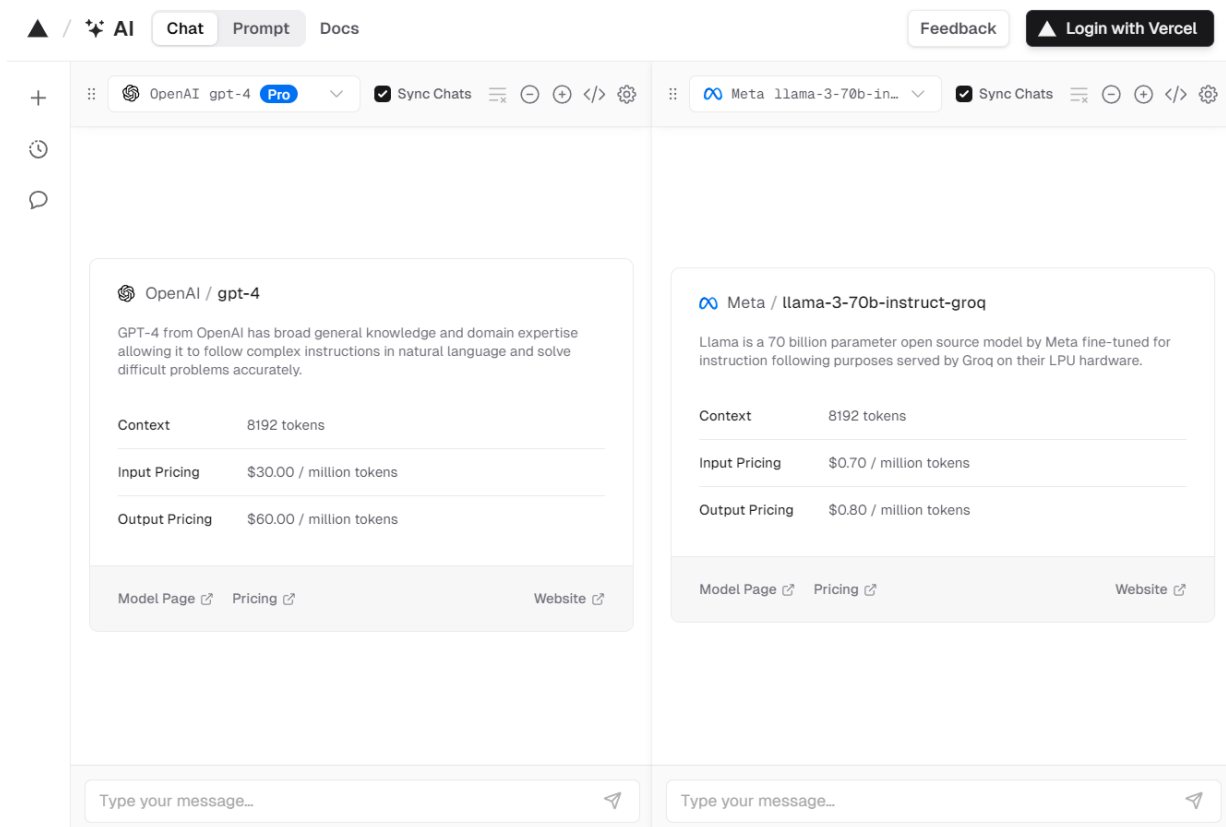


Figura 2.10: Plataforma de Vercel AI Playground

- **Gemini (Google Bard) [62]:** Bot conversacional de inteligencia artificial multimodal y generativa desarrollado por Google basado en la familia Gemini Pro.

Se examinarán aspectos como la experiencia del usuario, características distintivas y estrategias de marketing para informar y enriquecer el desarrollo de la nueva aplicación. Al comprender las fortalezas y debilidades de las aplicaciones existentes, se pueden tomar decisiones informadas para ofrecer un producto que no solo satisfaga las necesidades del usuario, sino que también se destaque en un mercado competitivo.

Capítulo 3

Desarrollo de arquitecturas

A continuación, se abordará la selección de las estructuras para nuestro prototipo, es decir, las herramientas utilizadas en el desarrollo de cada componente. También se detallarán los avances realizados en el diseño frontal, en la lógica del servidor, en las bases de datos y en la implementación de la aplicación.

3.1. Elección de tecnologías y herramientas

En la realización de este proyecto se optó por una selección cuidadosa de tecnologías que me permitieran desarrollar una aplicación fullstack multiplataforma con características específicas. El objetivo principal era crear un entorno donde los usuarios pudieran interactuar con múltiples sistemas de inteligencia artificial simultáneamente, comparando sus respuestas para seleccionar la más adecuada.

Para el desarrollo del frontend, la parte visual y de interacción directa con el usuario, se decidió emplear Angular junto con TypeScript y Tailwind CSS. Angular proporciona una estructura robusta y modular para la construcción de aplicaciones web complejas, mientras que TypeScript, al ser un superset¹ tipado de JavaScript, brinda ventajas significativas en términos de seguridad y mantenibilidad del código. Tailwind CSS, por su parte, ofrece un enfoque pragmático para el diseño web, permitiendo una rápida implementación de estilos mediante clases predefinidas.

En cuanto al backend, la parte que gestiona la lógica y la comunicación con las bases de datos, se eligió Flask, un framework ligero de Python. Flask proporciona una gran flexibilidad y simplicidad, lo que resulta ideal para proyectos de tamaño mediano que requieren una implementación ágil. Además, para la persistencia de datos, se optó por utilizar Supabase, una plataforma que se integra con PostgreSQL para ofrecer una solución de base de datos relacional en la nube, facilitando la escalabilidad y el manejo eficiente de la información.

Por otro lado, comentar que también se incorporó la plataforma Fireworks para integrar los modelos de inteligencia artificial junto con OpenAI. Fireworks proporciona una interfaz intuitiva para la gestión y despliegue de modelos de IA, lo que facilitó enormemente la incorporación y configuración de los sistemas de inteligencia artificial en la

¹**Superset:** TypeScript es un superset de Javascript. Es decir, un lenguaje que está construido encima del propio Javascript, en el sentido que agrega nuevas características al lenguaje Javascript, ofreciendo nuevas herramientas.

aplicación. Mediante esta plataforma, se pudo acceder a una variedad de modelos de IA preentrenados, así como personalizar y adaptar su funcionamiento según los requisitos específicos del proyecto.

La integración de Fireworks con OpenAI permitió aprovechar al máximo los recursos y capacidades de esta avanzada tecnología de procesamiento del lenguaje natural. De esta manera, los usuarios de la aplicación podrán interactuar de manera fluida y eficiente con múltiples sistemas de IA, comparando y evaluando sus respuestas en tiempo real.

Finalmente, para el despliegue de la aplicación, se seleccionaron Vercel y Render, plataformas de alojamiento que permiten la implementación sencilla de aplicaciones web y proporcionan herramientas para automatizar tareas de desarrollo, como la compilación y el despliegue continuo. Esta elección se basó en la facilidad de uso y en la capacidad de escalar el servicio según las necesidades del proyecto, garantizando así un funcionamiento óptimo en diferentes entornos y dispositivos.

3.2. Arquitectura

El enfoque arquitectónico utilizado en el desarrollo del proyecto se basa en una arquitectura de tipo cliente-servidor, combinada con el patrón de arquitectura de software Modelo-Vista-Controlador (MVC).

En esta arquitectura, el cliente es la interfaz de usuario de la aplicación, que se ejecuta en el navegador web del usuario y está construida utilizando tecnologías como Angular, TypeScript y Tailwind CSS. El cliente interactúa con el servidor a través de solicitudes HTTP², enviando peticiones para acceder a recursos o enviar datos al backend.

Por otro lado, el servidor consiste en el backend de la aplicación, desarrollado con Flask y Python. Este servidor se encarga de gestionar la lógica, procesar las solicitudes del cliente, interactuar con la base de datos y coordinar la comunicación con los sistemas de inteligencia artificial a través de la plataforma Fireworks y OpenAI.

La arquitectura MVC se utiliza en el lado del cliente para estructurar y organizar el código de la interfaz de usuario. En este patrón, el *modelo* representa los datos y la lógica, la *vista* es la presentación de la interfaz de usuario y el *controlador* maneja las interacciones del usuario y coordina las acciones entre el *modelo* y la *vista*.

- **Modelo (Model):** Este componente representa los datos y la lógica de la aplicación. En este caso, el *modelo* incluiría la estructura de datos que maneja la aplicación, como los mensajes del chat, las respuestas de los sistemas de inteligencia artificial, y cualquier otra información relevante. También podría incluir la lógica para procesar y analizar los datos, como la comparación de respuestas de las IA.
- **Vista (View):** La *vista* se encarga de la presentación de la interfaz de usuario. En la aplicación, la *vista* sería la parte del frontend construida con Angular, TypeScript y Tailwind CSS. Aquí se definen los elementos visuales, como botones, cuadros de texto y paneles de chat, así como el diseño y estilo de la aplicación. La *vista* se actualiza dinámicamente para reflejar los cambios en el *modelo* y para mostrar los resultados de las interacciones del usuario.

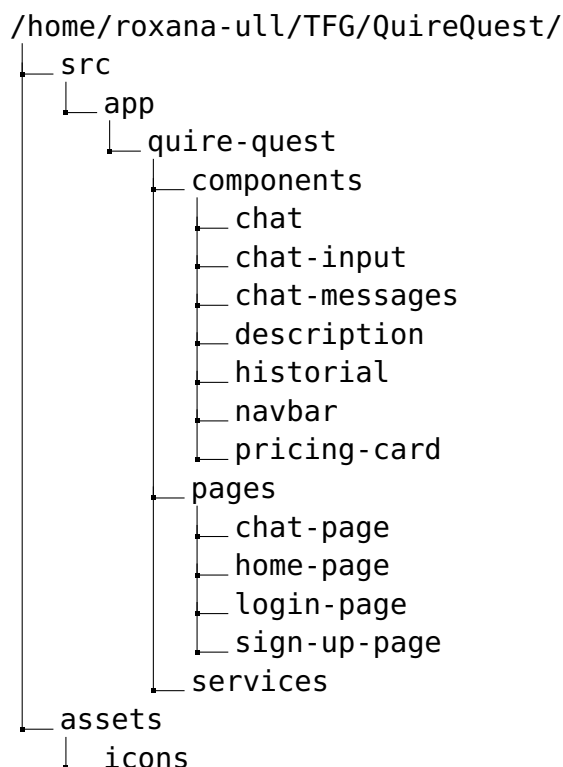
²HTTP: Se utiliza para cargar páginas web mediante enlaces de hipertexto.

- **Controlador (Controller):** El *controlador* actúa como intermediario entre el *modelo* y la *vista*, gestionando las interacciones del usuario y coordinando las acciones correspondientes. En la aplicación, el *controlador* se implementaría en el código TypeScript de Angular. Por ejemplo, puede manejar eventos de 'click' en los botones de la interfaz de usuario, enviar solicitudes al servidor para obtener o enviar datos, y actualizar el *modelo* y la *vista* en consecuencia.

Esta combinación de arquitectura cliente-servidor y el patrón MVC proporcionan una estructura modular y escalable para el desarrollo de la aplicación, permitiendo una clara separación de responsabilidades entre las distintas capas del sistema y facilitando su mantenimiento y evolución a lo largo del tiempo.

3.3. Frontend

El frontend de la aplicación constituye la interfaz de usuario con la que los usuarios interactúan. Esta sección, desglosa la estructura y funcionalidades de los archivos y directorios que componen el frontend de la aplicación, ubicados en el directorio `/src`.



3.3.1. src/app

El directorio `app` contiene todos los archivos relacionados con la lógica y componentes de la aplicación Angular:

- **app-routing.module.ts:** Define las rutas de la aplicación y las asocia con los componentes correspondientes.
- **app.component.*:** Estos archivos definen el componente principal de la aplicación, que actúa como la envoltura principal para los demás componentes.

- **app.module.ts:** Aquí se define el módulo principal de la aplicación Angular y se importan los diferentes componentes y servicios utilizados en la aplicación.

3.3.2. src/app/quire-quest

El directorio *quire-quest* contiene los componentes, páginas y servicios específicos de la aplicación QuireQuest:

- **components:** Contiene los componentes reutilizables de la app, como los componentes chat, entrada de chat, mensajes de chat, etc.
- **pages:** Aquí se encuentran las diferentes páginas de la aplicación, como la de inicio, la de chat, la de inicio de sesión, etc.
- **services:** Contiene los servicios de Angular utilizados para interactuar con la lógica y los servicios externos, como por ejemplo los servicios de chat, de mensajes, etc.

3.3.3. Páginas

A continuación se analizará la estructura y navegación de las diversas pantallas de la aplicación, tanto en la versión web como en la móvil.

- **Principal:** La pantalla principal de la aplicación web y móvil actúa como la puerta de entrada, proporcionando a los usuarios una visión general de las funcionalidades y beneficios de la aplicación. Aquí, los usuarios pueden encontrar información relevante sobre las características clave de la aplicación, así como los precios de suscripción disponibles. Además, desde esta pantalla, los usuarios tienen la opción de acceder fácilmente a la página de inicio de sesión.

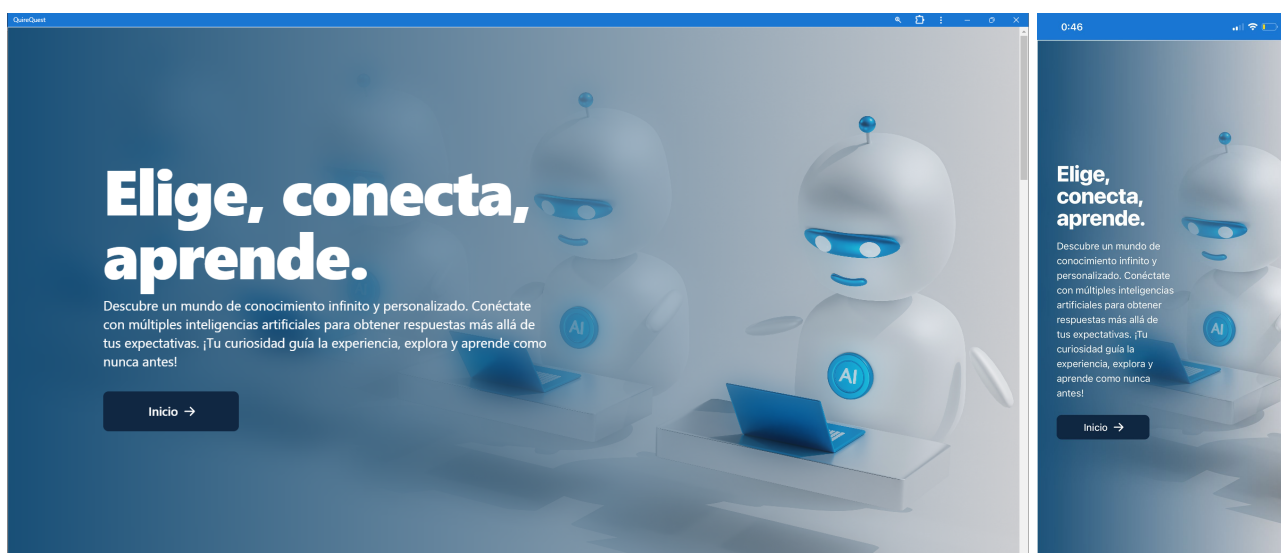


Figura 3.1: Página principal de la aplicación en la versión web y móvil

- **Inicio de Sesión:** La pantalla de inicio de sesión ofrece a los usuarios la posibilidad de ingresar a sus cuentas existentes. Aquí, los usuarios pueden introducir sus credenciales de inicio de sesión para acceder a sus perfiles. Además, la pantalla

de inicio de sesión proporciona enlaces directos para aquellos usuarios que deseen registrarse para obtener una nueva cuenta o volver a la pantalla principal en caso de no desear iniciar sesión en ese momento.

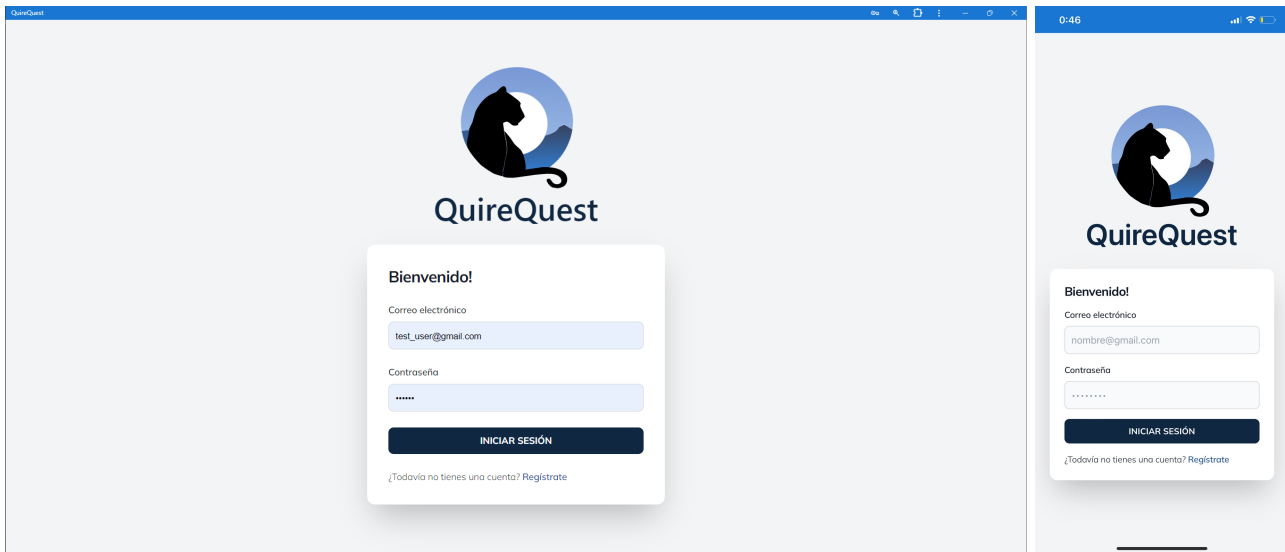


Figura 3.2: Página de inicio de sesión en la versión web y móvil

- **Registro:** La pantalla de registro ofrece a los usuarios la oportunidad de crear nuevas cuentas en la aplicación. Aquí, los usuarios pueden completar un formulario de registro con su información personal. Una vez completado el registro, los usuarios son redirigidos automáticamente a la pantalla de inicio de sesión, donde pueden iniciar sesión con las credenciales recién creadas. Además, desde la pantalla de registro, los usuarios tienen la opción de volver a la pantalla de inicio de sesión o regresar a la página principal si así lo desean.

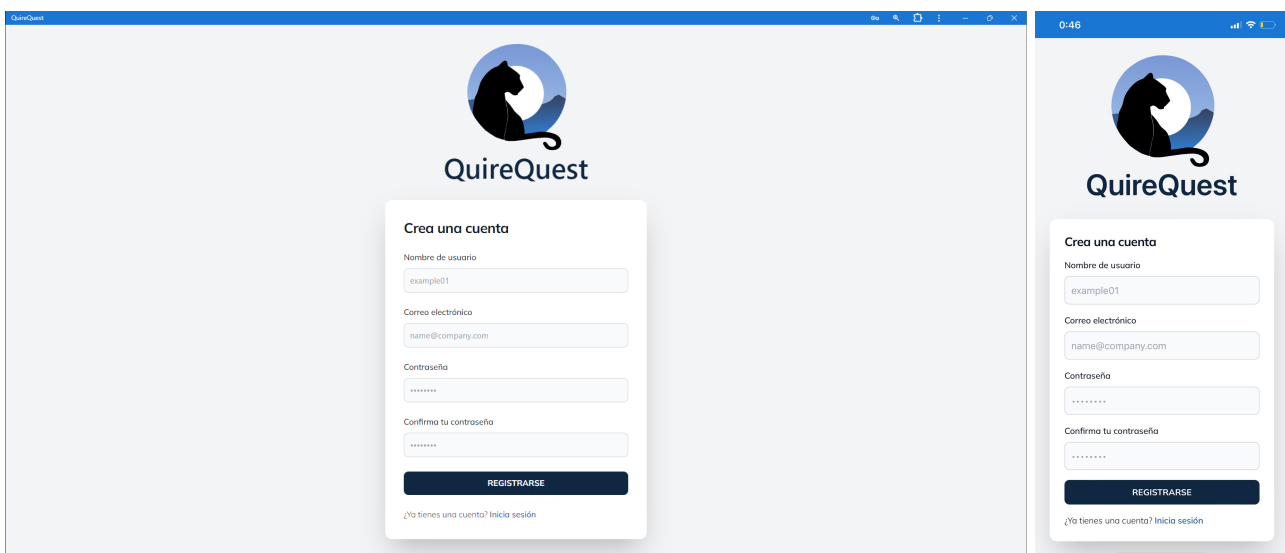


Figura 3.3: Página de registro de usuarios en la versión web y móvil

- **Chat:** La pantalla de chat es el corazón de la aplicación, donde los usuarios pueden interactuar con los diversos modelos de inteligencia artificial disponibles. Al iniciar

sesión, los usuarios son llevados directamente a esta pantalla, donde pueden iniciar conversaciones con los modelos de su elección. Esta pantalla presenta un apartado que muestra el historial de conversaciones y la opción de iniciar nuevos chats.

Además, hay dos barras de navegación que permiten seleccionar los modelos con los que desean interactuar. En el centro de la pantalla, se muestra el intercambio de mensajes entre el usuario y el modelo seleccionado, mientras que en la parte inferior se encuentra una caja de texto donde se pueden escribir las consultas.

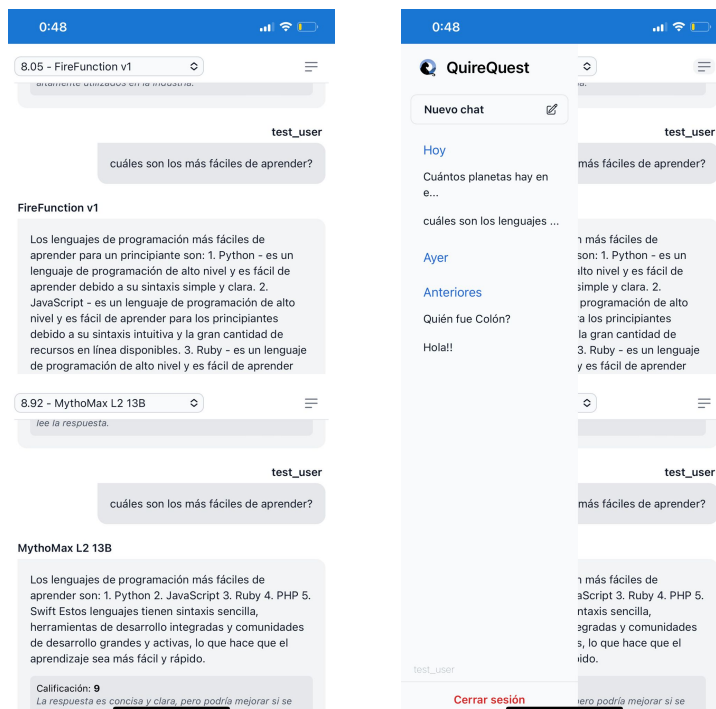
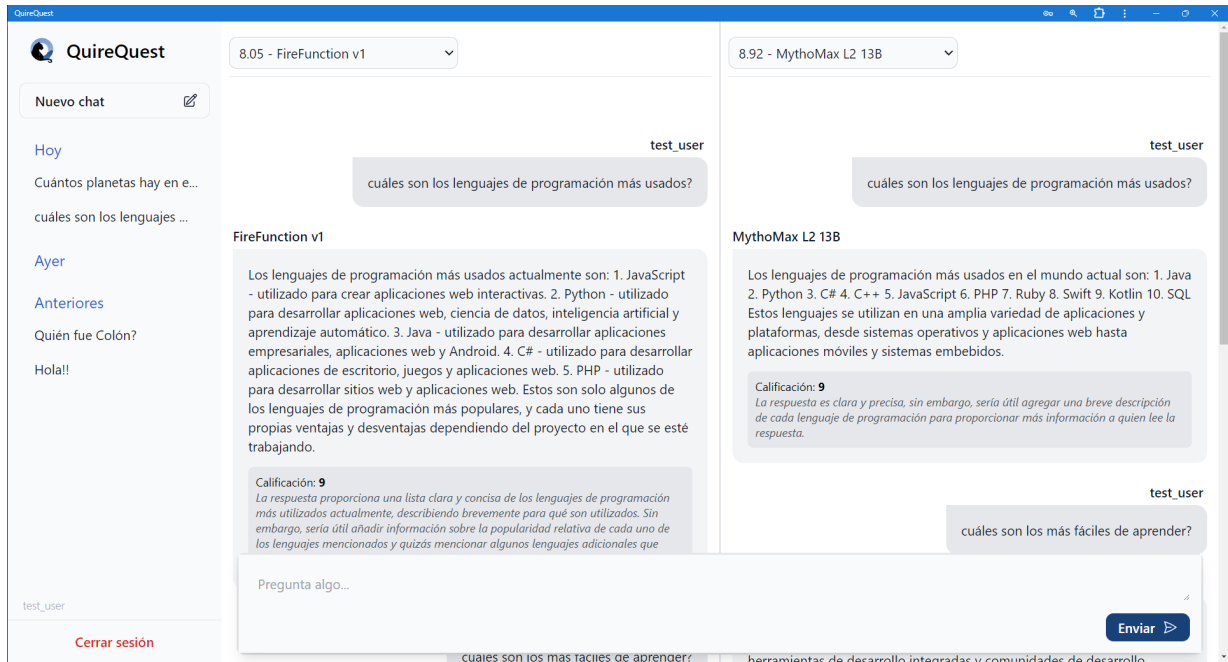


Figura 3.4: Página de chat en la versión web y móvil

3.4. Backend

El backend de una aplicación web despliega un papel crucial en la construcción y funcionamiento de sistemas modernos, sirviendo como el motor que impulsa la lógica, gestiona la persistencia de datos, y facilita la comunicación entre el cliente y los servicios externos.

En esta sección del informe, se explorará en detalle la estructura y funcionalidades del backend de la aplicación, desglosando cada uno de los componentes, archivos y directorios que componen esta parte esencial del sistema. Desde la gestión de modelos de IA hasta la configuración de rutas de la API y la interacción con la base de datos, cada aspecto del backend se ha diseñado con el objetivo de proporcionar una experiencia de usuario fluida, segura y altamente funcional.

```
/home/roxana-ull/TFG/QQ-Back/
├── README.md
├── app.py
├── models
│   ├── fire_function_v1.py
│   ├── gemma_7b_instruct.py
│   ├── llama2_13b_chat.py
│   ├── llama2_13b_code_instruct.py
│   ├── llama2_70b_chat.py
│   ├── mistral_7b_instruct.py
│   ├── mixtral_8x7b_instruct.py
│   └── mythomax_l2_13b.py
├── requirements.txt
├── server
│   ├── db.py
│   └── routes
│       ├── chat.py
│       ├── conversation.py
│       ├── message.py
│       ├── model.py
│       └── user.py
```

3.4.1. Models

El directorio `/models` alberga los archivos relacionados con la integración y gestión de los modelos de inteligencia artificial (IA) en la aplicación. Cada archivo Python dentro de este directorio representa un modelo específico de IA que se utiliza para procesar y generar respuestas en el chat de la aplicación.

Los modelos de IA son esenciales para la funcionalidad principal de la aplicación, ya que permiten interactuar con los usuarios a través del chat y ofrecer respuestas inteligentes y contextuales. Cada archivo de modelo puede contener la implementación de un modelo de lenguaje natural preentrenado o personalizado, así como lógica adicional para procesar las solicitudes del usuario y generar respuestas relevantes.

3.4.2. Server

El directorio `/server` contiene los componentes principales del backend de la aplicación, que se encargan de la lógica de negocio y la gestión de las solicitudes del cliente. Aquí se encuentran los siguientes elementos:

- **db.py**: Maneja la conexión y las operaciones con la base de datos de la aplicación.
- **routes**: Cada archivo dentro de esta carpeta corresponde a un recurso específico de la API, como usuarios, mensajes, modelos de IA, etc.
 - **chat.py**: Se implementan funciones para crear nuevos chats y actualizarlos.
 - **conversation.py**: Este archivo contiene las funciones para poder crear y obtener todas o una de las conversaciones en específico de un usuario.
 - **message.py**: En este fichero es donde se hace la comunicación con el modelo seleccionado por el usuario, se le manda el mensaje y se obtiene su respuesta.
 - **model.py**: Se definen las rutas y las funciones controladoras relacionadas con la gestión de los modelos de inteligencia artificial en la aplicación.
 - **user.py**: Este archivo contiene las rutas y las funciones controladoras para la gestión de usuarios en la aplicación. Incluye funciones para registrar nuevos usuarios, autenticar usuarios existentes y gestionar sesiones de usuario.

En conjunto, el directorio `/server` contiene la lógica principal del backend de la aplicación, que se encarga de procesar las solicitudes del cliente, interactuar con la base de datos y coordinar la comunicación entre los distintos componentes del sistema.

3.4.3. requirements.txt

El archivo `requirements.txt` especifica las dependencias de Python necesarias para ejecutar la aplicación. En este archivo se enumeran los paquetes Python junto con sus versiones correspondientes que deben instalarse para que la aplicación funcione correctamente. Este archivo se utiliza comúnmente con la herramienta `pip` para instalar todas las dependencias de una sola vez ejecutando el siguiente comando:

```
pip install -r requirements.txt
```

3.4.4. app.py

El archivo `app.py` es el núcleo principal del backend de la aplicación. En este archivo, se inicializa la aplicación Flask y se configuran todas las rutas y ajustes necesarios para que el servidor funcione correctamente. Esencialmente, `app.py` actúa como el punto de entrada para la lógica del servidor.

En `app.py`, se importa el objeto Flask de la biblioteca Flask, lo que permite crear una instancia de la aplicación. A continuación, se definen las rutas y se asignan las funciones controladoras correspondientes a cada ruta. Estas funciones controladoras son responsables de manejar las solicitudes HTTP entrantes y de ejecutar la lógica necesaria para procesarlas.

3.5. Base de datos

La base de datos subyacente de la aplicación se organiza en cinco tablas fundamentales, diseñadas para almacenar de manera eficiente la información relacionada con los chats, conversaciones, mensajes, modelos y usuarios.

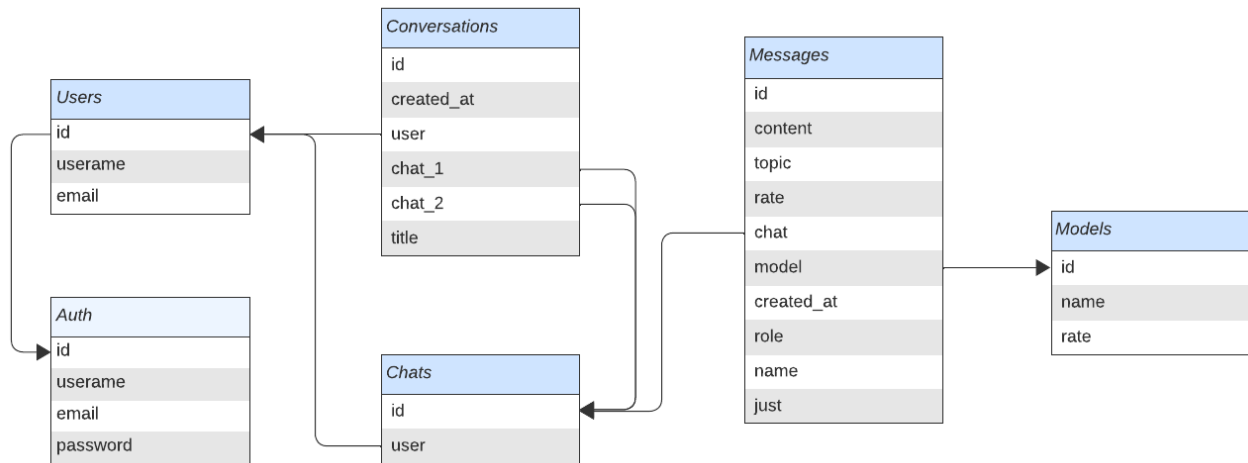


Figura 3.5: Relación y estructura de las tablas de la base de datos

3.5.1. Chats

- **id (uuid - PK³)**: Identificador único para cada chat, generado automáticamente.
- **user (uuid - FK⁴)**: Clave foránea que establece una relación con la tabla de usuarios, identificando al usuario que inició el chat.

3.5.2. Conversations

- **id (uuid - PK)**: Identificador único para cada chat, generado automáticamente.
- **created_at (timestampz)**: Establece la fecha de creación de cada conversación. Esto servirá para organizar las conversaciones según el día de creación.
- **user (uuid - FK)**: Clave foránea que establece una relación con la tabla de usuarios, identificando al usuario que inició el chat.
- **chat_1 & chat_2 (uuid - FK)**: Claves foráneas que establece una relación con la tabla de chats, identificando al chat en concreto que se está usando.
- **title (varchar)**: Título de la conversación. Será una previsualización del primer mensaje que haya mandado el usuario en dicha conversación.

³**PK**: Atributo o grupo de atributos que identifica de manera única a cada registro de la tabla.

⁴**FK**: Es una clave de base de datos que se utiliza para vincular dos tablas de la base de datos.

3.5.3. Messages

- **id (uuid - PK)**: Identificador único para cada mensaje, generado automáticamente.
- **content (text)**: Contenido del mensaje enviado por el usuario.
- **topic (varchar)**: Tema o asunto del mensaje.
- **rate (float4)**: Calificación asignada al mensaje por el sistema.
- **chat (uuid - FK)**: Clave foránea que vincula el mensaje a su chat correspondiente.
- **model (varchar - FK)**: Indica el modelo de IA asociado al mensaje.
- **created_at (timestampz)**: Indica la hora y el día en el que el mensaje fue creado.
- **role (text)**: Indica el rol de quién envía el mensaje, puede ser 'user' en caso de que se trate del usuario o 'assistant' en caso de que sea un modelo.
- **name (text)**: Guarda el nombre del usuario o del modelo en cada caso.
- **just (text)**: Guarda la justificación de la calificación establecida a la respuesta de un modelo que devuelve el que es el encargado de analizarlas.

3.5.4. Models

- **id (uuid - PK)**: Identificador único para cada modelo, generado automáticamente.
- **name (varchar)**: Nombre descriptivo del modelo.
- **rate (float4)**: Calificación media del modelo, calculada a partir de las calificaciones de los mensajes asociados.

3.5.5. Users

- **id (uuid - PK)**: Identificador único para cada usuario registrado.
- **username (text)**: Nombre de usuario elegido por el usuario.
- **email (text)**: Dirección de correo electrónico del usuario.

Es importante comentar que, además de la tabla **users** existe otra tabla **auth** enlazada con la anterior, que contendrá más datos sobre cada uno de los usuarios, como por ejemplo su contraseña encriptada. Sin embargo, esta tabla se encuentra en un schema privado para así ofrecer más seguridad.

3.6. Despliegue de la aplicación

Las plataformas escogidas para realizar el despliegue son Vercel para el frontend y Render para el backend, ya que son plataformas de despliegue en la nube que permite a los desarrolladores implementar aplicaciones web y API de manera sencilla y rápida. Son especialmente conocidas por su soporte para aplicaciones creadas con frameworks modernos como React, Angular, Next.js y otros.

3.6.1. Funcionamiento

- **Integración con Git:** Se integran estrechamente con sistemas de control de versiones como Git, lo que permite desplegar aplicaciones directamente desde los repositorios Git. Esto simplifica el proceso de despliegue y garantiza una integración continua fluida.
- **Infraestructura global:** Cuentan con una red de servidores distribuida globalmente, lo que garantiza una entrega rápida de contenido a los usuarios finales en cualquier parte del mundo.
- **SSL gratuito:** Proporcionan certificados SSL gratuitos para todas las aplicaciones desplegadas en su plataforma garantizando una comunicación segura entre los usuarios y la aplicación.

3.6.2. Ventajas

- **Facilidad de uso:** Ofrecen una interfaz intuitiva y fácil de usar que simplifica el proceso de despliegue de aplicaciones sin necesidad de configuraciones complejas.
- **Rápido tiempo de despliegue:** El tiempo de despliegue de una aplicación es significativamente reducido en comparación con otras plataformas.
- **Escalabilidad sin esfuerzo:** La capacidad de escalar automáticamente en función de la demanda del usuario es una ventaja significativa de Vercel, es decir que los desarrolladores no tienen que preocuparse por la gestión de la infraestructura.

3.7. Resultados

A continuación, se enumeran algunos enlaces a los recursos desarrollados para el prototipo.

- **Repositorio del Frontend:** <https://github.com/RoxanaMB/QuireQuest>
- **Repositorio del Backend:** <https://github.com/RoxanaMB/QQ-Back>
- **Aplicación web:** <https://quire-quest.vercel.app/>



Figura 3.6: QR con acceso a la aplicación web

Capítulo 4

Estudio de viabilidad económica

En esta sección, se llevará a cabo un examen detallado de la viabilidad económica asociada con la implementación de este proyecto que se realizaría en un entorno profesional, haciendo uso de recursos humanos altamente cualificados y con gran cantidad de funcionalidades avanzadas. Además, se elaborará un plan integral de comercialización para promover eficazmente el producto.

4.1. Funcionalidades adicionales

Las funcionalidades adicionales que se incorporarán en la aplicación final abarcan un amplio espectro de características avanzadas, diseñadas para mejorar la experiencia del usuario. Se enumeran algunas de las funcionalidades de valor añadido, si bien el esquema será diseñado de tal modo que se podrán añadir nuevas funcionalidades en el futuro de forma cómoda:

- **Chat con multimedia y documentos PDF:** Los usuarios podrán adjuntar imágenes, vídeos, audios y/o documentos PDF durante las conversaciones en el chat, permitiendo consultas específicas sobre el contenido y posibilitando que el chat responda en base al contenido adjunto.
- **Organizador de tareas diarias:** La aplicación permitirá a los usuarios gestionar y organizar sus tareas diarias de manera eficiente. Incluirá características como la creación de listas de tareas, recordatorios, prioridades, y seguimiento del progreso. También ofrecerá opciones de sincronización con calendarios y la posibilidad de recibir notificaciones para asegurar el cumplimiento de las tareas.
- **Generar contenido multimedia a partir de texto:** La aplicación utilizará avanzados algoritmos de inteligencia artificial para transformar texto en archivos multimedia realistas y naturales. Los usuarios podrán introducir descripciones o guiones, y la IA generará imágenes, vídeos, audios, e incluso animaciones basadas en la información proporcionada. Esta funcionalidad será ideal para la creación de presentaciones, contenido educativo, marketing visual, y otros propósitos creativos, permitiendo a los usuarios producir material multimedia de alta calidad con facilidad y precisión.
- **Transcripción de multimedia a texto:** Los usuarios podrán convertir contenido multimedia como audios y vídeos a texto escrito. Esta funcionalidad será útil para la creación de subtítulos, la transcripción de entrevistas o reuniones, y la conversión

de notas de voz a texto, facilitando la accesibilidad y el análisis de información contenida en formatos audiovisuales.

- **Análisis de código fuente:** Los usuarios podrán adjuntar archivos que contengan código fuente o solicitar la creación de código específico. La aplicación posteriormente ofrecerá la capacidad de analizar todos los códigos generados, proporcionando información sobre su optimización, posibles errores de compilación, velocidad de ejecución, entre otros aspectos.
- **Generador de rutinas, recetas, etc:** La aplicación ayudará a los usuarios a crear rutinas personalizadas, recetas de cocina, planes de ejercicio, y otros tipos de guías prácticas. Utilizará información proporcionada por los usuarios, así como datos contextuales y de preferencias, para generar contenido relevante y útil que se adapte a sus necesidades y objetivos específicos.
- **Organizador de viajes:** Esta herramienta permitirá a los usuarios planificar y organizar sus viajes de manera integral. Incluirá funciones para buscar y reservar vuelos, hoteles, y actividades, así como la creación de itinerarios personalizados. También ofrecerá recomendaciones basadas en el destino, alertas de vuelos y clima, y la posibilidad de compartir planes de viaje con otros participantes.
- **Variedad de idiomas:** La aplicación será multilingüe, ofreciendo una variedad de opciones de idiomas para adaptarse a la diversidad de usuarios. Esto permitirá una experiencia más inclusiva y accesible a nivel global.

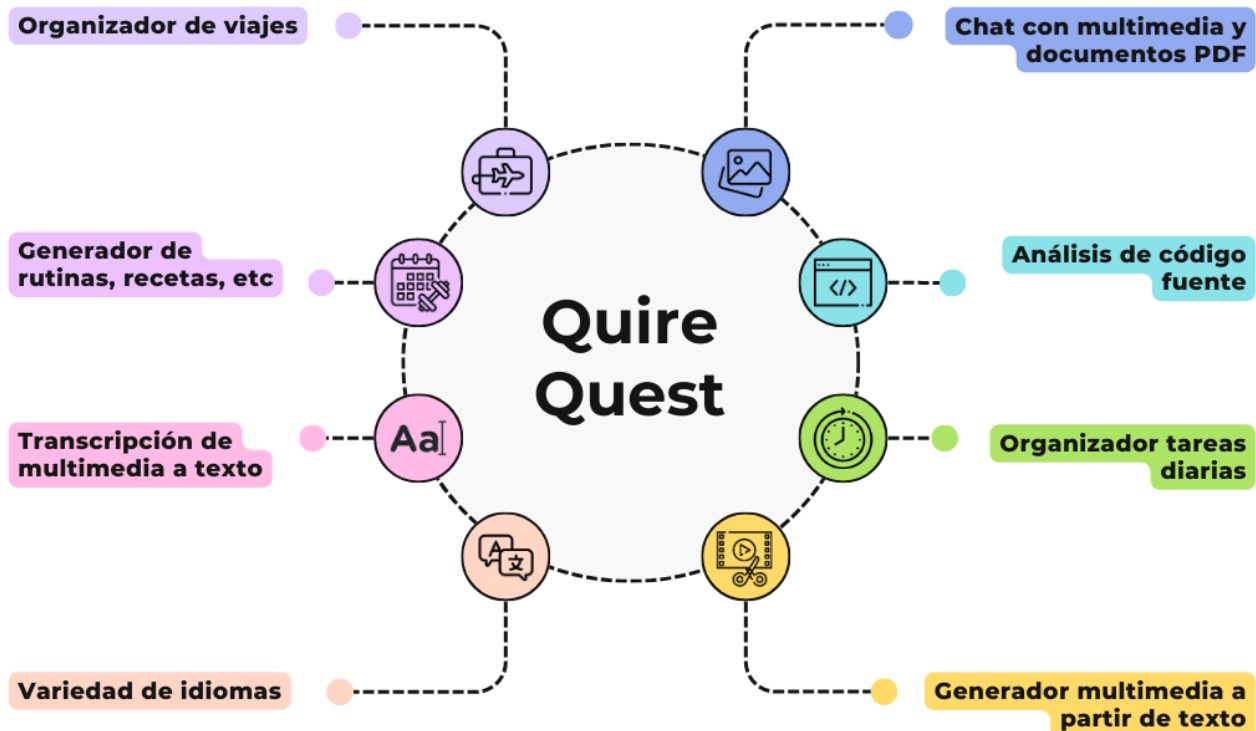


Figura 4.1: Funcionalidades de la aplicación profesional

Estas funcionalidades avanzadas buscan no solo enriquecer la interacción del usuario con la aplicación, sino también proporcionar herramientas potentes que abarquen diversas formas de comunicación y creación de contenido.

Por otro lado, comentar que la implementación de la IA en estas funciones permite una mayor personalización y adaptabilidad a las preferencias de los usuarios. Además, la aplicación se beneficiará de los avances continuos en el campo de la IA, lo que garantizará que las capacidades de generación de contenido evolucionen con el tiempo.

4.2. Desarrollo del proyecto

El desarrollo del proyecto se llevará a cabo en seis fases bien definidas, cada una con objetivos específicos para garantizar un proceso eficiente y de alta calidad. A continuación, se detalla en profundidad cada una de las fases:

- **Análisis (27 días):** La fase inicial del proyecto es crucial, ya que implica comprender a fondo el problema a resolver y establecer las bases del software a desarrollar. Se detallan los requisitos y funcionalidades en un documento, lo que permite fijar plazos y costos. Es esencial cerrar este documento para evitar la inclusión de nuevos requisitos durante el desarrollo. La comunicación en esta etapa, a través de reuniones con el equipo, asegura avances adecuados en el análisis y alinea las expectativas del proyecto. La claridad en la definición de requisitos es clave para un desarrollo exitoso.
- **Diseño (24 días):** En la segunda fase, se exploran las posibles implementaciones del software y se establece su arquitectura general en respuesta a los requisitos definidos previamente. Esta etapa implica diseñar una solución que cumpla con los requisitos, examinando y eligiendo fuentes de datos y tecnologías que se alineen de manera óptima con el proyecto. Una vez completada esta selección, se desarrollan las arquitecturas que constituirán el producto final. Además, se aborda la estructura inicial de la base de datos y se define el diseño de las interfaces de usuario tanto para la aplicación web como para la móvil. En esencia, esta fase sienta las bases técnicas y estructurales necesarias para la implementación exitosa del proyecto.
- **Desarrollo e implementación (135 días):** Esta fase es la etapa central del proyecto, donde se llevará a cabo la ejecución de tareas para construir la aplicación, siguiendo los objetivos establecidos en el proyecto, es decir el desarrollo del front-end, backend, base de datos y la integración de todas las partes. Se seguirán las mejores prácticas de desarrollo de software para garantizar la eficiencia y la calidad del código. Además, se implementarán las funcionalidades planificadas en las fases anteriores. Esta etapa incluirá la creación de la aplicación web y móvil, asegurando su compatibilidad con diversas plataformas y dispositivos.
- **Pruebas (130 días):** Durante el desarrollo, se llevarán a cabo pruebas para garantizar la estabilidad y la funcionalidad del sistema. Esto incluirá pruebas de unidad, pruebas de integración y pruebas de usuario. Se identificarán y corregirán posibles errores y se asegurará que la aplicación cumpla con los estándares de calidad establecidos.

- **Despliegue (10 días):** En esta fase, la aplicación estará lista para ser lanzada al público. Se realizan los preparativos necesarios para la implementación en entornos de producción. Esto incluirá la configuración de servidores, la gestión de la base de datos en producción y la configuración de servicios en la nube si es necesario. El despliegue se llevará a cabo de manera planificada para minimizar cualquier impacto en la disponibilidad del sistema.
- **Mantenimiento:** Después de la implementación, se establecerá un período de mantenimiento continuo. Esto implica monitorear el rendimiento de la aplicación, solucionar problemas emergentes, aplicar actualizaciones de seguridad y realizar mejoras continuas. El mantenimiento asegura que la aplicación siga siendo eficiente y cumpla con las expectativas a lo largo del tiempo. Además, se podrán incorporar nuevas funcionalidades y mejoras en función de las necesidades del usuario y del mercado. Este período de mantenimiento será una práctica continua para garantizar la sostenibilidad y la evolución constante de la aplicación.




















		Costo	Nombre	Duración	Inicio	Terminado
1		114050,00 €	<input type="checkbox"/> BookPlanning	191 days?	30/01/24 8:00	22/10/24 17:00
2		8280,00 €	<input type="checkbox"/> Análisis	27 days?	30/01/24 8:00	6/03/24 17:00
3		3200,00 €	Análisis de requisitos	10 days?	30/01/24 8:00	12/02/24 17:00
4		3200,00 €	Análisis de funcionalidades	10 days?	13/02/24 8:00	26/02/24 17:00
5		1480,00 €	Análisis de costes	5 days?	27/02/24 8:00	4/03/24 17:00
6		400,00 €	Acta de Constitución del proyecto	2 days?	5/03/24 8:00	6/03/24 17:00
7		13870,00 €	<input type="checkbox"/> Diseño	24 days?	7/03/24 8:00	9/04/24 17:00
8		2050,00 €	Elección de tecnologías	5 days?	7/03/24 8:00	13/03/24 17:00
9		1750,00 €	Diseño de la arquitectura	7 days?	14/03/24 8:00	22/03/24 17:00
10		2150,00 €	Diseño de la base de datos	7 days?	25/03/24 8:00	2/04/24 17:00
11		7920,00 €	Diseño de interfaces de usuario (web y móvil)	24 days?	7/03/24 8:00	9/04/24 17:00
12		65200,00 €	<input type="checkbox"/> Desarrollo	135 days?	3/04/24 8:00	8/10/24 17:00
13		1600,00 €	<input type="checkbox"/> Implementación de la base de datos	5 days?	3/04/24 8:00	9/04/24 17:00
14		1600,00 €	Configuración de la base de datos	5 days?	3/04/24 8:00	9/04/24 17:00
15		17200,00 €	<input type="checkbox"/> Desarrollo del backend	85 days?	10/04/24 11:00	7/08/24 11:00
16		2000,00 €	Crear modelos de la base de datos	15 days?	10/04/24 11:00	1/05/24 11:00
17		2000,00 €	Implementar la librería con las IA	20 days?	1/05/24 11:00	29/05/24 11:00
18		5600,00 €	Implementar la API REST	50 days?	29/05/24 11:00	7/08/24 11:00
19		7600,00 €	Realizar tests	70 days?	1/05/24 11:00	7/08/24 11:00
20		20800,00 €	<input type="checkbox"/> Desarrollo del frontend web	90 days?	10/04/24 8:00	13/08/24 17:00
21		10400,00 €	Implementar pantallas para los distintos módulos	90 days?	10/04/24 8:00	13/08/24 17:00
22		10400,00 €	Realizar tests	90 days?	10/04/24 8:00	13/08/24 17:00
23		25600,00 €	<input type="checkbox"/> Desarrollo del frontend móvil	40 days?	14/08/24 8:00	8/10/24 17:00
24		12800,00 €	Implementar pantallas para los distintos módulos	40 days?	14/08/24 8:00	8/10/24 17:00
25		12800,00 €	Realizar tests	40 days?	14/08/24 8:00	8/10/24 17:00
26		15600,00 €	<input type="checkbox"/> Pruebas	130 days?	10/04/24 8:00	8/10/24 17:00
27		10800,00 €	Testeo de componentes y módulos de la web	90 days?	10/04/24 8:00	13/08/24 17:00
28		4800,00 €	Testeo de componentes y módulos de la app móvil	40 days?	14/08/24 8:00	8/10/24 17:00
29		2800,00 €	<input type="checkbox"/> Despliegue	10 days?	9/10/24 8:00	22/10/24 17:00
30		1600,00 €	Desplegar la app en los entornos seleccionados	10 days?	9/10/24 8:00	22/10/24 17:00
31		1200,00 €	Comprobar el funcionamiento	10 days?	9/10/24 8:00	22/10/24 17:00

Figura 4.2: Tareas definidas para el desarrollo del proyecto

La tareas que se han definido para la realización del proyecto se pueden visualizar según su orden de ejecución en el siguiente **diagrama de Gantt**.

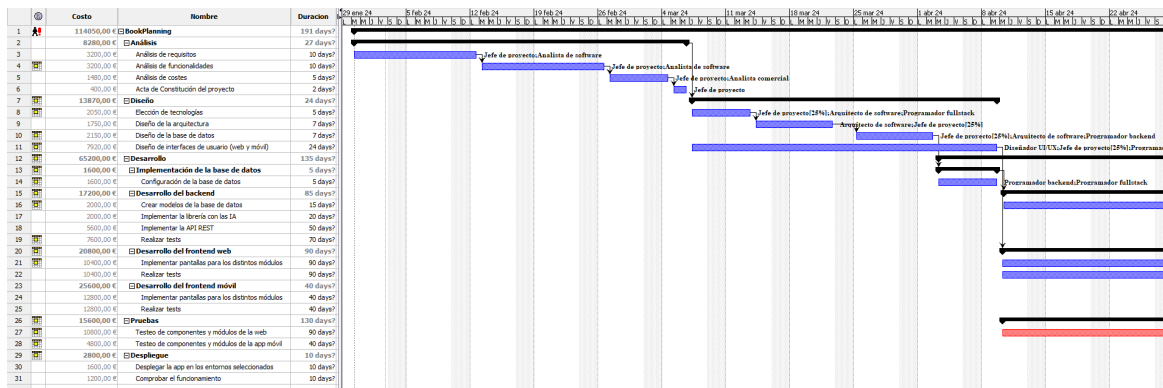


Figura 4.3: Diagrama de Gantt (1)

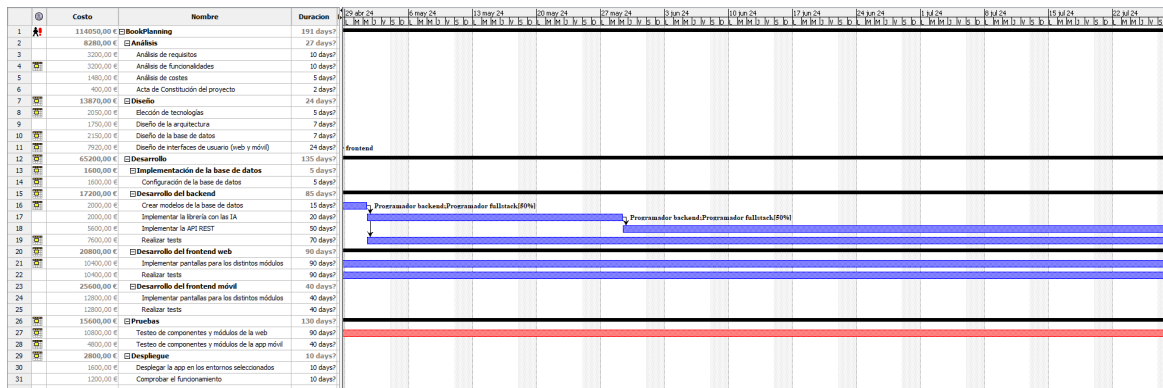


Figura 4.4: Diagrama de Gantt (2)

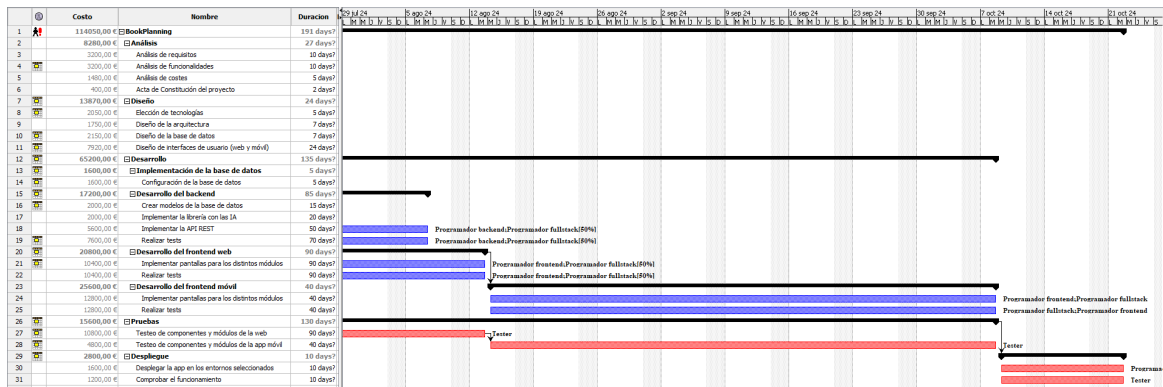


Figura 4.5: Diagrama de Gantt (3)

4.3. Coste y duración del proyecto

La ejecución del proyecto se llevará a cabo mediante un equipo que ha sido meticulosamente conformado, integrando diversas habilidades y roles esenciales. Estos perfiles son fundamentales para garantizar el éxito y la coherencia del proyecto. A continuación, se presenta la estructura del equipo:

- **Jefe de proyecto:** Encargado de la supervisión general del proyecto, coordinando todas las fases y asegurando la alineación con los objetivos estratégicos.
- **Analista de Software:** Responsable de analizar los requisitos del proyecto y traducirlos en especificaciones técnicas detalladas, estableciendo las bases para el desarrollo.
- **Analista comercial:** Responsable de analizar los posibles costos que supondría crear la aplicación e introducirla al mercado.
- **Arquitecto de Software:** Encargado de diseñar la arquitectura general del software, definiendo la estructura técnica y asegurando una base sólida para la implementación.
- **Diseñador UI/UX (Gráfico):** Responsable de crear interfaces de usuario atractivas y funcionales, asegurando una experiencia de usuario intuitiva y agradable.
- **Desarrolladores (Programadores):** Encargados de llevar a cabo la implementación del software, trabajando en el desarrollo tanto del backend como del frontend según las especificaciones definidas.
- **Tester:** Realiza pruebas exhaustivas para identificar y corregir posibles errores, garantizando la calidad del software en todas las etapas del desarrollo.

Este equipo, multidisciplinario, aborda cada aspecto del proyecto, desde la concepción hasta la implementación, asegurando un enfoque integral y colaborativo para alcanzar los objetivos establecidos. Sin embargo, además del equipo humano, para el desarrollo de la aplicación, es esencial contar con los siguientes materiales:

- **Portátiles:** Equipos esenciales para la ejecución del desarrollo de software, incluyendo estaciones de trabajo con la capacidad necesaria para la codificación y la gestión de proyectos.
- **Móvil Android**¹: Un dispositivo Android para probar y verificar la funcionalidad de la aplicación en el sistema operativo Android.
- **Móvil iOS**²: Un dispositivo iOS para realizar pruebas y asegurar la compatibilidad de la aplicación con el sistema operativo iOS.

Estos materiales constituyen la infraestructura básica necesaria para llevar a cabo el desarrollo de la aplicación de manera efectiva y garantizar la compatibilidad en diversas plataformas y dispositivos. La elección y calidad de estos materiales contribuirán significativamente al rendimiento y éxito del equipo de desarrollo.

¹**Android:** Sistema operativo móvil basado en el núcleo Linux y otros software de código abierto.

²**iOS:** Sistema operativo móvil de código cerrado desarrollado por Apple Inc. Originalmente desarrollado para el iPhone, después se utilizó en dispositivos como el iPod touch y el iPad


















		Nombre	Tipo	Tasa Estandar
1		Jefe de proyecto	Trabajo	25,00 €/hora
2		Analista de software	Trabajo	15,00 €/hora
3		Analista comercial	Trabajo	12,00 €/hora
4		Arquitecto de software	Trabajo	25,00 €/hora
5		Diseñador UI/UX	Trabajo	15,00 €/hora
6		Programador fullstack	Trabajo	20,00 €/hora
7		Programador frontend	Trabajo	20,00 €/hora
8		Programador backend	Trabajo	20,00 €/hora
9		Tester	Trabajo	15,00 €/hora
10		Portátil	Material	1800,00 €
11		Portátil	Material	1800,00 €
12		Portátil	Material	1800,00 €
13		Portátil	Material	1000,00 €
14		Portátil	Material	1000,00 €
15		Móvil Android	Material	250,00 €
16		Móvil iOS	Material	600,00 €

Figura 4.6: Recursos necesarios para llevar a cabo la aplicación profesional

Luego de identificar todas las tareas esenciales y determinar sus respectivas duraciones, se estima que el proyecto se completará en un período de **191 días hábiles**. Este plazo abarca desde el **30 de enero de 2024** hasta el **22 de octubre de 2024**, considerando una jornada laboral estándar de 8 horas al día, de lunes a viernes. En términos financieros, el costo total para la ejecución exitosa del proyecto se cifra en **114.050 euros**, sin embargo, hay que tener en cuenta los sueldos de los empleados todos aquellos meses que se tarde en conseguir beneficios por lo que se pedirá una inversión inicial de **300.000 euros**.

4.4. Modelo de comercialización

El modelo de comercialización constituye la estrategia mediante la cual se generarán ingresos a través de la utilización de la aplicación. En este contexto, se presentan tres alternativas distintas:

Plan de Suscripciones

Se propone un sistema de suscripciones que brinda a los usuarios diversas opciones en función de sus necesidades y preferencias. Este plan incluye:

- Gratuito
- Básico (13.95 euros mensuales / 149 euros anuales)
- Pro (24.95 euros mensuales / 249 euros anuales)
- Business (34.95 euros mensuales / 349 euros anuales)

Características	Gratis	Básico	Pro	Business
Análisis de código fuente	SI	SI	SI	SI
Variedad de idiomas	SI	SI	SI	SI
Consultas a varios modelos a la vez	SI	SI	SI	SI
Organizador de tareas diarias	NO	SI	SI	SI
Generador de rutinas, recetas, etc	NO	SI	SI	SI
Chat con multimedia y documentos PDF	NO	NO	SI	SI
Generar multimedia a partir de texto	NO	NO	SI	SI
Transcripción de multimedida a texto	NO	NO	SI	SI
Organizador de viajes	NO	NO	SI	SI
Acceso a todos los modelos de lenguaje	NO	NO	SI	SI
Mensajes ilimitados	NO	NO	NO	SI
Personalización avanzada	NO	NO	NO	SI
Soporte exclusivo 24 horas	NO	NO	NO	SI
Actualizaciones prioritarias	NO	NO	NO	SI
Integración con herramientas externas	NO	NO	NO	SI

Tabla 4.1: Características de cada uno de los planes de suscripción

Cada opción de suscripción ofrece un valor escalonado, permitiendo a los usuarios elegir el nivel que mejor se adapte a sus necesidades y recursos financieros. Este enfoque de suscripción no solo diversifica las fuentes de ingresos, sino que también ofrece flexibilidad a los usuarios para seleccionar el nivel de servicio que mejor se alinee con sus expectativas y requerimientos.

4.5. Estimación del retorno de la inversión [63]

Después de haber completado el desarrollo y lanzamiento de la aplicación al mercado, es crucial evaluar cuándo se alcanzará el Retorno de la Inversión (ROI). Este cálculo implica la consideración de los costos asociados al desarrollo, así como los gastos de mantenimiento recurrentes, ya sea mensuales o anuales, en comparación con los ingresos generados por el uso de la aplicación. La meta es estimar cuándo se recuperará la inversión inicial, generalmente proyectando una semana específica para este logro.

En este análisis, se parte de la suposición de que la adopción de usuarios seguirá una trayectoria exponencial. Se han realizado diversos cálculos, iniciando con la proyección del crecimiento de usuarios. Este enfoque permitirá determinar el momento aproximado en el cual los ingresos generados superarán los costos, llevando al punto de equilibrio y alcanzando el retorno de la inversión.

Usuarios

Como se ha mencionado con anterioridad, la aplicación se segmenta en cuatro tipos de usuarios: gratuitos, básicos, pro y business. Cabe destacar que en las categorías de pago, se ofrecen cuentas con opciones de renovación mensual y anual. La premisa inicial del análisis sugiere que la distribución de usuarios seguirá una tendencia descendente, siendo la opción gratuita la preferida por la mayoría en sus primeras interacciones con la plataforma.

Usuarios potenciales (acumulados)	Promedio de uso	Promedio de pago
4000	1200	39
8400	2520	83
13230	3969	130
18522	5556,6	183
24310	7293	240
30630	9189	303
37522	11256,6	371
45027	13508,1	445
53188	15956,4	526
62053	18615,9	614
71671	21501,3	709
82096	24628,8	812
93384	28015,2	924
105596	31678,8	1045
118795	35638,5	1176
133051	39915,3	1317
148435	44530,5	1469
165025	49507,5	1633
182903	54870,9	1810
202156	60646,8	2001
222877	66863,1	2206
245164	73549,2	2427
269123	80736,9	2664
294866	88459,8	2919

Figura 4.7: Usuarios potenciales y los que llegarían a usarla y/o pagarla

Dentro de esta lógica, se ha llevado a cabo una simulación que modela un crecimiento gradual de usuarios, considerando que la gran mayoría optará por iniciar su experiencia con el plan gratuito. Con el objetivo de captar la atención y participación de un número significativo de usuarios, se ha propuesto una inversión mensual de **2.000 euros** en anuncios proporcionados por Meta, asegurando así que, en promedio, **4.000 usuarios** tengan la oportunidad de visualizar el anuncio de nuestra aplicación.

De este conjunto de 4.000 usuarios, se estima que aproximadamente el 30 % decidirá utilizar la aplicación en su plan gratuito, siendo este un punto crucial en el embudo de conversión. Dentro de este grupo, solo el 5 % se espera que realice la transición hacia alguno de los planes de pago disponibles. Esta estrategia se basa en la premisa de que, al ofrecer una experiencia inicial gratuita robusta y atractiva, podemos motivar a un porcentaje significativo de usuarios a explorar y aprovechar las funcionalidades premium ofrecidas.

En este contexto, la inversión publicitaria se presenta como una herramienta crucial para maximizar la visibilidad y el conocimiento de la aplicación, generando así una base sólida de usuarios que podrían potencialmente convertirse en suscriptores de los planes de pago a medida que descubren y valoran las funcionalidades adicionales proporcionadas por estas ofertas.

Ingresos por suscripciones

En relación a los ingresos provenientes de las suscripciones, se han fijado los precios según la duración de dichas suscripciones y la cantidad de funcionalidades de las que se desee disponer. Conforme mencionado anteriormente, los costos varían entre los 13.95 euros a los 34.95 euros para la suscripción mensual y entre los 149 euros y los 349 euros en suscripciones anuales. Utilizando la cantidad acumulada de usuarios de cada tipo, se puede proceder a calcular los beneficios mensuales.

Este enfoque permite proyectar los ingresos derivados de las suscripciones a lo largo del tiempo, considerando la variabilidad en la duración de las mismas y la distribución de usuarios en cada categoría.

Para hacer los cálculos más realistas, según el estudio realizado por YouGov [64] El 30 % de los usuarios a nivel mundial prefieren pagos mensuales, a un 27 % les es indiferente y otro 11 % no saben. Para poder hacer un cálculo aproximado, supongamos que estos dos grupos de usuarios también pagarán la cuota mensual es decir que un 68 % de los usuarios que se decidieron por pagar una suscripción escogerán una mensual y el 32 % restante por una anual.

Promedio Básico	Mensual	Anual	Promedio Pro	Mensual	Anual	Promedio Premium	Mensual	Anual	Ganancias Mensuales
11	7	4	19	13	6	7	5	2	3.413,22 €
24	16	8	41	28	13	16	11	5	7.501,61 €
39	27	12	65	44	21	26	18	8	12.033,06 €
54	37	17	91	62	29	36	24	12	16.757,81 €
72	49	23	120	82	38	48	33	15	22.214,88 €
90	61	29	151	103	48	60	41	19	27.865,25 €
111	75	36	185	126	59	74	50	24	34.247,94 €
133	90	43	222	151	71	89	61	28	41.113,18 €
157	107	50	263	179	84	105	71	34	48.614,79 €
184	125	59	307	209	98	122	83	39	56.713,28 €
212	144	68	354	241	113	141	96	45	65.429,76 €
243	165	78	406	276	130	162	110	52	75.071,87 €
277	188	89	462	314	148	184	125	59	85.407,50 €
313	213	100	522	355	167	209	142	67	96.650,38 €
352	239	113	588	400	188	235	160	75	108.780,09 €
395	269	126	658	447	211	263	179	84	121.795,94 €
440	299	141	734	499	235	293	199	94	135.776,88 €
489	333	156	816	555	261	326	222	104	150.972,71 €
543	369	174	905	615	290	362	246	116	167.537,22 €
600	408	192	1000	680	320	400	272	128	185.124,00 €
661	449	212	1103	750	353	441	300	141	204.118,95 €
728	495	233	1213	825	388	485	330	155	224.539,76 €
799	543	256	1332	906	426	532	362	170	246.465,38 €
875	595	280	1459	992	467	583	396	187	269.991,78 €

Figura 4.8: Promedio de usuarios en cada plan y las ganancias mensuales

Gastos de la aplicación

En la sección de gastos, se incluye la tarifa anual de desarrollador de Apple de 99 euros y los 25 euros correspondientes al pago único de desarrollador de Google. Estas tarifas son esenciales para poder publicar la aplicación en las tiendas de aplicaciones y aprovechar otras utilidades proporcionadas por estas plataformas.

Además, se contemplan las cuotas de Publicidad de Meta, para poder dar a conocer la aplicación y que llegue al máximo de usuarios posibles y así asegurar un crecimiento más rápido.

Por otro lado, también hay que tener en cuenta los distintos planes de suscripción de las plataformas de IA. Considerando la generación de contenido multimedia mediante Inteligencia Artificial, se requerirán planes de suscripción que proporcionen acceso a los servicios de IA necesarios para la generación de audio, imágenes, video, código. texto, etc. También será importante tener en cuenta las nóminas que se deberán seguir pagando a los empleados que en un principio se contrató para llevar a cabo la aplicación.

En el análisis de gastos, también se incorporan las tarifas de DigitalOcean, que abarcan el despliegue de la web, la API, la base de datos, entre otros servicios. Se añade un recargo variable según el número de usuarios activos, ya que a medida que aumentan las transacciones por usuario o la demanda de recursos de servicios específicos, se refleja en un incremento de costos.

Gastos acumulados - Anuales		Gastos acumulados - Mensuales	
iOS	99,00 €	Nóminas (los trabajadores del desarrollo)	26.752,00 €
Seguridad	50,00 €	Anuncios	2.000,00 €
Dominio	10,00 €	Mantenimiento	350,00 €
Gastos mensuales cada año	349.224,00 €		
Servidor (primer año 84 euros)	1.008,00 €	Gastos Desarrollador Android	25,00 €
Servidor (resto años 168 euros)	2.016,00 €		

Figura 4.9: Gastos a tener en cuenta

Ingresos frente a gastos

Esta sección es crucial para calcular el Retorno de la Inversión. En este proceso, se restan los gastos acumulados, que incluyen la inversión inicial, de los ingresos que se van generando a lo largo del tiempo. Este análisis proporcionará una visión clara de cuándo la aplicación comenzará a generar beneficios netos en relación con la inversión realizada.

Si se logra recuperar la inversión inicial, en algún momento esta operación arrojará un resultado positivo, señalando que a partir de ese punto se ha igualado la inversión y se comenzará a obtener beneficios netos. En este caso, tras **16 meses** desde que se publica la aplicación.

Ganancias Acumuladas	Gastos Acumulados	ROI	Mes
1.046,15 €	329.345,00 €	-99,68%	1
8.547,76 €	358.531,00 €	-97,62%	2
20.580,82 €	387.717,00 €	-94,69%	3
37.338,62 €	416.903,00 €	-91,04%	4
59.553,50 €	446.089,00 €	-86,65%	5
87.418,75 €	475.275,00 €	-81,61%	6
121.666,69 €	504.461,00 €	-75,88%	7
162.779,87 €	533.647,00 €	-69,50%	8
211.394,66 €	562.833,00 €	-62,44%	9
268.107,94 €	592.019,00 €	-54,71%	10
333.537,70 €	621.205,00 €	-46,31%	11
408.609,57 €	650.391,00 €	-37,17%	12
494.017,07 €	679.820,00 €	-27,33%	13
590.667,45 €	709.090,00 €	-16,70%	14
699.447,54 €	738.360,00 €	-5,27%	15
821.243,48 €	767.630,00 €	6,98%	16
957.020,36 €	796.900,00 €	20,09%	17
1.107.993,06 €	826.170,00 €	34,11%	18
1.275.530,28 €	855.440,00 €	49,11%	19
1.460.654,28 €	884.710,00 €	65,10%	20
1.664.773,23 €	913.980,00 €	82,15%	21
1.889.312,99 €	943.250,00 €	100,30%	22
2.135.778,37 €	972.520,00 €	119,61%	23
2.405.770,15 €	1.001.790,00 €	140,15%	24

Figura 4.10: Evolución de los gastos frente a los ingresos

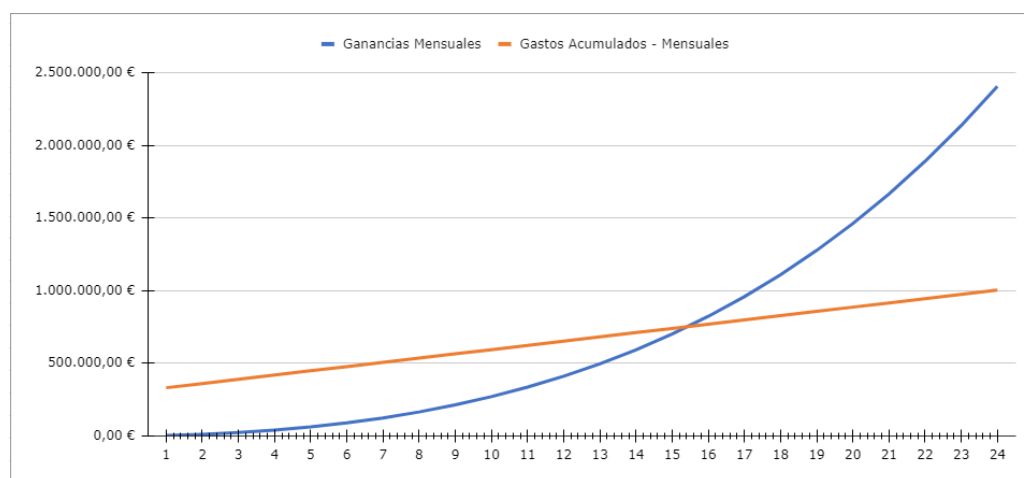


Figura 4.11: Punto en el que se empieza a tener beneficios

Capítulo 5

Conclusiones y líneas futuras

El proyecto representa una respuesta a la creciente relevancia de la inteligencia artificial (IA) en el entorno tecnológico actual. En un mundo donde la IA está en constante evolución y se integra en diversas áreas de nuestra vida, una aplicación que facilite la interacción con diferentes modelos de IA se vuelve indispensable. Esta aplicación ofrece a los usuarios la posibilidad de acceder a una variedad de perspectivas sobre diversos temas, obtener respuestas precisas y especializadas a sus consultas, y acceder eficientemente a información especializada.

Para llevar a cabo este proyecto, se han seleccionado tecnologías como Angular, Flask, Supabase, Vercel y Render. Estas tecnologías proporcionan una base sólida tanto en el frontend como en el backend de la aplicación, asegurando una experiencia fluida y segura para los usuarios. Angular ofrece una interfaz de usuario robusta y receptiva, mientras que Flask simplifica el desarrollo del backend con Python. Supabase garantiza una gestión segura y eficiente de la base de datos mientras que Vercel y Render facilitan el despliegue rápido y automatizado en la nube.

La integración de servicios de IA en una sola plataforma proporciona una solución eficiente para los usuarios, eliminando la necesidad de recurrir a múltiples plataformas para acceder a herramientas como generación de imágenes, videos, transcripción de video a texto y conversación con documentos e imágenes. Esto simplifica y optimiza el acceso a estas herramientas, permitiendo a los usuarios aprovechar al máximo las capacidades de la IA de manera integrada y sin complicaciones.

Desde el punto de vista económico, el proyecto presenta una perspectiva prometedora. Se espera que la inversión inicial en su desarrollo profesional genere beneficios significativos a largo plazo, con un aumento exponencial de usuarios y beneficios a partir de la semana 16 del lanzamiento. Además, ofrece una amplia gama de oportunidades de expansión futura, como la integración de nuevos modelos de IA, la personalización de perfiles de usuario y la colaboración con empresas interesadas en incorporar capacidades de IA en sus productos o servicios.

En resumen, el proyecto representa una solución innovadora y rentable para aprovechar al máximo el potencial de la IA, tanto para usuarios como para inversores. Con un enfoque en la mejora continua y la adaptación a las necesidades del mercado, se espera que el proyecto siga creciendo y evolucionando en el futuro, ofreciendo nuevas funcionalidades y beneficios para todos sus usuarios.

Capítulo 6

Summary and Conclusions

The project represents a response to the growing relevance of artificial intelligence (AI) in the current technological landscape. In a world where AI is constantly evolving and integrating into various aspects of our lives, an application that facilitates interaction with different AI models becomes indispensable. This application offers users the ability to access a variety of perspectives on various topics, obtain precise and specialized answers to their queries, and efficiently access specialized information.

To carry out this project, technologies such as Angular, Flask, Supabase, Vercel and Render have been selected. These technologies provide a solid foundation both in the frontend and backend of the application, ensuring a smooth and secure experience for users. Angular offers a robust and responsive user interface, while Flask simplifies backend development with Python. Supabase ensures secure and efficient database management, while Vercel and Render facilitate rapid and automated deployment in the cloud.

The integration of AI services into a single platform provides an efficient solution for users, eliminating the need to resort to multiple platforms to access tools such as image and video generation, video-to-text transcription, and conversation with documents and images. This simplifies and optimizes access to these tools, allowing users to make the most of AI capabilities in an integrated and uncomplicated manner.

From an economic standpoint, the project presents promising prospects. It is expected that the initial investment in its professional development will generate significant long-term benefits, with an exponential increase in users and profits from week 16 onwards. Additionally, it offers a wide range of future expansion opportunities, such as the integration of new AI models, user profile customization, and collaboration with companies interested in incorporating AI capabilities into their products or services.

In summary, the project represents an innovative and cost-effective solution to harness the full potential of AI, both for users and investors. With a focus on continuous improvement and adaptation to market needs, it is expected that the project will continue to grow and evolve in the future, offering new functionalities and benefits for all its users.

Capítulo 7

Presupuesto

Además de analizar la viabilidad económica del proyecto, también se ha llevado a cabo una evaluación de los costos asociados con la ejecución de esta aplicación. La estimación del tiempo dedicado se basa en la duración total del proyecto, que se establece en 300 horas de acuerdo con las pautas de la Resolución del 21 de marzo de 2011 de la Universidad de La Laguna.

Según dicha resolución, un Trabajo de Fin de Grado equivale a 12 créditos ECTS, y dado que cada crédito implica un total de 25 horas, se concluye que la duración total del proyecto abarca 300 horas.

Tareas	Horas dedicadas	Coste
Análisis del problema y tendencia del mercado	20 horas	16 €/h
Estudio y selección de tecnologías	40 horas	16 €/h
Implementación del Backend	75 horas	20 €/h
Implementación del Frontend web y móvil	100 horas	20 €/h
Documentación del proyecto	65 horas	16 €/h
TOTAL	300 horas	5.500 €

Tabla 7.1: Presupuesto del proyecto de Trabajo de Fin de Grado

Por lo tanto, para realizar este proyecto hicieron falta **300 horas** con un coste de **5.500€**.

Bibliografía

- [1] «ChatGPT.» dirección: <https://chat.openai.com/>
- [2] «Grammarly.» dirección: <https://www.grammarly.com/>
- [3] «Murf.» dirección: <https://murf.ai/>
- [4] «Jaspe.» dirección: <https://www.jasper.ai/?fpr=savenow>
- [5] «Syntesia.» dirección: <https://www.synthesia.io/>
- [6] «Trint.» dirección: <https://trint.com/>
- [7] «Otter.ai.» dirección: <https://otter.ai/>
- [8] «Speechify.» dirección: <https://speechify.com/es/>
- [9] «Hooper.» dirección: <https://hopper.com/es>
- [10] McKinsey & Company (2022, 6 diciembre). 'El estado de la IA en 2022 y el balance de media década'. <https://www.mckinsey.com/featured-insights/destacados/el-estado-de-la-ia-en-2022-y-el-balance-de-media-decada/es>
- [11] Grand View Research (2024, 8 febrero). 'Generative AI Market Size, Share & Trends Analysis Report by component (Software, Services), by technology, by end-use, by application, by model, by region, and segment forecasts, 2024 - 2030'. <https://www.grandviewresearch.com/industry-analysis/generative-ai-market-report>
- [12] Grand View Research (2023, 8 marzo). 'Chatbot Market Size, share & Trends, Analysis Report by application (Customer Services, Branding & Advertising), by type, by Vertical, by Region (North America, Europe, Asia Pacific, South America), and segment Forecasts, 2023 - 2030.'. <https://www.grandviewresearch.com/industry-analysis/chatbot-market>
- [13] Conway M. & Egginton E. (2021, 30 noviembre). 'IBM Acquires Hybrid Cloud Consulting Business in Australia'. *IBM Newsroom*. <https://newsroom.ibm.com/2021-11-30-IBM-Acquires-Hybrid-Cloud-Consulting-Business-in-Australia>
- [14] «OpenAI» dirección: <https://openai.com/>
- [15] «HTML» dirección: <https://developer.mozilla.org/es/docs/Web/HTML>
- [16] «CSS» dirección: <https://developer.mozilla.org/es/docs/Web/CSS>
- [17] «JavaScript» dirección: <https://developer.mozilla.org/es/docs/Web/JavaScript>

- [18] «Tailwind CSS» dirección: <https://tailwindcss.com/>
- [19] «Bootstrap» dirección: <https://getbootstrap.com/>
- [20] «Materialize» dirección: <https://materializecss.com/>
- [21] «React» dirección: <https://es.react.dev/>
- [22] «Ionic» dirección: <https://ionicframework.com/>
- [23] «Angular» dirección: <https://angular.io/>
- [24] «Vue.js» dirección: <https://vuejs.org/>
- [25] «Python» dirección: <https://www.python.org/>
- [26] «Java» dirección: <https://www.java.com/es/>
- [27] «Ruby» dirección: <https://www.ruby-lang.org/es/>
- [28] «PHP» dirección: <https://www.php.net/>
- [29] «Go» dirección: <https://go.dev/>
- [30] «Express.js» dirección: <https://expressjs.com/>
- [31] «Django» dirección: <https://www.djangoproject.com/>
- [32] «Spring Boot» dirección: <https://spring.io/>
- [33] «Ruby on Rails» dirección: <https://rubyonrails.org/>
- [34] «Laravel» dirección: <https://laravel.com/>
- [35] «Flask» dirección: <https://flask.palletsprojects.com/en/3.0.x/>
- [36] «MySQL» dirección: <https://www.mysql.com/>
- [37] «PostgreSQL» dirección: <https://www.postgresql.org/>
- [38] «Oracle Database» dirección: <https://www.oracle.com/es/database/>
- [39] «MongoDB» dirección: <https://www.mongodb.com/es>
- [40] «Redis» dirección: <https://redis.io/>
- [41] «Supabase» dirección: <https://supabase.com/>
- [42] «Visual Studio Code» dirección: <https://code.visualstudio.com/>
- [43] «Git» dirección: <https://git-scm.com/>
- [44] «GitHub» dirección: <https://github.com/>
- [45] «Docker» dirección: <https://www.docker.com/>
- [46] «Node.js» dirección: <https://nodejs.org/en>

- [47] «Postman» dirección: <https://www.postman.com/>
- [48] «Figma» dirección: <https://www.figma.com/>
- [49] «Project Libre» dirección: <https://www.projectlibre.com/>
- [50] «Overleaf» dirección: <https://es.overleaf.com>
- [51] «PWA (Progressive Web App)» dirección: https://developer.mozilla.org/es/docs/Web/Progressive_web_apps
- [52] «Firewoks.ia» dirección: <https://fireworks.ai/>
- [53] «Amazon Web Services» dirección: <https://aws.amazon.com/es/>
- [54] «Google Cloud Platform» dirección: <https://cloud.google.com/?hl=es>
- [55] «Microsoft Azure» dirección: <https://azure.microsoft.com/es-es>
- [56] «DigitalOcean» dirección: <https://www.digitalocean.com/>
- [57] «Netlify» dirección: <https://www.netlify.com/>
- [58] «Vercel» dirección: <https://vercel.com/>
- [59] «Heroku» dirección: <https://www.heroku.com/>
- [60] «Render» dirección: <https://render.com/>
- [61] «Vercel AI Playground» dirección: <https://sdk.vercel.ai/>
- [62] «Gemini» dirección: <https://gemini.google.com/app>
- [63] «Estudio de Viabilidad Económica» dirección: https://docs.google.com/spreadsheets/d/1qoehM_7Dz9zLd-W2IKFLjDY_PjfHnPSQ/edit?usp=sharing&oid=109085627910166737476&rtpof=true&sd=true
- [64] Becerra González A. (2022, 13 abril). 'Suscripción mensual o pago único: ¿qué prefieren los consumidores?'. YouGov. <https://es.yougov.com/international/articles/42103-suscripcion-mensual-o-pago-unico-que-prefieren-los>