

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR
FACULTAD DE HÁBITAT, INFRAESTRUCTURA Y CREATIVIDAD

Carrera en: Sistemas de la Información



Trabajo de Titulación

Desarrollo de un prototipo de aplicación móvil para crear apuntes inteligentes mediante la técnica de repetición espaciada y el uso de palacios mentales, apoyada por inteligencia artificial y desarrollada con tecnología React Native.

AUTOR:

Erick Antonio López Escobar

DIRECTORA:

Susana Masapanta Carrión

QUITO DM, JUNIO DE 2025

Dedicatoria

Dedico este trabajo a Dios, quien me ha guiado a través de valiosas lecciones que forjaron mi camino hasta este momento, permitiéndome afirmar mi propósito y superar mis propios límites.

A mis padres, quienes con sacrificio hicieron posible este logro. A mi hermana, pilar fundamental en mi formación como ser humano. Y a mi fiel compañero canino, cuya presencia hizo más llevaderas las largas jornadas de programación.

Agradecimientos

Extiendo mi más profunda gratitud a mis padres por su inquebrantable apoyo. A mi tía, quien representa un extraordinario modelo a seguir, digno de admiración. A mi directora, por compartir generosamente su conocimiento y por la paciencia demostrada durante todo el desarrollo de este trabajo.

Resumen

Este trabajo de titulación abarca el desarrollo de un prototipo de aplicación móvil enfocado en la creación de apuntes inteligentes mediante técnicas de repetición espaciada y palacios mentales, potenciado por inteligencia artificial.

Para el desarrollo de la aplicación se utilizó *React Native*, debido a su característica principal de permitir un desarrollo multiplataforma eficiente para la interfaz de usuario (*UI*). Además, la aplicación consume la *API* de OpenAI, la cual nos ayuda en algunos procesos realizados por nuestro *MCP*, que es n8n. En cuanto al almacenamiento de datos, optamos por *PostgreSQL*, una base de datos relacional que se adapta bien a nuestras necesidades. También empleamos una base de datos vectorial en *Supabase* para el almacenamiento especializado de ciertos datos.

La aplicación tiene como objetivo dar a conocer la técnica del método Loci, integrándola con los avances actuales presentes al momento de realizar este trabajo de titulación, tales como conceptos como *RAG*, *MCP*, agentes y bases de datos vectoriales.

Índice

Contenido

1. Introducción	10
1.1. Tema	10
1.2. Justificación	10
1.3. Planteamiento del problema.....	11
1.4. Objetivos	12
1.4.1. Objetivo General	12
1.4.2. Objetivos Específicos	12
1.5. Antecedentes	12
1.6. Alcance	13
2. Marco Teórico	14
2.1. Marco Conceptual	14
2.1.1. Expo React Native	14
2.1.2. Clerk.....	14
2.1.3. Frontend	15
2.1.4. Backend.....	15
2.1.5. API REST	15
2.1.6. Base de Datos Relacional	15
2.1.7. Bases de Datos Vectoriales.....	16
2.1.8. Programación Extrema (XP).....	16
2.1.9. MCP	16
2.1.10. N8N.....	17
2.1.11. Técnica de Repetición Espaciada	17
2.1.12. Palacios Mentales	17
2.2. Inteligencia Artificial en Aplicaciones Educativas	17
2.3. Frameworks para Desarrollo Móvil	17
2.4. Metodología Ágil Seleccionada	20
3. Modelado, Diseño y Desarrollo del Prototipo de la Aplicación.....	21
3.1. Iteración 1 – Presentación	21
3.1.1. Planificación	21
3.1.2. Codificación	26
3.1.3. Pruebas	26

3.2.	Iteración 2 – Home	29
3.2.1.	Planificación	29
3.2.2.	Diseño	33
3.2.3.	Tarjeta CRC	34
3.2.4.	Codificación	36
3.3.	Iteración 3 – Diccionario	42
3.3.1.	Planificación	42
3.3.2.	Diseño	45
3.3.3.	Codificación	48
3.3.4.	Pruebas	51
3.4.	Iteración 4 – Chat	53
3.4.1.	Planificación	53
3.4.2.	Diseño	56
3.4.3.	Codificación	58
3.4.4.	Pruebas	61
3.5.	Modelo Conceptual de la Aplicación.....	63
3.6.	Arquitectura Funcional.....	65
3.7.	Definición de los Módulos Principales.....	70
3.7.1.	Personalización de la aplicación con IA	70
3.7.2.	Implementación de la Repetición Espaciada	70
3.7.3.	Creación de los Palacios Mentales	71
4.	Pruebas y Evaluación	72
4.1.	Definición del entorno de prueba controlado.....	72
4.2.	Pruebas Funcionales.....	73
4.3.	Evaluación de la experiencia del Usuario.....	74
4.3.1.	Metodología	74
4.3.2.	Perfil de los usuarios:	74
4.3.3.	Estructura de las pruebas	75
4.3.4.	Métodos utilizados	76
4.4.	Análisis de Resultados	76
4.4.1.	Resultados Pre-Test	76
4.4.2.	Resultados Post-Test	83
5.	Conclusiones y Recomendaciones	100

5.1. Conclusiones	100
5.2. Recomendaciones	101
6. Referencias.....	102
7. Anexos.....	104

Lista de Tablas

Tabla 1	Ventajas y Desventajas de LynxsJS.....	18
Tabla 2	Ventajas y Desventajas de React Native	19
Tabla 3	Ventajas y Desventajas de Flutter	19
Tabla 4	Historia de Usuario: Presentación.....	21
Tabla 5	Plan de Iteración: Presentación	22
Tabla 6	Criterio de Aceptación: Presentación	23
Tabla 7	Tarjeta CRC: Presentación.....	24
Tabla 8	Pruebas de Aceptación: Presentación	27
Tabla 9	Prueba de Caja Blanca: Presentación	28
Tabla 10	Historia de Usuario: Home	29
Tabla 11	Plan de Iteración: Home	31
Tabla 12	Criterio de Aceptación - Visualización del Carrusel en Home	32
Tabla 13	Criterio de Aceptación - Subida de Contenido y Embedding en Home	32
Tabla 14	Criterio de Aceptación - Edición de Perfil en Home	32
Tabla 15	Tarjeta CRC - Home	34
Tabla 16	Prueba de Aceptación - Visualización del Carrusel en Home	39
Tabla 17	Prueba de Aceptación - Contenido y Embedding en Home.....	39
Tabla 18	Prueba de Aceptación - Visualización del Carrusel en Home	40
Tabla 19	Historia de Usuario - Diccionario	42
Tabla 20	Plan de Iteración - Diccionario	44
Tabla 21	Criterio de Aceptación - Visualización Diccionario	44
Tabla 22	Criterio de Aceptación - Definiciones Diccionario	45
Tabla 23	Criterio de Aceptación - Edición de Palabras.....	45
Tabla 24	Tarjeta CRC - Diccionario	47
Tabla 25	Prueba de Aceptación - Visualización Diccionario	51
Tabla 26	Pruebas de Aceptación - Consulta Diccionario	51
Tabla 27	Caja Blanca - Diccionario.....	52
Tabla 28	Historia de Usuario - Chat	54
Tabla 29	Plan de Iteración - Chat	55
Tabla 30	Criterio de Aceptación - Repasar en Chat	55
Tabla 31	Criterio de Aceptación - Consultar en Chat	56
Tabla 32	Tarjeta CRC - Chat	57
Tabla 33	Prueba de Aceptación - Repasar en Chat.....	61
Tabla 34	Prueba de Aceptación - Consultar en Chat	61
Tabla 35	Prueba de Caja Blanca - Chat.....	62

Lista de Figuras

Figura 1	Diseño: Presentación	24
Figura 2	Prototipo: Presentación	25
Figura 3	Diseño: Home.....	34
Figura 4	Diseño: Home.....	36
Figura 5	Diseño: Diccionario.....	46
Figura 6	Prototipo: Diccionario.....	48
Figura 7	Diseño: Chat.....	57
Figura 8	Prototipo: Chat.....	58
Figura 9	Esquema de Base de Datos	65
Figura 10	Flujo de Embedding.....	67
Figura 11	Flujo de Extracción de Ideas Principales	68
Figura 12	Agente de Creación de Historias.....	69
Figura 13	Resultados de la pregunta 1 Pre-Test	77
Figura 14	Resultados de pregunta 2 Post-Test.....	77
Figura 15	Resultados de la pregunta 3 Post-Test.....	78
Figura 16	Resultados de la pregunta 4 Pre-Test	79
Figura 17	Resultados de la pregunta 5 Pre-Test	79
Figura 18	Resultados de la pregunta 6 Pre-Test	80
Figura 19	Resultados de la pregunta 7 Pre-Test	81
Figura 20	Resultados de la pregunta 8 Pre-Test	81
Figura 21	Resultados de pregunta 9 Pre-Test.....	82
Figura 22	Resultados de pregunta 10 Pre-Test.....	83
Figura 23	Resultados de pregunta 1 Post-Test	84
Figura 24	Resultados de pregunta 2 Post-Test	85
Figura 25	Resultados de la pregunta 3 Post-Test.....	85
Figura 26	Resultados de la pregunta 4 Post-Test.....	86
Figura 27	Resultados de la pregunta 5 Post-Test.....	87
Figura 28	Resultados de la pregunta 6 Post-Test.....	88
Figura 29	Resultados de la pregunta 7 Post-Test.....	89
Figura 30	Resultados de la pregunta 8 Post-Test.....	90
Figura 31	Resultados de la pregunta 9 Post-Test.....	91
Figura 32	Resultados de la pregunta 10 Post-Test	91
Figura 33	Respuestas de la pregunta 11 Post-Test	92
Figura 34	Resultados de la pregunta 12 Post-Test	93
Figura 35	Resultados de la pregunta 13 Post-Test	93
Figura 36	Resultados de la pregunta 14 Post-Test	94

1. Introducción

1.1. Tema

Desarrollo de un prototipo de aplicación móvil para crear apuntes inteligentes mediante la técnica de repetición espaciada y el uso de palacios mentales, apoyada por inteligencia artificial y desarrollada con tecnología *React Native*.

1.2. Justificación

En la actualidad podemos ver que de manera contemplativa como muchos estudiantes y profesionales enfrentan la necesidad de memorizar y organizar grandes cantidades de información en poco tiempo, esto debido a la consecuencia directa que tienen los avances tecnológicos que cada día dan grandes pasos agigantados, lo que ha llevado a buscar técnicas más efectivas de estudio. Una de ellas es el uso de palacios mentales, estrategia mnemotécnica que permite estructurar y almacenar conocimientos mediante asociaciones espaciales y visuales que, si nos detenemos a pensar un instante, tienen una connotación muy parecida la estructura de un cuento, como menciona el autor Domínguez (2023) “Si algo hace especial al ser humano es su capacidad por contar historias. El porqué es una cuestión a debate, pero lo incuestionable es que lo lleva haciendo desde el principio de los tiempos”. Sin embargo, su aplicación manual requiere altos niveles de esfuerzo, constancia y creatividad, lo cual representa un obstáculo para su adopción.

Además, las herramientas tecnológicas existentes como *Notion*, *Google*, *NotebookLM*, *RoomSearch* y *Obsidian*, aunque cuentan con aplicaciones móviles, no están diseñadas específicamente para guiar al usuario en la construcción adecuada de mapas mentales ni

incorporan métodos de memoria, como el método Loci en sus funcionalidades. *Notion* es una plataforma de productividad para crear notas y bases de datos; *Google NotebookLM* es una herramienta basada en IA que permite organizar y analizar información compleja, incluyendo mapas mentales dinámicos; *Obsidian* facilita la creación de redes de conocimiento mediante enlaces entre notas. Esta falta de soporte específico hace que muchas personas, a pesar de su interés terminen abandonando estas técnicas.

Frente a este escenario, el desarrollo de una aplicación móvil que facilite la creación de apuntes inteligentes y combine técnicas de repetición espaciada con la lógica que mantienen los palacios mentales, representa una propuesta de innovación relevante.

1.3. Planteamiento del problema

A pesar de la existencia de técnicas como la repetición espaciada y los palacios mentales, su aplicación sigue siendo escasa debido a dos factores principales: la falta de familiaridad con su funcionamiento y la carencia de herramientas tecnológicas que faciliten su uso.

Al momento de realizar el presente trabajo de titulación, la mayoría de las aplicaciones disponibles no ofrecen funcionalidades integradas ni acompañamiento inteligente para estructurar el conocimiento de manera personalizada, como lo hace el método Loci o Palacios Mentales, lo cual limita su impacto real en su comprensión profunda.

Ante esta situación, surge la necesidad de diseñar y desarrollar un prototipo de aplicación móvil que integre técnicas de repetición espaciada y palacios mentales, asistida por inteligencia artificial, con el objetivo de ofrecer una herramienta accesible, interactiva y adaptable que promueva y facilite el uso de la técnica del método Loci.

En función de esta problemática se plantea la siguiente pregunta principal de investigación:

¿El prototipo de aplicación móvil permitirá a los usuarios practicar y reforzar la técnica del método Loci mediante ejercicios interactivos y personalizados?

1.4. Objetivos

1.4.1. Objetivo General

Desarrollar una aplicación móvil con *React Native* que cree apuntes inteligentes usando repetición espaciada y palacios mentales asistidos por IA, para optimizar el tiempo de creación de apuntes, usando el método de Palacios Mentales.

1.4.2. Objetivos Específicos

- Investigar acerca de la repetición espaciada y los palacios mentales, para su implementación en una aplicación móvil.
- Incorporar inteligencia artificial (LLM) usando *LangChain* que es un *framework* que te permite crear aplicaciones inteligentes utilizando modelos de lenguaje avanzados (como GPT) para personalizar la creación de apuntes digitales.
- Diseñar y desarrollar la interfaz y funcionalidades de la aplicación móvil con *React Native*, garantizando una experiencia de usuario intuitiva.
- Validar que la aplicación ofrece una experiencia intuitiva y eficaz en la creación de apuntes con IA, mediante pruebas con usuarios.

1.5. Antecedentes

El uso de palacios mentales funciona mediante la creación de imágenes en lugares específicos, donde cada imagen actúa como un disparador (*trigger*) que detona o asocia información determinada a esa imagen (Defeyter et al., 2009). “Sus orígenes se remontan aproximadamente al año 500 a.C., hace más de 2,000 años” (Day, 2020). Según la historia, un hombre que había dado un discurso en un lugar fue el único sobreviviente de un derrumbe. Al identificar a las víctimas, pudo recordar con exactitud quién estaba sentado en cada lugar gracias a la reciente experiencia del discurso, lo que evidenció la efectividad de esta técnica.

Con el tiempo, diversas técnicas de memoria como la repetición, mapas mentales, rimas y tarjetas de memoria se han convertido en herramientas clave para muchas personas. Por mencionar alguna de ellas tenemos a Notion que funciona como una base de datos para registrar tus apuntes o también por el otro lado esta Obsidian ofreciendo una propuesta que permite conectar tus notas en una sola red de conocimiento, buscando relaciones en lo que escribes y finalmente tenemos una aplicación lanzada por Google que la idea es ayudarte a consolidar ese conocimiento en uno solo con ayudas como: resúmenes, transcripciones, creación de notas y guías de estudio llamada Google Notebook LM.

No obstante, en el año 2019 ya existía una aplicación que incorporaba la idea de un palacio mental en sus procesos, el nombre de esta aplicación es memoryOs que con la idea de un juego busca ayudar al usuario a retener de la información con esta técnica que no muchas personas conocen.

1.6. Alcance

Este proyecto abarca el diseño y desarrollo de un prototipo funcional de una aplicación móvil para la creación de apuntes inteligentes usando la técnica de repetición espaciada y el

método Loci o también conocido como palacios mentales con la ayuda de un modelo de lenguaje para la creación de apuntes. Para el desarrollo de la aplicación se optó por *utilizar React Native* para la interfaz de usuario, esto debido a su compatibilidad multiplataforma.

Las principales características de esta aplicación cubren particularidades como: generación de notas, asignaciones de repasos empleando la repetición espaciada y un glosario para mantener todas las asociaciones referidas a una idea principal sobre el contenido subido. Cualquier expansión hacia un producto final o su lanzamiento al mercado quedará fuera del alcance actual y dependerá de decisiones futuras.

2. Marco Teórico

2.1. Marco Conceptual

2.1.1. *Expo React Native*

En la aplicación, Expo React Native se utiliza como el *framework* principal para el desarrollo del *frontend* móvil. Gracias a Expo, se simplifica la creación de la interfaz de usuario y la gestión de funcionalidades nativas en dispositivos Android y iOS, permitiendo un desarrollo ágil y multiplataforma. Además, Expo facilita el testing y despliegue rápido de la aplicación durante las iteraciones (Coursera Staff, 2024).

2.1.2. *Clerk*

Clerk se integra como el sistema de autenticación moderno y seguro para la aplicación. Tyagi (2024) describió que puede proporcionar funcionalidades como registro, inicio de sesión, autenticación multifactor y *login* social (Google, GitHub, etc.), garantizando una gestión eficiente y segura de los usuarios.

2.1.3. Frontend

El *frontend* de la aplicación es la interfaz desarrollada con *React Native* y *Expo*, que como lo menciona Nicole (2025) permite a los usuarios interactuar con las funcionalidades como el *onboarding*, el chat, el diccionario, y la gestión de contenido. Es la capa visible que recibe las acciones del usuario y muestra la información procesada por el backend, asegurando una experiencia amigable y responsiva.

2.1.4. Backend

El *backend*, desarrollado con *NestJS*, es el motor invisible que maneja la lógica de negocio, la gestión de usuarios, el procesamiento de archivos, la generación de *embeddings* y la interacción con la base de datos (Nicole, 2025). Se encarga de recibir las solicitudes del *frontend*, procesarlas de forma segura y eficiente, y devolver las respuestas adecuadas para mantener la integridad y funcionalidad del sistema.

2.1.5. API REST

La comunicación entre el *frontend* y *backend* se realiza mediante una API REST, que sigue los principios de diseño REST para garantizar una interfaz clara, escalable y mantenible (*¿Qué Es Una API de REST?*, 2023). Esta API permite realizar operaciones CRUD sobre usuarios, historias, diccionarios, y gestionar la subida y procesamiento de contenido, facilitando la integración y modularidad del sistema.

2.1.6. Base de Datos Relacional

Se utiliza una base de datos relacional PostgreSQL para almacenar la información estructurada de usuarios, diccionarios, palabras, historias, documentos y notas. Esta base de

datos permite mantener relaciones claras y consistentes entre las entidades, facilitando consultas complejas y asegurando la integridad de los datos a lo largo del ciclo de vida de la aplicación (*¿Qué Es Una Base de Datos Relacional? | IBM, 2021*).

2.1.7. Bases de Datos Vectoriales

Para las búsquedas semánticas y la gestión avanzada de contenido, la aplicación utiliza una base de datos vectorial implementada con *Supabase* y la extensión *pgvector*. Esta base almacena los *embeddings* generados a partir de los textos procesados, permitiendo realizar búsquedas por similitud y mejorar la precisión en la recuperación de información relevante para el usuario. Como menciona el autor Schwaber-Cohen (2023) “una base de datos vectorial indexa y almacena incrustaciones vectoriales para una recuperación rápida y búsqueda de similitudes, con capacidades como operaciones CRUD, filtrado de metadatos, escalado horizontal y sin servidor”.

2.1.8. Programación Extrema (XP)

“La programación extrema es una metodología ágil de gestión de proyectos que se centra en la velocidad y la simplicidad con ciclos de desarrollo cortos y con menos documentación” (Raeburn, 2025).

2.1.9. MCP

El MCP (Model Context Protocol) actúa como un servidor intermediario que conecta el modelo de IA (GPT-4.1-mini) con las fuentes de datos de la aplicación (González, 2025). Facilita la consulta y recuperación en tiempo real de información procesada, como *embeddings* y resúmenes, permitiendo que el modelo genere respuestas contextualizadas y personalizadas para el usuario.

2.1.10. N8N

“N8N es una plataforma de automatización de flujos de trabajo de código abierto. Creado en 2019 por Jan Oberhauser en Berlín, permite el modelado de procesos de negocio y la integración e interacción de aplicaciones” (*N8n: An Overview of the Workflow Automation Tool*, 2024).

2.1.11. Técnica de Repetición Espaciada

“La repetición espaciada, también conocida como repaso espaciado, es una técnica de aprendizaje memorístico que consiste en aprender un material determinado dejando pasar espacios de tiempo cada vez más largos entre una sesión de entrenamiento y la siguiente” (Figueroba, 2017) .

2.1.12. Palacios Mentales

El método de los loci es una técnica probada para mejorar la memoria que es tan antigua como la antigua Grecia y Roma. Utiliza enlaces entre la información y las ubicaciones físicas visualizadas. Nuestro cerebro recuerda lugares más fácilmente que hechos, y esa es la naturaleza de la técnica de los loci (Von Essen, 2022).

2.2. Inteligencia Artificial en Aplicaciones Educativas

2.3. Frameworks para Desarrollo Móvil

Al momento de realizar el presente trabajo de titulación, en el del desarrollo móvil multiplataforma, contamos con tres opciones destacadas, cada una con sus particularidades: que se detallan a continuación:

- *React Native*, se ha consolidado como una solución madura y ampliamente adoptada. Su mantenimiento continuo y robusto ecosistema lo han posicionado como referente en la industria.
- *Flutter*, ofrece un enfoque integral con capacidades multiplataforma y un rendimiento gráfico excepcional, gracias a su arquitectura basada en widgets personalizables.
- *LynxJS*, emerge como un competidor prometedor, especialmente diseñado para superar limitaciones de rendimiento y ofrecer mayor flexibilidad en el desarrollo.

Como se observa en las Tablas 1, 2 y 3, a partir del análisis comparativo de ventajas y desventajas, se concluye que el framework más adecuado para este desarrollo es *React Native*.

Tabla 1

Ventajas y Desventajas de LynxsJS

Cita	Framework	Ventajas	Desventajas
(<i>Lynx JS vs Flutter vs React Native</i> , 2025)	Lynx JS	<ol style="list-style-type: none"> 1. Alto rendimiento con arquitectura multihilo y motor <i>Rust</i>. 2. Uso de <i>JavaScript</i> y CSS, fácil para desarrolladores web. 3. Uso de JavaScript y CSS, fácil para desarrolladores web. 4. Excelente para apps con UI muy 	<ul style="list-style-type: none"> • Comunidad pequeña y ecosistema limitado. • No es ideal para crear apps desde cero, más para integrar UI. • Herramienta nueva, menos documentación y recursos.

interactivas y animaciones.

Tabla 2

Ventajas y Desventajas de React Native

Cita	Framework	Ventajas	Desventajas
(Jain, 2025)	React Native	<ol style="list-style-type: none">1. Ecosistema maduro y gran comunidad.2. Reutilización de código entre web y móvil con React.3. Amplia variedad de librerías y herramientas.4. Desarrollo rápido y soporte de Expo para facilitar el flujo.	<ul style="list-style-type: none">• Tamaño de app mayor por motor JS• Puede requerir optimizaciones para apps complejas.• Rendimiento limitado por puente JS a código nativo

Tabla 3

Ventajas y Desventajas de Flutter

Cita	Framework	Ventajas	Desventajas
(Jain, 2025)	Flutter	<ol style="list-style-type: none">1. UI <i>pixel-perfect</i> con control total gracias a su motor de renderizado.2. Alto rendimiento con compilación nativa y GPU acelerada.3. Soporte oficial para móvil, web y escritorio.4. Comunidad grande y en crecimiento con muchas librerías.	<ul style="list-style-type: none">• Tamaño de app más grande por motor <i>Skia</i> y runtime <i>Dart</i>.• Curva de aprendizaje por usar <i>Dart</i>.

2.4. Metodología Ágil Seleccionada

Debido a la naturaleza del proyecto, se ha decidido implementar la metodología ágil *Extreme Programming*, ya que permite una entrega más eficiente y es especialmente adecuada porque requiere menos documentación en comparación con otras metodologías.

3. Modelado, Diseño y Desarrollo del Prototipo de la Aplicación

3.1. Iteración 1 – Presentación

3.1.1. Planificación

3.1.1.1. Historias de Usuario

Para esta historia de usuario, se buscará crear las pantallas de *onboarding* de la aplicación, que darán la bienvenida a los usuarios. Además, se incluirá el proceso de registro dentro de la aplicación. También se formularán pruebas y criterios de aceptación relacionados con este proceso, los cuales estarán detallados en la Tabla 4.

Tabla 4

Historia de Usuario: Presentación

Código de Historia	Enunciado de la Historia				Criterio de Aceptación			
	Rol	Funcionalidad	Razón	Escenario	Criterio	Contexto	Evento	Resultado
A1	Administrador	Guiar al usuario a través de pantallas antes del registro	Mostrar las principales funciones de la aplicación	1	Ver las pantallas al iniciar la aplicación	Pantalla de Bienvenida mostrada	Usuario presiona el botón para continuar la guía	Usuario conoce los puntos importantes de la aplicación.
A2	Administrador	Registrarse en la aplicación	Para tener una base de datos de las personas que usan	1	El usuario puede registrarse con correo o	Pantalla de registro mostrada después del <i>onboarding</i>	Usuario completa y envía el formulario	El usuario es creado exitosamente en el Sistema

			nuestra app		Google Auth.		de registro	
--	--	--	-------------	--	--------------	--	-------------	--

Nota. Fuente: Elaboración propia

Durante el desarrollo de la aplicación, se pudo constatar que una de las necesidades primarias que debe ser cubierta es realizar una introducción sobre los conceptos que va a tratar la aplicación que en este caso es la idea de un palacio mental con el propósito de que los nuevos usuarios puedan tener una idea más clara de la técnica que va a abordar la aplicación

De la misma manera se encontró la necesidad de contar con un proceso de registro para poder asociar cada uno de los documentos que serán subidos a la aplicación a un determinado usuario y para facilidad de este, fue una buena idea implementar el registro con Google. Además, este proceso sienta las bases para futuras funcionalidades de gestión de usuarios.

3.1.1.2. Plan de Iteración

Las fechas que se prevén para el desarrollo del módulo de Presentación están contempladas para el 1 de junio del 2025 hasta el 5 de junio del 2025.

Tabla 5

Plan de Iteración: Presentación

Plan de Iteración					
# de Iteración	Fecha Inicio	Fecha Fin	Numero de Historia	Nombre de Historia	Responsable
1	01/06/2025	05/06/2025	1	Presentación	Erick López

2	06/06/2025	11/06/2025	2	Home	Erick López
3	12/06/2025	17/06/2025	3	Diccionario	Erick López
4	18/06/2025	25/06/2025	4	Chat	Erick López

Nota. Fuente: Elaboración propia

3.1.1.3. Criterios de Prueba de Aceptación

El usuario podrá visualizar varias pantallas que le servirán de apoyo para poder entender un poco más sobre la técnica a aplicar. También al finalizar estas tarjetas de presentación podrá continuar hacia una pantalla de registro. Asegurando que las funcionalidades cumplan con los criterios de aceptación previamente establecidos.

Tabla 6

Criterio de Aceptación: Presentación

Nombre	Visión de la Prueba		
	Presentación de la Aplicación	Código	E1
N. Historia	1		
Iteración	1		
Función Crítica	Si		
Descripción	Verificar la creación del usuario y ver las pantallas de presentación.		
Categoría	Presentación		
Prerrequisito	La aplicación tiene que estar expuesta usando <i>Expo Go</i> para poder conectarse mediante un dispositivo compatible como Android o IOS además de que los servicios del <i>backend</i> deben estar activos.		

Nota. Fuente: Elaboración propio

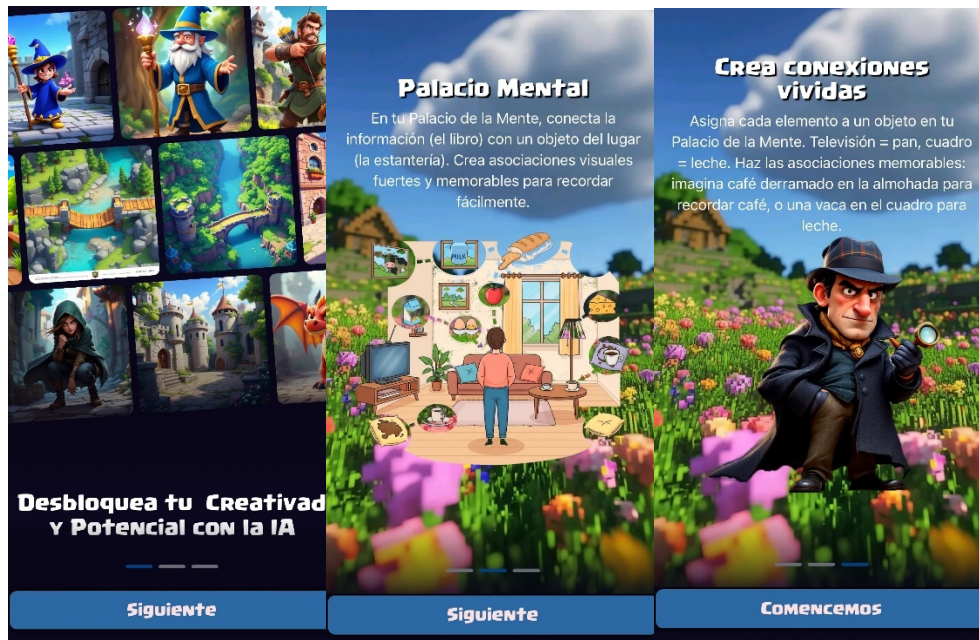
3.1.1.4. Diseño

Para el diseño de esta historia se tiene en mente utilizar tarjetas deslizables en una misma pantalla como un carrusel con una animación suave y transitiva para que el usuario pueda ver de forma dinámica las principales características de la aplicación. En la última tarjeta el botón

redireccionará a la pantalla del registro donde el usuario podrá ingresar su nombre, email y contraseña.

Figura 1

Diseño: Presentación



3.1.1.5. Tarjeta CRC

Para las tarjetas CRC se diseñó una manera de asignar de manera clara las responsabilidades y métodos implementados en la aplicación, alineadas con las funcionalidades definidas en la historia de usuario.

Tabla 7

Tarjeta CRC: Presentación

Numero	T01	Nombre	Presentación
--------	-----	--------	--------------

Descripción	Permite el ingreso de los datos en cada uno de los campos como los son: nombre, email y contraseña	
Responsabilidades	Colaboradores	Métodos
Registro del Usuario	Usuario	onSignupPress

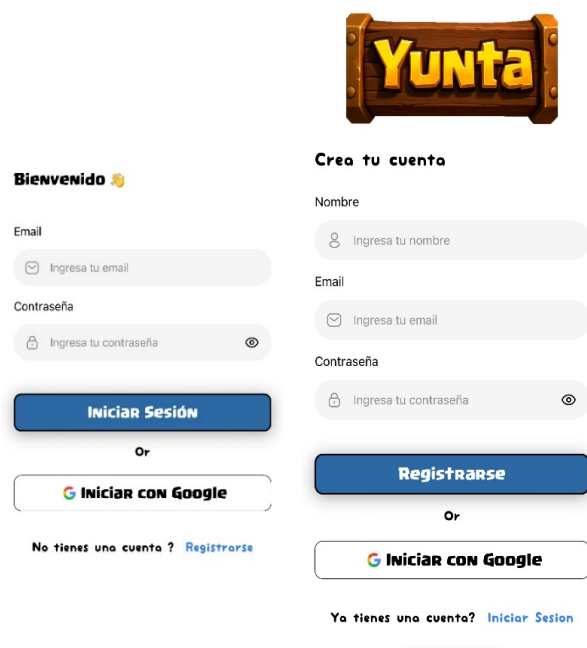
Nota. Fuente: Elaboración propia

3.1.1.6. Prototipo

El prototipo incluirá una visualización dinámica de las tarjetas de presentación con una animación suave, así como la captura de los datos en cada uno de los campos especificados en la pantalla de registro para la creación del usuario.

Figura 2

Prototipo: Presentación



3.1.2. Codificación

3.1.2.1. Estándares de Codificación

Identificadores: El ID será un valor único y auto incremental para cada registro.

Variables: Las variables harán referencia directa a las características o atributos del objeto que representan.

Funciones: Tendrán el nombre asociado a la acción que realizan.

Bases: Se implementará un ORM (Object-Relational Mapping), en este caso *Prisma*, que permitirá una interacción eficiente con la base de datos *PostgreSQL*.

3.1.2.2. Código

Clase: Usuario contará con los siguientes campos: Id, Nombre, Correo Electrónico y Contraseña.

Controlador: `getUsers`, `createUser`, `updateUser`, `deleteUser` y `getUserById`:

- **getUsers:** Listará los usuarios registrados en una tabla que permitirá realizar búsquedas y filtrados.
- **createUser:** Almacenará los datos de un nuevo usuario en la base de datos.
- **updateUser:** Actualizará la información del usuario seleccionado.
- **deleteUser:** Eliminará al usuario correspondiente según su Id.
- **getUserById:** Recuperará los roles asignados a un usuario específico.

3.1.3. Pruebas

El proceso de pruebas se enfocó en validar que las funcionalidades implementadas cumplieran con los criterios de aceptación establecidos previamente, asegurando así el correcto funcionamiento del módulo y su aceptación por parte de los usuarios finales y *stakeholders* del proyecto.

3.1.3.1. Pruebas de Aceptación

Tabla 8

Pruebas de Aceptación: Presentación

Pruebas de Aceptación	
Nombre:	Prueba de Registro y Presentación de Usuario
Descripción:	Verificar que el usuario pueda visualizar correctamente las pantallas de presentación y completar el proceso de registro mediante correo electrónico o autenticación externa, cumpliendo con los criterios de aceptación definidos.
Condiciones de Ejecución	<ol style="list-style-type: none"> 1. La aplicación debe estar conectada a <i>Expo Go</i> en la misma red y funcionando en un dispositivo compatible. 2. El <i>backend</i> y servicios de autenticación (<i>Clerk</i>) deben estar activos. 3. El usuario debe iniciar la aplicación sin sesión activa previa.
Resultado Esperado	<ul style="list-style-type: none"> ○ La pantalla de presentación se muestra correctamente con la animación del carrusel.

- El usuario puede navegar por la guía introductoria sin errores.
- El formulario de registro acepta datos válidos y permite completar el registro.
- El sistema confirma la creación exitosa del usuario.

Evaluación de la Prueba Se consideró exitoso, ya que todas las pantallas se muestran correctamente y el usuario puede registrarse sin errores.

Nota. Fuente: Elaboración propia

3.1.3.2. Pruebas de Caja Blanca

Tabla 9

Prueba de Caja Blanca: Presentación

Entradas	Resultado Esperado	Resultado
Usuario abre la aplicación por primera vez	Se muestra la pantalla de bienvenida con el carrusel de presentación animado	Pantalla mostrada correctamente
Usuario presiona botón "Siguiente" en la guía	Se avanza a la siguiente tarjeta del carrusel sin errores	Navegación correcta
Usuario presiona botón "Anterior" en la guía	Se regresa a la tarjeta anterior sin errores	Navegación correcta
Usuario completa formulario con datos válidos: nombre, email, contraseña	El método <i>onSignupPress</i> valida y crea el usuario en la base de datos	Usuario creado exitosamente
Usuario completa formulario con email inválido	El método valida y muestra mensaje de error por formato incorrecto	Mensaje de error mostrado

Usuario intenta registrarse con campos vacíos	El sistema bloquea el envío y muestra mensajes de validación	Validaciones activas y mensajes mostrados
---	--	---

Nota. Fuente: Elaboración propia

3.2. Iteración 2 – Home

3.2.1. Planificación

3.2.1.1. Historias de Usuario

Para esta historia de usuario, se buscará crear la pantalla principal (Home) de la aplicación, que permitirá al usuario visualizar un carrusel con tres historias de ejemplo para revisar antes de subir su propio contenido. Además, se incluirá un botón para que el usuario pueda subir nuevas historias, iniciando el proceso de generación de *embeddings* y su inserción en la base de datos vectorial. También se tiene pensando poder darle al usuario la opción de personalizar un poco su perfil, donde podrá cambiar su icono y nombre de usuario. Además, es importante destacar que se formularán pruebas y criterios de aceptación relacionados con estas funcionalidades que se encuentran detallados en la Tabla 10.

Tabla 10

Historia de Usuario: Home

	Enunciado de la Historia				Criterio de Aceptación			
Código de Historia	Rol	Funcionalidad	Razón	Escenario	Criterio	Contexto	Evento	Resultado
B1	Usuario	Visualizar carrusel con historias de ejemplo	Permitir al usuario conocer ejemplos antes de	1	Ver el carrusel al ingresar	Usuario autenticado y en Home	Usuario navega por el	Usuario puede revisar las historias

			subir contenido		r a Home		carrusel	de ejemplo
B2	Usuario	Subir contenido para procesamiento de <i>embeddings</i>	Permitir al usuario agregar sus propias historias y que se procesen para búsquedas semánticas	1	Botón “crear historia” visible y funcional	Usuario en Home	Usuario presiona botón “crear historia” para subir historia	Se inicia el proceso de <i>embedding</i> y almacenamiento en vector store
B3	Usuario	Editar perfil de usuario	Permitir al usuario actualizar su icono, y nombre de usuario		Pantalla de perfil accesible	Usuario en Home	Usuario accede a perfil y modifica datos	Cambios guardados y reflejados en la base de datos

Nota. Fuente: Elaboración propia.

Una vez recolectada la información de la historia de usuario podemos inferir que una de las características claves que se busca en la pantalla del Home es que sea simple y fácil de comprender y para poder acercarnos a ese resultado se realizara un carrusel que por defecto contara con tres ítems que serán historias que servirán como ejemplo de lo que el usuario puede llegar a conseguir al momento de subir su contenido con el botón de “agregar historia” que se encontrara al final del carrusel.

Finalmente, el usuario contara con la opción de poder personalizar su perfil, permitiéndole actualizar su icono, correo electrónico y nombre de usuario.

3.2.1.2. *Plan de Iteración*

Las fechas que se prevén para el desarrollo del módulo de Presentación están contempladas para el 6 de junio del 2025 hasta el 11 de junio del 2025.

Tabla 11

Plan de Iteración: Home

Plan de Iteración					
# de Iteración	Fecha Inicio	Fecha Fin	Numero de Historia	Nombre de Historia	Responsable
1	01/06/2025	05/06/2025	1	Presentación	Erick López
2	06/06/2025	11/06/2025	2	Home	Erick López
3	12/06/2025	17/06/2025	3	Diccionario	Erick López
4	18/06/2025	25/06/2025	4	Chat	Erick López

Nota. Fuente: Elaboración propia

3.2.1.3. *Criterios de Prueba de Aceptación*

Para la historia de usuario Home se tiene pensando verificar que el usuario pueda desplazar cada una de las historias deslizando su dedo de izquierda a derecha para poder avanzar o retroceder a la siguiente historia. También se validará que el usuario pueda personalizar su perfil cambiando el icono de su perfil que viene por defecto, así como también su nombre de usuario. Y como ultimo pasa se verificará que el usuario suba algún tipo de contenido con extensiones .pdf, .md o .txt.

Tabla 12*Criterio de Aceptación - Visualización del Carrusel en Home*

	Visión de la Prueba		
Nombre	Visualización del Carrusel	Código	E2
N. Historia	2		
Iteración	2		
Función Crítica	Si		
Descripción	Verificar que el carrusel con las historias de ejemplo se muestre correctamente en Home.		
Categoría	Home		
Prerrequisito	Usuario autenticado, aplicación <i>Expo Go</i> abierta en la misma red en undispositivo compatible, sin errores en <i>backend</i> .		

Nota. Fuente: Elaboración propia**Tabla 13***Criterio de Aceptación - Subida de Contenido y Embedding en Home*

	Visión de la Prueba		
Nombre	Subida de Contenido y Embedding	Código	E3
N. Historia	2		
Iteración	2		
Función Crítica	Si		
Descripción	Comprobar al momento de presionar el botón “crear historia” que se inicie el proceso de subida y generación de <i>embeddings</i> , y que la historia generada por los agentes sea exitosa.		
Categoría	Home		
Prerrequisito	Usuario autenticado, <i>backend</i> y servicios de <i>embedding</i> activos, conexión estable a internet.		

Nota. Fuente: Elaboración propia**Tabla 14***Criterio de Aceptación - Edición de Perfil en Home*

	Visión de la Prueba		
Nombre	Edición de Perfil	Código	E4
N. Historia	2		
Iteración	2		
Función Crítica	Si		
Descripción	Verificar que en el momento en que el usuario presione sobre el icono donde está su correo sea capaz modificar su icono y nombre de usuario, y		

	que después de haber realizado esos cambios los cambios se guarden correctamente.
Categoría	Home
Prerrequisito	Usuario autenticado, aplicación funcionando correctamente, base de datos accesible.

Nota. Fuente: Elaboración propia

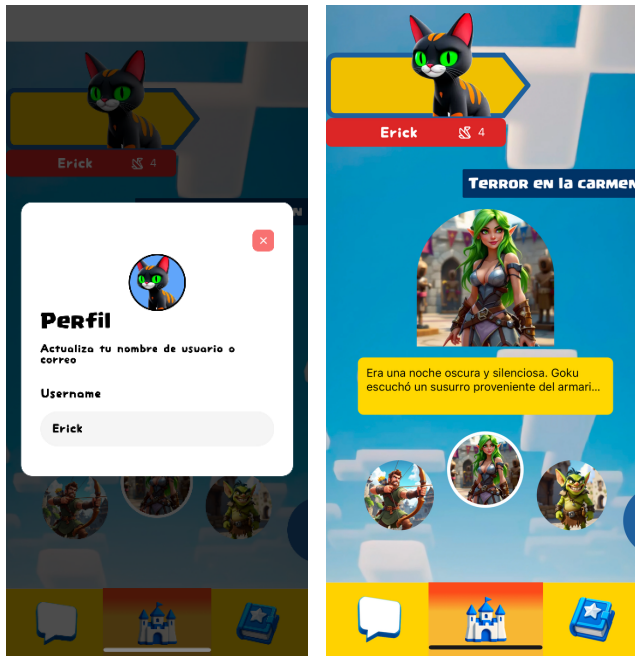
3.2.2. Diseño

Para el diseño de la pantalla Home se tiene pensado utilizar una lista que cuente con tres ítems, que van a poder ocupar la pantalla principal con un deslizamiento de izquierda a derecha con una animación que cambia el contenido por el contenido del *item* que este seleccionado. Al final de estos tres ítems se colocará una opción que contara con el símbolo del “+” para poder desplegar el botón de “agregar historia” que su funcionalidad entra en juego cuando el usuario presiona el botón para que pueda aparecer un modal en donde el usuario podrá elegir el contenido con extensión .pdf, .md o .txt que desee subir a la aplicación.

Además, en la parte superior izquierda de la pantalla, se encuentra el perfil del usuario que está representado por un icono que viene por defecto y el correo del usuario.

Figura 3

Diseño: Home



3.2.3. Tarjeta CRC

Tabla 15

Tarjeta CRC - Home

Numero	T02	Nombre	Presentación
Descripción	Gestiona la visualización de historias de ejemplo, subida de contenido y edición de perfil		
Responsabilidades	Colaboradores	Métodos	
Mostrar carrusel de historias	Usuario	<i>onSignupPress</i>	
Iniciar subida de contenido	Usuario	<i>onUploadPress</i>	
Gestionar edición de perfil	Usuario	<i>onProfileEdit</i>	

Nota. Fuente: Elaboración propia

3.2.3.1. Prototipo

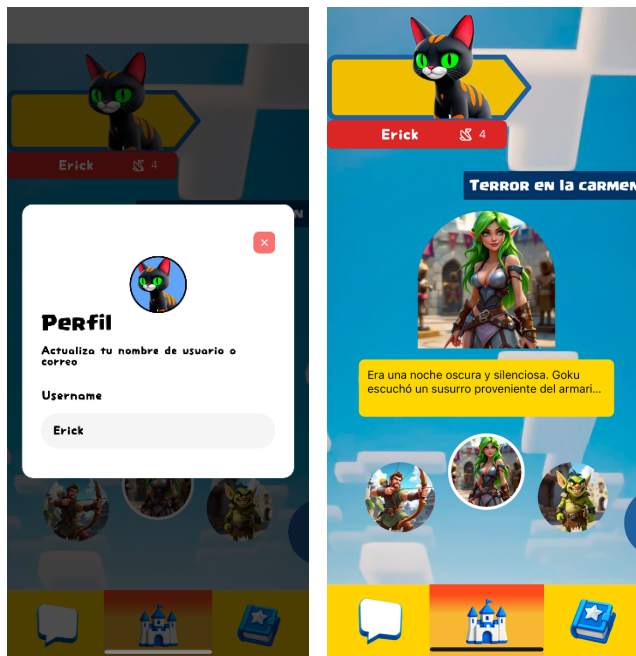
El prototipo de la pantalla Home está diseñado para facilitar la interacción del usuario con las funcionalidades principales de la aplicación. Incluye un carrusel horizontal con tres historias de ejemplo que el usuario puede revisar deslizando de forma fluida y visualmente atractiva. Cada tarjeta del carrusel al presionarla muestra la historia, con una sección más en donde estarán las palabras utilizadas por las historias.

En la parte inferior derecha al final de nuestro carrusel, se encuentra un botón flotante con el símbolo “+” que despliega al usuario una opción que dice “crear historia” y con esa pulsación inicia el proceso de subida de su propia historia. Al presionar en este botón, se desplegará el explorador de archivos del usuario, donde el mismo deberá seleccionar archivos con extensión .pdf, .txt o .md.

En la esquina superior izquierda, se encontrará un icono que viene por defecto que al momento de tocarlo, se abre un modal donde el usuario podrá cambiar el icono que viene por defecto o su nombre de usuario si así lo deseara.

Figura 4

Diseño: Home



3.2.4. Codificación

3.2.4.1. Estándares de Codificación

Identificadores: Serán autoincrementales y numéricos.

Variables: Las variables tendrán nombres claros que reflejen su propósito, evitando abreviaturas ambiguas.

Funciones: Las funciones tendrán el nombre conforme a la acción que van a realizar.

Componentes: Los componentes *React Native* estarán organizados en carpetas según su funcionalidad, con archivos separados para estilos, lógica y pruebas cuando sea necesario. Se utilizarán *hooks* para manejar el estado y efectos secundarios.

Manejo de Estado: Se empleará un manejo de estado centralizado, para mantener la sincronización de datos entre el carrusel, el perfil y la subida de contenido.

Validaciones: Se implementarán validaciones tanto en el *frontend* como en el *backend* para asegurar la integridad de los datos ingresados por el usuario, especialmente en la edición de perfil y subida de contenido.

Bases: El *backend* seguirá utilizando *NestJS* con *Prisma ORM* para la interacción con la base de datos *PostgreSQL*. Las operaciones relacionadas con la subida de contenido y actualización de perfil se implementarán con métodos claros y seguros.

3.2.4.2. Código

Clase: *HomeScreen*

Campos principales:

- `storiesExample`: Array que contiene las tres historias de ejemplo para mostrar en el carrusel.
- `userProfile`: Objeto con la información del usuario (icono, correo, nombre).

Métodos principales:

- `showCarousel()`: Carga y muestra las historias de ejemplo en un carrusel con animación.

- `onUploadPress()`: Maneja el evento de presionar el botón “crear historia”, abriendo el formulario o pantalla para subir contenido.
- `onProfileEdit(updatedProfile)`: Recibe los datos modificados del usuario, valida y envía la actualización al backend.

Backend (NestJS):

Se implementarán los siguientes controladores y servicios:

- `HomeController`:
 - `getExampleStories()`: Devuelve las tres historias de ejemplo para el carrusel.
 - `uploadStory()`: Recibe el contenido subido por el usuario, inicia el proceso de generación de *embeddings* y almacenamiento en el vector store.
 - `updateUserProfile()`: Actualiza la información del usuario en la base de datos.

HomeService:

Métodos que implementan la lógica para manejar las operaciones mencionadas, incluyendo validaciones y llamadas a servicios externos (OpenAI, Supabase).

3.2.4.3. Prueba de Aceptación

El proceso de pruebas para la historia de usuario *Home* se enfocará en validar que las funcionalidades implementadas cumplan con los criterios de aceptación establecidos, garantizando una experiencia de usuario fluida, segura y sin errores.

Tabla 16

Prueba de Aceptación - Visualización del Carrusel en Home

Pruebas de Aceptación	
Nombre:	Prueba de Visualización del Carrusel
Descripción:	Verificar que el usuario pueda visualizar correctamente el carrusel con las tres historias de ejemplo y navegar entre ellas sin errores.
Condiciones de Ejecución	La aplicación debe estar funcionando en <i>Expo Go</i> en un dispositivo compatible.
Resultado Esperado	El usuario puede navegar entre las historias sin fallos.
Evaluación de la Prueba	Se considero exitosa debido a que el carrusel funciona correctamente y no presenta errores visuales o de navegación.

Nota. Fuente: Elaboración propia

Tabla 17

Prueba de Aceptación - Contenido y Embedding en Home

Pruebas de Aceptación	
Nombre:	Prueba de Subida de Contenido y <i>Embedding</i>

Descripción:	Verificar que al presionar el botón “crear historia” se inicie el proceso de subida de contenido, generación de <i>embeddings</i> y almacenamiento en el vector store.
Condiciones de Ejecución	Usuario autenticado en Home, <i>Backend</i> y servicios de <i>embedding</i> activos y conexión estable a internet
Resultado Esperado	Se abre el explorador de archivos del teléfono para seleccionar el archivo.
Evaluación de la Prueba	El contenido se procesa y almacena correctamente.

Nota. Fuente: Elaboración Propia

Tabla 18

Prueba de Aceptación - Visualización del Carrusel en Home

Pruebas de Aceptación

Nombre:	Prueba de Edición de Perfil
Descripción:	Verificar que el usuario pueda acceder a la pantalla de perfil, modificar su icono y nombre de usuario, y que los cambios se guarden correctamente.
Condiciones de Ejecución	Usuario autenticado y base de datos funcionando correctamente

Resultado Esperado	La pantalla de perfil se muestra correctamente.
Evaluación de la Prueba	Los cambios se validan y guardan sin errores

Nota. Fuente: Elaboración propia

3.2.4.4. Pruebas de Caja Blanca

Estas pruebas se enfocan en validar el correcto funcionamiento interno de los métodos y flujos implementados en el módulo *Home*, asegurando que cada componente y función opere según lo esperado.

Entradas	Resultado Esperado	Resultado
Usuario autenticado abre la pantalla Home	Se carga y muestra el carrusel con las tres historias de ejemplo sin errores	Se carga y muestra el carrusel con las tres historias de ejemplo sin errores.
Usuario navega por el carrusel desplazando su dedo	El carrusel avanza o retrocede correctamente entre las historias	El carrusel avanza o retrocede correctamente entre las historias.
Usuario presiona botón “+” para subir historia	Se abre el formulario o pantalla de subida sin errores	Se abre el formulario o pantalla de subida sin errores.
Usuario completa formulario con contenido válido y envía	El sistema bloquea el envío y muestra mensaje de validación	El método <i>onUploadPress()</i> valida, procesa y envía el contenido al <i>backend</i> para <i>embedding</i> correctamente.
Usuario intenta subir contenido vacío	El sistema bloquea el envío y muestra mensaje de validación	El sistema bloquea el envío y muestra mensaje de

Usuario accede a perfil y modifica datos válidos	El método <i>onProfileEdit()</i> valida y actualiza correctamente la información	validación si el contenido está vacío. El método <i>onProfileEdit()</i> valida y actualiza correctamente la información del perfil.
Usuario intenta guardar perfil con datos inválidos	El sistema muestra mensajes de error y no permite guardar cambios	El sistema muestra mensajes de error y no permite guardar cambios si los datos del perfil son inválidos.

Nota. Fuente Elaboración propia

3.3. Iteración 3 – Diccionario

3.3.1. Planificación

3.3.1.1. Historias de Usuario

Para esta historia de usuario, se buscará crear la pantalla de *Diccionario*, que funcionará como un glosario interactivo donde el usuario podrá consultar, buscar y gestionar las palabras utilizadas en la creación de sus historias. La pantalla mostrará un abecedario para facilitar la navegación por letras, y por defecto incluirá palabras de ejemplo asociadas a las tres historias mostradas en el *Home*. El usuario tendrá la opción de poder filtrar palabras a través de la barra de búsqueda, ver sus definiciones y las asociaciones que tiene esa palabra, así como editar o eliminar.

Tabla 19

Historia de Usuario - Diccionario

	Enunciado de la Historia	Criterio de Aceptación
--	---------------------------------	-------------------------------

Código de Historia	Rol	Funcionalidad	Razón	Escenario	Criterio	Contexto	Evento	Resultado
C1	Usuario	Consultar palabras del diccionario	Facilitar la búsqueda y consulta de palabras usadas en historias	1	Mostrar abecedario y lista de palabras	Usuario autenticado en pantalla Diccionario	Usuario navega por el carrusel	Usuario selecciona una letra o busca palabra
C2	Usuario	Ver definición y relación de palabra	Entender el significado y contexto de cada palabra	1	Mostrar definición y relaciones	Usuario en lista de palabras	Usuario presiona botón "crear historia" para subir historia	Usuario selecciona palabra
C3	Usuario	Editar o eliminar palabra	Permitir al usuario mantener actualizado su diccionario		Opciones de editar y eliminar visibles	Usuario en detalle de palabra	Usuario accede a perfil y modifica datos	Usuario edita o elimina palabra

Nota. Fuente: Elaboración propia

Para el desarrollo de esta historia de usuario se consideró una buena alternativa implementar un glosario además de contar con la opción de poder hacer la búsqueda de una palabra en específico a través de una barra de búsquedas. Además, también se consideró necesario poder realizar las acciones que están relacionadas a un CRUD.

3.3.1.2. *Plan de Iteración*

Las fechas que se prevén para el desarrollo del módulo de Presentación están contempladas para el 12 de junio del 2025 hasta el 17 de junio del 2025.

Tabla 20

Plan de Iteración - Diccionario

Plan de Iteración					
# de Iteración	Fecha Inicio	Fecha Fin	Numero de Historia	Nombre de Historia	Responsable
1	01/06/2025	05/06/2025	1	Presentación	Erick López
2	06/06/2025	11/06/2025	2	Home	Erick López
3	12/06/2025	17/06/2025	3	Diccionario	Erick López
4	18/06/2025	25/06/2025	4	Chat	Erick López

Nota. Fuente: Elaboración propia

3.3.1.3. *Criterios de Prueba de Aceptación*

Para la presente historia de usuario, se validará que el usuario pueda consultar, buscar, filtrar, editar y eliminar palabras del glosario, con el fin de poder verificar que todas estas funcionalidades cumplan con los criterios de aceptación.

Tabla 21

Criterio de Aceptación - Visualización Diccionario

	Visión de la Prueba		
Nombre	Visualización del Abecedario y Lista de Palabras	Código	E5

N. Historia	3
Iteración	3
Función Crítica	Si
Descripción	Verificar que el abecedario y la lista de palabras se muestren correctamente y permitan navegación y filtrado.
Categoría	Diccionario
Prerrequisito	Usuario autenticado, aplicación abierta en dispositivo compatible, <i>backend</i> activo.

Nota. Fuente: Elaboración propia

Tabla 22

Criterio de Aceptación - Definiciones Diccionario

	Visión de la Prueba		
Nombre	Consulta de Definición y Relaciones	Código	E6
N. Historia	3		
Iteración	3		
Función Crítica	Si		
Descripción	Verificar que al seleccionar una palabra se muestre su definición y palabras relacionadas.		
Categoría	Diccionario		
Prerrequisito	Usuario en pantalla de lista de palabras, conexión estable.		

Nota. Fuente: Elaboración propia

Tabla 23

Criterio de Aceptación - Edición de Palabras

	Visión de la Prueba		
Nombre	Edición y Eliminación de Palabras	Código	E7
N. Historia	3		
Iteración	3		
Función Crítica	Si		
Descripción	Verificar que el usuario pueda editar o eliminar palabras y que los cambios se reflejen correctamente en la base de datos.		
Categoría	Diccionario		
Prerrequisito	Usuario autenticado, permisos adecuados, <i>backend</i> y base de datos funcionando.		

Nota. Fuente: Elaboración propia

3.3.2. Diseño

El diseño de la pantalla *Diccionario* se compondrá por una barra de búsqueda en la parte superior de la pantalla. También se tiene pensando colocar una lista letras que de igual manera funcione como un criterio de filtro para ubicar las palabras que comiencen con una determinada letra en caso de así querer hacerlo.

Las palabras que cumplan con ese criterio de búsqueda aparecerán en un componente que contendrá la opción de switch para poder ver los detalles de esa palabra

Figura 5

Diseño: Diccionario



3.3.2.1. Tarjeta CRC

La tarjeta CRC para el módulo *Diccionario* se generó de manera clara para que cada una de las clases y componentes conozca sus responsabilidades y métodos.

Tabla 24*Tarjeta CRC - Diccionario*

Numero	T03	Nombre	Presentación
Descripción	Gestiona la visualización, búsqueda, edición y eliminación de palabras del diccionario		
Responsabilidades	Colaboradores	Métodos	
Mostrar abecedario y lista de palabras.	Usuario	<i>showAlphabet</i>	
Gestionar búsqueda	Usuario	<i>filterWords</i>	
Edición y eliminación.	Usuario	<i>showWordDetails</i> <i>editWord()</i> , <i>deleteWord()</i>	

Nota. Fuente: Elaboración propia

3.3.2.2. Prototipo

Para el prototipo de esta pantalla lo sé que busca es poder visualizar todas las palabras que tenga creadas el usuario además de también poder contar con los respectivos métodos de consultar, agregar, actualizar y eliminar una palabra.

También se implementará una lista que contenga todas las letras del diccionario con el propósito de poder filtrar según ese concepto de igual manera.

Figura 6

Prototipo: Diccionario



3.3.3. Codificación

3.3.3.1. Estándares de Codificación

Identificadores: El ID será auto incremental y será numérico.

Variables: Las variables tendrán nombres claros y específicos que reflejen su función, evitando abreviaturas ambiguas. Se priorizará el uso de constantes para valores fijos y variables con alcance limitado para evitar efectos secundarios.

Funciones: Los nombres de las funciones serán verbos en infinitivo que describan claramente la acción que realizan, como `loadWords()`, `filterWords()`, `editWord()`, `deleteWord()`. Cada función debe tener una única responsabilidad y ser modular.

Componentes: Los componentes *React Native* estarán organizados en carpetas según su funcionalidad, con archivos separados para estilos, lógica y pruebas cuando sea necesario. Se utilizarán *hooks* para manejar el estado y efectos secundarios.

Manejo de Estado: Se empleará un manejo de estado global, para mantener la sincronización de datos entre la lista de palabras, el filtro y la edición.

Validaciones: Se implementarán validaciones en *frontend* y *backend* para asegurar la integridad de los datos ingresados por el usuario, especialmente en la edición y eliminación de palabras.

Bases: El *backend* seguirá utilizando *NestJS* con *Prisma ORM* para la interacción con la base de datos PostgreSQL. Las operaciones relacionadas con la gestión del diccionario se implementarán con métodos claros y seguros.

3.3.3.2. Código

Clase: DictionaryScreen

Campos principales:

- `wordList`: Array que contiene las palabras del diccionario filtradas según la letra o búsqueda.
- `selectedWord`: Objeto con la información de la palabra seleccionada (definición, relaciones).

Métodos principales:

- `showAlphabet()`: Carga y muestra el abecedario para la navegación rápida.

- filterWords(letterOrQuery): Filtra la lista de palabras según la letra seleccionada o término de búsqueda.
- showWordDetails(wordId): Muestra la definición y palabras relacionadas de la palabra seleccionada.
- editWord(updatedWord): Valida y actualiza la información de la palabra en la base de datos.
- deleteWord(wordId): Elimina la palabra del diccionario tras confirmación del usuario.

Backend (NestJS):

Se implementarán los siguientes controladores y servicios:

- DictionaryController:
- getWordsByLetter(letter): Devuelve las palabras que comienzan con la letra especificada.
- searchWords(query): Devuelve palabras que coinciden con el término de búsqueda.
- getWordDetails(wordId): Devuelve la definición y relaciones de una palabra.
- updateWord(): Actualiza la información de una palabra.
- deleteWord(): Elimina una palabra del diccionario.

DictionaryService:

Métodos que implementan la lógica para manejar las operaciones mencionadas, incluyendo validaciones y acceso a la base de datos.

3.3.4. Pruebas

3.3.4.1. Prueba de Aceptación

El proceso de pruebas para la historia de usuario *Diccionario* se enfocará en validar que las funcionalidades implementadas cumplan con los criterios de aceptación establecidos, garantizando una experiencia de usuario fluida, segura y sin errores.

Tabla 25

Prueba de Aceptación - Visualización Diccionario

Pruebas de Aceptación	
Nombre:	Prueba de Visualización del Abecedario y Lista de Palabras
Descripción:	Verificar que el usuario pueda visualizar correctamente el abecedario y la lista de palabras filtradas y navegar entre ellas sin errores.
Condiciones de Ejecución	La aplicación debe estar expuesta en <i>Expo Go</i> y funcionando en un dispositivo compatible. Usuario autenticado y en la pantalla Diccionario
Resultado Esperado	El abecedario y la lista se muestran correctamente.
Evaluación de la Prueba	El usuario puede filtrar y navegar sin fallos.

Nota. Fuente: Elaboración propia

Tabla 26

Pruebas de Aceptación - Consulta Diccionario

Pruebas de Aceptación	
Nombre:	Prueba de Consulta de Definición y Relaciones

Descripción:	Verificar que al seleccionar una palabra se muestre su definición y palabras relacionadas correctamente.
Condiciones de Ejecución	La aplicación debe estar expuesta en <i>Expo Go</i> y funcionando en un dispositivo compatible. Usuario autenticado y en la pantalla Diccionario
Resultado Esperado	La definición y relaciones se muestran sin errores.
Evaluación de la Prueba	La prueba fue exitosa dado que la información se muestra correctamente y sin inconsistencias.

Nota. Fuente: Elaboración propia

3.3.4.2. *Pruebas de Caja Blanca*

Estas pruebas se enfocan en validar el correcto funcionamiento interno de los métodos y flujos implementados en el módulo *Diccionario*, asegurando que cada componente y función opere según lo esperado.

Tabla 27

Caja Blanca - Diccionario

Entradas	Resultado Esperado	Resultado
Usuario autenticado abre la pantalla Diccionario	Se carga y muestra el carrusel con las tres historias de ejemplo sin errores	OK
Usuario selecciona una letra del abecedario	La lista de palabras se filtra correctamente según la letra seleccionada	OK
Usuario realiza búsqueda por palabra	La lista se filtra correctamente según el término de búsqueda	OK
Usuario selecciona una palabra	Se muestra la definición y palabras relacionadas sin errores	OK

Usuario edita una palabra con datos válidos	El método editWord() valida y actualiza correctamente la palabra	OK
Usuario intenta editar palabra con datos inválidos	El sistema muestra mensajes de error y no permite guardar cambios	OK
Usuario elimina una palabra	El método deleteWord() elimina la palabra tras confirmación	OK
Usuario intenta eliminar palabra y cancela	La palabra no se elimina y la lista permanece intacta	

Nota. Fuente: Elaboración propia

3.4. Iteración 4 – Chat

3.4.1. Planificación

3.4.1.1. Historias de Usuario

Para esta historia de usuario, se buscará crear la pantalla de *Chat*, que permitirá al usuario interactuar mediante un chat tradicional con dos opciones principales: *Repasar* y *Consultar*. En la opción *Repasar*, el chat formulará preguntas relacionadas con los palacios mentales creados por el usuario para facilitar la revisión del contenido. En cambio, con la opción *Consultar*, el usuario podrá hacer preguntas en base a los documentos que ha subido para la creación de sus palacios mentales, obteniendo respuestas basadas en el contenido almacenado.

Tabla 28

Historia de Usuario - Chat

Código de Historia	Enunciado de la Historia				Criterio de Aceptación			
	Rol	Funcionalidad	Razón	Escenario	Criterio	Contexto	Evento	Resultado
D1	Usuario	Interactuar en chat con opción Repasar	Facilitar el repaso mediante preguntas sobre palacios mentales	1	Chat inicia preguntas de repaso	Usuario autenticado en pantalla Chat	Usuario selecciona opción Repasar	Chat formula preguntas relacionadas a palacios mentales
D2	Usuario	Interactuar en chat con opción Consultar	Permitir consultas sobre documentos subidos para palacios	1	Chat responde consultas basadas en documentos	Usuario autenticado en pantalla Chat	Usuario selecciona opción Consultar	Chat responde con información relevante de documentos

Nota. Fuente: Elaboración propia

Después de recopilar la información obtenida por la historia de usuario, se identificó la necesidad de ofrecer un canal de interacción como el de un chat, que permita al usuario repasar contenidos clave mediante preguntas dirigidas o simplemente brindarle la opción de consultar información específica basada en sus documentos subidos.

3.4.1.2. Plan de Iteración

Las fechas que se prevén para el desarrollo del módulo de Presentación están contempladas para el 12 de junio del 2025 hasta el 17 de junio del 2025.

Tabla 29*Plan de Iteración - Chat*

Plan de Iteración					
# de Iteración	Fecha Inicio	Fecha Fin	Numero de Historia	Nombre de Historia	Responsable
1	01/06/2025	05/06/2025	1	Presentación	Erick López
2	06/06/2025	11/06/2025	2	Home	Erick López
3	12/06/2025	17/06/2025	3	Diccionario	Erick López
4	18/06/2025	25/06/2025	4	Chat	Erick López

Nota. Fuente: Elaboración propia**3.4.1.3. Criterios de Prueba Aceptación**

Para esta historia de usuario se tiene pensando validar cada una de las opciones con los siguientes criterios que se encuentran en las Tablas: 30 y 31.

Tabla 30*Criterio de Aceptación - Repasar en Chat*

Nombre	Visión de la Prueba		
	Interacción en Opción Repasar	Código	E8
N. Historia	4		
Iteración	4		
Función Crítica	Si		
Descripción	Verificar que el chat formule preguntas relacionadas con los palacios mentales del usuario.		
Categoría	Chat		
Prerrequisito	Usuario autenticado, palacios mentales creados y disponibles, backend y servicios activos.		

Nota. Fuente: Elaboración propia

Tabla 31

Criterio de Aceptación - Consultar en Chat

	Visión de la Prueba		
Nombre	Interacción en Opción Consultar	Código	E9
N. Historia	4		
Iteración	4		
Función Crítica	Si		
Descripción	Verificar que el chat responda consultas basadas en los documentos subidos para palacios.		
Categoría	Chat		
Prerrequisito	Usuario autenticado, documentos subidos y procesados, <i>backend</i> y servicios activos.		

Nota. Fuente: Elaboración propia

3.4.2. Diseño

Para el diseño de esta pantalla se piensa tener un *scroll* que permita deslizar la pantalla tanto como para bajar o subir el contenido conforme a la extensión de los mensajes que se vayan escribiendo dentro de este bloque.

Se piensa integrar un recuadro en donde el usuario podrá ingresar su consulta sobre algún tema

Figura 7

Diseño: Chat



3.4.2.1. Tarjeta CRC

La tarjeta CRC para este módulo de Chat se definió con el propósito de que cada uno tengo su método correspondiente.

Tabla 32

Tarjeta CRC - Chat

Numero	T03	Nombre	Presentación
Descripción	Gestiona la interacción del usuario con las opciones de repaso y consulta en el chat		
Responsabilidades	Colaboradores	Métodos	
Manejar selección de opción.	Usuario	<i>selectOption</i>	
Enviar y recibir mensajes,	Usuario	<i>sendMessage</i> <i>receiveResponse</i>	

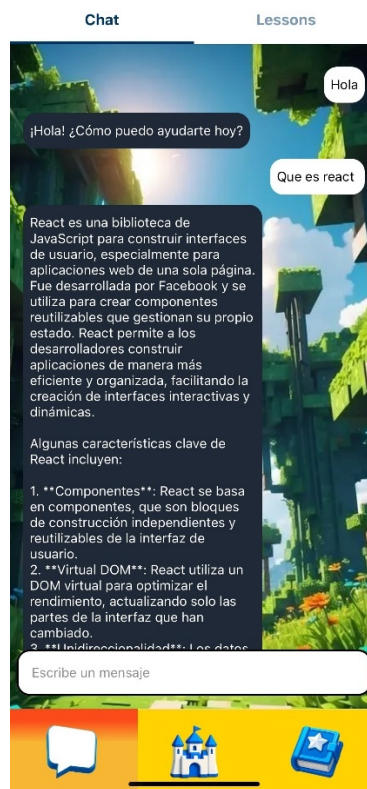
Nota. Fuente: Elaboración propia

3.4.2.2. *Prototipo*

Para el prototipo de la pantalla Chat lo que se busco fue que el usuario tenga la capacidad de hacer preguntas sobre el contenido que ha subido para la creación de sus historias. Además también de tener la capacidad de poder repasar sobre un contenido de una historia en específico.

Figura 8

Prototipo: Chat



3.4.3. *Codificación*

3.4.3.1. *Estándares de Codificación*

Identificadores: El ID será auto incremental y será numérico

Variables: Para las variables se consideró que reflejen su función, evitando abreviaturas ambiguas. Se priorizará el uso de constantes para valores fijos y variables con alcance limitado para evitar efectos secundarios.

Funciones:

Los nombres de las funciones serán verbos en infinitivo que describan claramente la acción que realizan, como `selectOption()`, `sendMessage()`, `receiveResponse()`, `processReview()`, `processQuery()`. Cada función debe tener una única responsabilidad y ser modular.

Componentes:

Los componentes *React Native* estarán organizados en carpetas según su funcionalidad, con archivos separados para estilos, lógica y pruebas cuando sea necesario. Se utilizarán *hooks* para manejar el estado y efectos secundarios.

Manejo de Estado:

Se empleará un manejo de estado global para mantener la sincronización de datos entre la selección de opciones, mensajes y respuestas.

Validaciones:

Se implementarán validaciones en *frontend* y *backend* para asegurar la integridad de los datos ingresados por el usuario, especialmente en la entrada de mensajes y selección de opciones.

3.4.3.2. Código

Clase: ChatScreen

Campos principales:

- chatMessages: Array que contiene los mensajes intercambiados en la conversación.
- selectedOption: Variable que indica si el usuario eligió *Repasar* o *Consultar*.

Métodos principales:

- selectOption(option): Permite al usuario elegir entre las opciones *Repasar* o *Consultar*.
- sendMessage(message): Envía el mensaje del usuario al backend o modelo de lenguaje.
- receiveResponse(): Recibe y muestra la respuesta generada por el sistema.
- processReview(): Gestiona la lógica para formular preguntas relacionadas con los palacios mentales en la opción *Repasar*.
- processQuery(): Gestiona la lógica para responder consultas basadas en documentos en la opción *Consultar*.

Backend (NestJS):

Se implementarán los siguientes controladores y servicios:

- ChatController:
- startReviewSession(userId): Inicia una sesión de repaso con preguntas basadas en palacios mentales.
- handleQuery(userId, query): Procesa consultas del usuario basadas en documentos subidos.

ChatService:

Métodos que implementan la lógica para manejar las operaciones mencionadas, incluyendo integración con modelos de lenguaje y acceso a la base de datos.

3.4.4. Pruebas

3.4.4.1. Pruebas de Aceptación

El proceso de pruebas para la historia de usuario *Chat* se enfocará en validar que las funcionalidades implementadas cumplan con los criterios de aceptación establecidos, garantizando una experiencia de usuario fluida, segura y sin errores.

Tabla 33

Prueba de Aceptación - Repasar en Chat

Pruebas de Aceptación	
Nombre:	Prueba de Interacción en Opción Repasar
Descripción:	Verificar que el chat formule preguntas relacionadas con los palacios mentales y permita respuestas del usuario.
Condiciones de Ejecución	La aplicación debe estar expuesta en <i>Expo Go</i> y funcionando en un dispositivo compatible. Usuario autenticado y en la pantalla Chat
Resultado Esperado	El chat inicia preguntas de repaso.
Evaluación de la Prueba	El usuario pudo responder y avanzar sin errores.

Nota. Fuente: Elaboración propia

Tabla 34

Prueba de Aceptación - Consultar en Chat

Pruebas de Aceptación

Nombre:	Prueba de Interacción en Opción Repasar
Descripción:	Verificar que el chat responda consultas basadas en documentos subidos y muestre respuestas relevantes.
Condiciones de Ejecución	La aplicación debe estar expuesta en <i>Expo Go</i> y funcionando en un dispositivo compatible. Usuario autenticado y en la pantalla Chat
Resultado Esperado	El chat inicia preguntas de repaso.
Evaluación de la Prueba	Se considero exitosa la prueba dado que las respuestas son correctas y la experiencia es estable.

Nota. Fuente: Elaboración propia

3.4.4.2. *Pruebas de Caja Blanca*

Estas pruebas se enfocan en validar el correcto funcionamiento interno de los métodos y flujos implementados en el módulo *Chat*, asegurando que cada componente y función opere según lo esperado.

Tabla 35

Prueba de Caja Blanca - Chat

Entradas	Resultado Esperado	Resultado
Usuario autenticado abre la pantalla Chat	Se muestra la pantalla con opciones <i>Repasar</i> y <i>Consultar</i> sin errores	OK
Usuario selecciona opción <i>Repasar</i>	El chat inicia preguntas relacionadas con palacios mentales	OK
Usuario responde preguntas de repaso	El sistema procesa respuestas y avanza a la siguiente pregunta sin errores	OK
Usuario selecciona opción <i>Consultar</i>	El chat permite ingresar consultas y muestra respuestas basadas en documentos	OK

Usuario envía consulta válida	El método processQuery() procesa y responde correctamente	OK
Usuario envía consulta invalida	El sistema muestra mensaje de error y no procesa la consulta	OK

Nota. Fuente: Elaboración propia

3.5. Modelo Conceptual de la Aplicación

La aplicación está diseñada para facilitar la creación de palacios mentales y consulta de los mismos mediante una experiencia interactiva y personalizada. Su modelo conceptual se basa en la integración de varios módulos funcionales que interactúan con una base de datos relacional y un sistema avanzado de procesamiento semántico mediante *embeddings*.

Entidades Principales

Usuario: Representa a la persona que utiliza la aplicación. Cada usuario tiene un perfil con información personal (nombre, correo, icono) y está asociado a múltiples diccionarios, historias y documentos. La autenticación se gestiona mediante un servicio externo (*Clerk*) que garantiza seguridad y flexibilidad.

Diccionario: Es un glosario personalizado para cada usuario que contiene palabras clave utilizadas en sus historias. Cada palabra está asociada a definiciones y relaciones con otras palabras, permitiendo una navegación y gestión eficiente.

Historia: Contiene narrativas creadas o subidas por el usuario, vinculadas a personajes y lugares específicos. Las historias están relacionadas con diccionarios y son la base para la generación de palacios mentales.

Documento: Representan el contenido de texto que el usuario puede subir. Una vez el documento se encuentre subido los elementos de ese contenido son procesados para extraer información relevante y generar *embeddings* que alimentan el sistema de búsqueda semántica.

Palacio Mental: Técnica mnemotécnica generada automáticamente a partir de los documentos subidos por el usuario facilitando de este modo la creación de estas historias.

Chat: Interfaz de interacción donde el usuario puede interactuar con un modelo de lenguaje para consultarle información sobre sus historias.

Relaciones y Procesos

Los **usuarios** pueden crear y gestionar múltiples **diccionarios**, **historias**, **documentos** y **notas**.

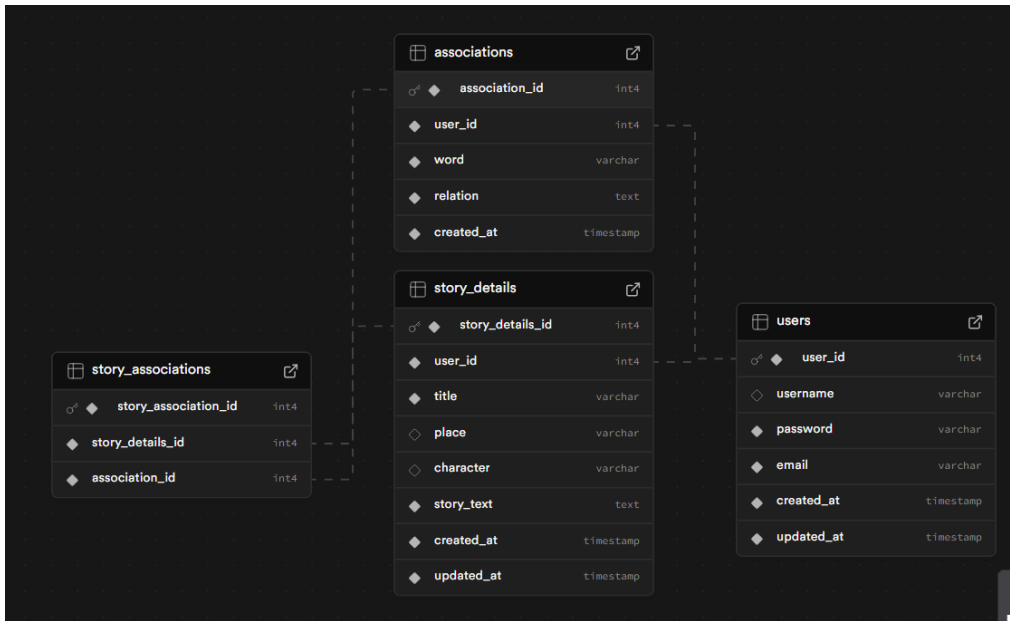
Cada **diccionario** contiene muchas **palabras**, que pueden ser consultadas, editadas o eliminadas por el usuario.

Las **historias** están vinculadas a **diccionarios** y contienen información contextual como personajes y lugares.

Los **documentos** y **notas** subidos por el usuario son procesados mediante un flujo automatizado que genera *embeddings* utilizando modelos de lenguaje (OpenAI GPT-4.1-mini) y los almacena en un vector store (*Supabase* con *pgvector*).

Figura 9

Esquema de Base de Datos



Nota. Esta figura presenta el esquema de la base de datos, detallando las tablas y las relaciones entre ellas.

El flujo utiliza estos *embeddings* porque es con ellos que de esta forma comunica cada uno de estos vectores en un número de *batch* determinado para el modelo de lenguaje, con la instrucción de que recupere los puntos más importantes de ese vector.

El chat permite dos modos de interacción: *Repasar*, en este modo el modelo de lenguaje formula preguntas basadas en el contenido que utilizo para generar los palacios mentales mientras que en el modo *Consultar*, responde preguntas basadas en los documentos y notas del usuario.

3.6. Arquitectura Funcional

El flujo de la aplicación comienza en la interfaz desarrollada con *React Native*. Al momento abrir la aplicación, lo primero que encuentra el usuario es una pantalla de presentación, que sirve como bienvenida y punto de partida. Desde aquí, el usuario al momento de llegar a la última definición que comparte la aplicación, el usuario se redirigido hacia la pantalla de registro.

Para este punto es donde ocurre la primera interacción con un servicio externo fundamental el cual es *Clerk*, su papel es encargarse de la autenticación de los usuarios. *Clerk* actúa como un servicio de autenticación, gestionando tanto el registro de nuevas cuentas como el inicio de sesión de usuarios existentes.

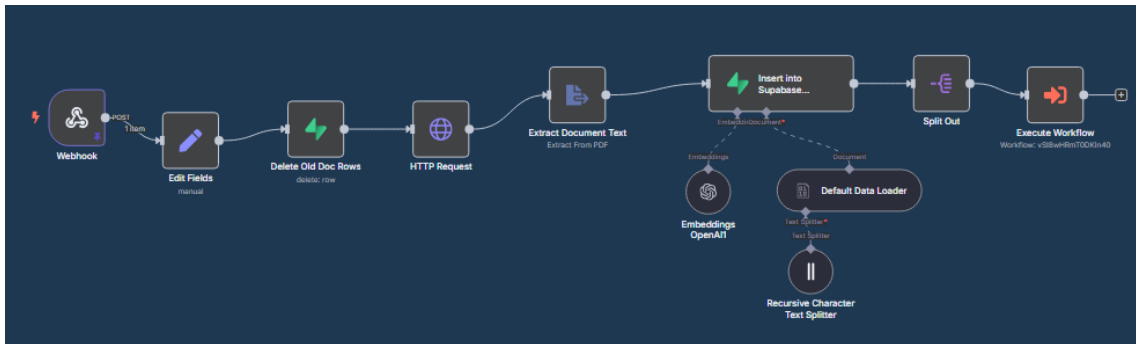
Una vez que el usuario ha iniciado sesión es redirigido hacia la pantalla principal que es el Home. En esta sección es donde el usuario tiene la posibilidad de subir el contenido de sus clases y apuntes.

Para este punto, entra en acción el *backend* de la aplicación. Cuando el usuario selecciona un archivo para subir, la aplicación realiza una llamada a nuestro *backend*, el cual está desarrollado con *NestJS*. En este mismo *backend es* en donde utilizamos la dependencia de *Multer*, una librería especializada que permite recibir y almacenar archivos de manera eficiente y segura en nuestro servidor.

Una vez que el archivo se ha subido correctamente al servidor, el *backend* realiza una llamada a un *webhook* configurado en nuestro sistema de automatización *n8n*, que está alojado en nuestro propio *VPS*.

Figura 10

Flujo de Embedding



Nota. El gráfico representa el flujo que sigue el MCP para la creación de vectores mediante *embeddings*.

La función principal de este primer flujo dentro de *n8n* es recibir la ruta del archivo que ha sido alojado previamente en nuestro *backend*. Antes de proceder con la descarga y la conversión del archivo en vectores, el flujo antes llama a un nodo de *Supabase* que a través del id del documento que se está subiendo revisa si existe para que elimine los registros que actualmente se encuentran en nuestra base de datos vectorial para volver a generarlos más adelante en los siguientes nodos, esto se hace con el propósito de no llenar nuestro vector store con información redundante cada vez que se trate del mismo archivo.

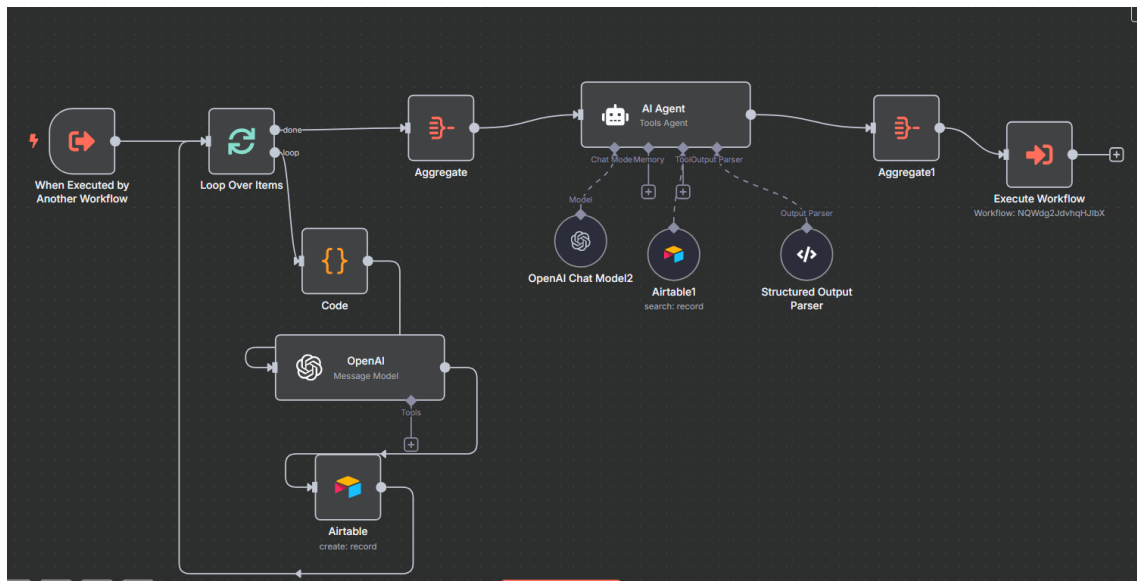
Una vez que se haya validado si el archivo es nuevo o no, el siguiente paso es conectarse a nuestro *vector store* en *Supabase*. Para ello, primero descargamos el archivo y lo procesamos, convirtiendo su contenido en vectores (*embeddings*). Utilizamos la API de OpenAI, específicamente el modelo *text-embedding-3-small*, para generar los *embeddings*. Es importante

mencionar que la configuración que se realizó para este nodo en específico fue definir tanto el número de fragmentos (*chunks*), como el grado de solapamiento (*overlap*) entre ellos.

Una vez generados los vectores, estos se insertan en el vector store, quedando el contenido listo para ser consultado y utilizado.

Figura 11

Flujo de Extracción de Ideas Principales



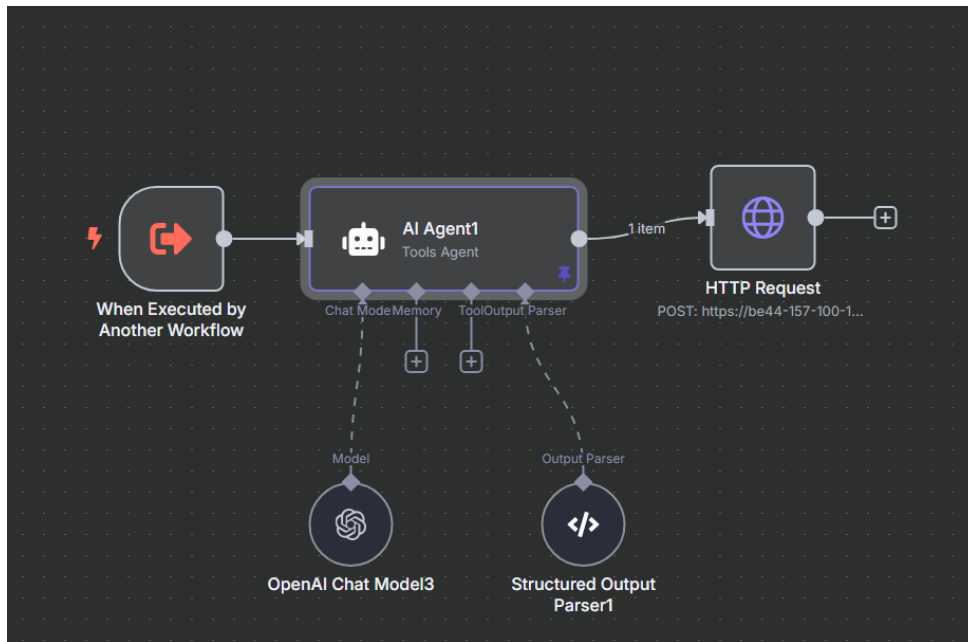
Nota. El gráfico representa el flujo de creación e identificación de ideas principales, seguido por la consolidación de dichas ideas en una única lista resumen.

Tras la finalización del primer flujo el ultimo nodo por ejecutarse en ese flujo llama a u otro flujo de actividades que recibe todos los vectores transformados e insertados en nuestro vector store. Para poder enviar la estos *chunks* de una manera un poco estructurada se utilizo un nodo de código en donde se implemento un poquito de *javascript*. Después de ese nodo nuestro modelo tiene la tarea de encontrar los puntos más importantes que tiene cada vector con el fin de

poder consolidar todos esos puntos importantes en una única lista de ideas principales que será almacenada en nuestra tabla de *Airtable*

Figura 12

Agente de Creación de Historias



Nota. El gráfico representa al agente y los nodos conectados que le permiten enviar la información de la historia de manera estructurada en formato JSON.

Finalmente, cuando ya se ha consolidado y procesado toda la información de los pasos anteriores, el sistema hace la llamada a un nuevo *workflow*. El sentido de este nuevo flujo tiene como objetivo principal la creación de la historia para el palacio mental asociando palabras a conceptos complejos o resúmenes como lo es en este caso para que de esta manera sea más fáciles de recordar.

Para lograr esto, se utiliza nuevamente un agente basado en el modelo de lenguaje **GPT-4.1-mini**. El agente toma la información resumida y estructurada, y genera tanto la narrativa del palacio mental como una lista de palabras clave que servirán para asociar los conceptos principales de la historia.

Una vez generada esta estructura, la información se prepara en un formato adecuado y se realiza una petición **POST** a nuestro servidor. Nuestro *backend*, al recibir esta solicitud, revisa y valida la respuesta utilizando el *ORM Prisma*, luego de una validación exitosa de la forma en que fueron enviados estos datos el *backend* ingresa finalmente todos estos datos en la base de datos, asegurando que la historia, las palabras clave y la estructura del palacio mental queden correctamente almacenadas y listas para ser consultadas o utilizadas por el usuario en el futuro.

3.7. Definición de los Módulos Principales

3.7.1. Personalización de la aplicación con IA

La aplicación utiliza inteligencia artificial para ofrecer una experiencia personalizada a cada usuario. A través del análisis de los contenidos subidos, las historias creadas y las interacciones en el chat, el sistema adapta las respuestas y sugerencias según el perfil y necesidades individuales. Para ello, se integra la API de OpenAI, que permite utilizar el modelo de lenguaje GPT-4.1-mini para procesar la información y generar palacios mentales, resúmenes y preguntas específicas. Asegurando que cada usuario reciba contenido relevante a la información que almacena.

3.7.2. Implementación de la Repetición Espaciada

La forma en cómo se planea implementar esta técnica se encuentra en un flujo automatizado que registra las fechas en que el usuario realiza sesiones de repaso en el chat. Un

agente especializado monitorea estas interacciones y almacena los registros en *Airtable*, donde se calcula el momento óptimo para la siguiente lección. Cuando se acerca la fecha programada, el sistema envía automáticamente un correo electrónico al usuario recordándole realizar su siguiente sesión de estudio. Este mecanismo garantiza que el aprendizaje se refuerce en intervalos adecuados.

3.7.3. Creación de los Palacios Mentales

La creación de palacios mentales es una funcionalidad central de la aplicación en donde a partir del contenido procesado y resumido de las historias y documentos del usuario, un agente basado en GPT-4.1-mini genera narrativas estructuradas y listas de palabras clave que representan los elementos principales. Estas asociaciones visuales y conceptuales se almacenan en la base de datos, permitiendo al usuario acceder a un recurso personalizado que potencia la retención y el recuerdo mediante la técnica del palacio mental.

4. Pruebas y Evaluación

4.1. Definición del entorno de prueba controlado

Para la realización de las pruebas, se utilizó un entorno controlado que permitió evaluar el funcionamiento de la aplicación en condiciones representativas y reproducibles.

Hardware utilizado: Las pruebas se llevaron a cabo en varios dispositivos móviles para asegurar la compatibilidad y el desempeño en diferentes plataformas. Se utilizaron un iPhone 7, un iPhone 14 y un *Honor Infinix*, cubriendo así tanto dispositivos de gama media como alta y diferentes sistemas operativos.

Software y herramientas: Para las pruebas es necesario utilizar *Expo Go*, esto lo hacemos porque es el puente que facilita la ejecución de nuestra aplicación en los dispositivos mencionados. En aspectos como el software para el tema del *backend* utilizamos *NestJS* que se ejecuta localmente en el equipo del desarrollador durante las pruebas. Por otro lado, el sistema de automatización n8n se encuentra alojado y activo en un servidor VPS, garantizando la disponibilidad de los flujos de trabajo necesarios para el procesamiento de datos.

Estado inicial de la aplicación: Para que la aplicación funcione de manera correcta es primordial primero conectar el celular a *Expo Go* a través del código qr que se genera al momento de correr la aplicación. Pero antes de ello es primordial recordar que para expo funcione de la mejor manera, tanto el dispositivo móvil como el ordenador en donde este corriendo la aplicación deben encontrarse en la misma red.

4.2. Pruebas Funcionales

La razón de las pruebas funcionales fue verificar que cada una de las funcionalidades implementadas en la aplicación cumpliera con los requisitos especificados y funcionara correctamente.

Alcance de las pruebas:

Se evaluaron los principales módulos y flujos de la aplicación, incluyendo:

- Registro y autenticación de usuarios mediante *Clerk*.
- Visualización y navegación en las pantallas de Presentación y Home, incluyendo el carrusel de historias de ejemplo.
- Subida de contenido y procesamiento de *embeddings* a través de *n8n*.
- Gestión del diccionario: búsqueda, filtrado, edición y eliminación de palabras.
- Interacción en el chat con las opciones de *Repasar* y *Consultar*.
- Edición y actualización del perfil de usuario.

Metodología:

Las pruebas se realizaron de forma manual en los dispositivos mencionados, siguiendo casos de prueba previamente definidos que cubrían escenarios positivos y negativos. Se verificó la correcta integración entre *frontend*, *backend* y servicios externos, así como la respuesta del sistema ante entradas válidas e inválidas.

Resultados esperados:

Cada funcionalidad debía ejecutarse sin errores, respetando los criterios de aceptación definidos en las historias de usuario. Se prestó especial atención a la experiencia del usuario, tiempos de respuesta y manejo adecuado de errores y validaciones.

4.3. Evaluación de la experiencia del Usuario

Propósito de las pruebas de usabilidad: El objetivo principal fue evaluar la facilidad de uso, efectividad y satisfacción de los usuarios al interactuar con la aplicación, la cual está diseñada para facilitar y promover la técnica Loci.

Contexto de la aplicación: La aplicación permite a los usuarios subir y organizar contenido educativo, crear palacios mentales personalizados, interactuar mediante un chat inteligente y gestionar diccionarios de palabras clave además que esta asistida por inteligencia artificial.

4.3.1. Metodología

Diseño de las pruebas: Se llevaron a cabo pruebas de usabilidad presenciales con un grupo de 20 estudiantes en un entorno controlado.

Rol: Un único facilitador y monitor, quien explicó el contexto, objetivos y tareas a realizar, respondió preguntas durante las sesiones, registró comportamientos, y envió el enlace para que puedan acceder a los cuestionarios pre-test y post-test.

4.3.2. Perfil de los usuarios:

Los participantes seleccionados para las pruebas fueron personas que, aunque no necesariamente tienen conocimientos previos sobre técnicas o conceptos de memorización, están familiarizadas con el uso básico de dispositivos móviles y aplicaciones. Se consideró un rango de edad hasta 35 años. Además, se priorizó la inclusión de usuarios que habitualmente toman apuntes, guardan o gestionan gran cantidad de información.

4.3.3. Estructura de las pruebas

1. Introducción al contexto y funcionalidades principales de la aplicación.
2. Aplicación de un cuestionario Pre-Test para recopilar datos demográficos, experiencia previa con tecnologías educativas y conocimientos sobre técnicas mnemotécnicas.

Realización de tareas específicas dentro de la aplicación, tales como:

- Registro y acceso a la aplicación.
- Navegación por las pantallas de presentación y Home, incluyendo la revisión de historias de ejemplo.
- Subida y procesamiento de contenido educativo.
- Uso del diccionario para búsqueda y gestión de palabras.
- Interacción con el chat en modos *Repasar* y *Consultar*.
- Edición del perfil de usuario.

Se estableció un tiempo límite para cada actividad, buscando medir eficiencia y detectar posibles dificultades.

3. Aplicación de un cuestionario Post-Test para evaluar la satisfacción, usabilidad y percepción general tras la interacción con la aplicación.
4. Cierre con espacio para comentarios, sugerencias y entrega de incentivos a los participantes.

4.3.4. Métodos utilizados

Observación directa de los participantes durante la realización de las tareas, con registro de tiempos y comportamientos por parte de los monitores.

Cuestionarios estructurados Pre-Test y Post-Test para obtener datos cuantitativos y cualitativos sobre la experiencia del usuario.

Análisis de resultados para identificar fortalezas, debilidades y oportunidades de mejora en la aplicación.

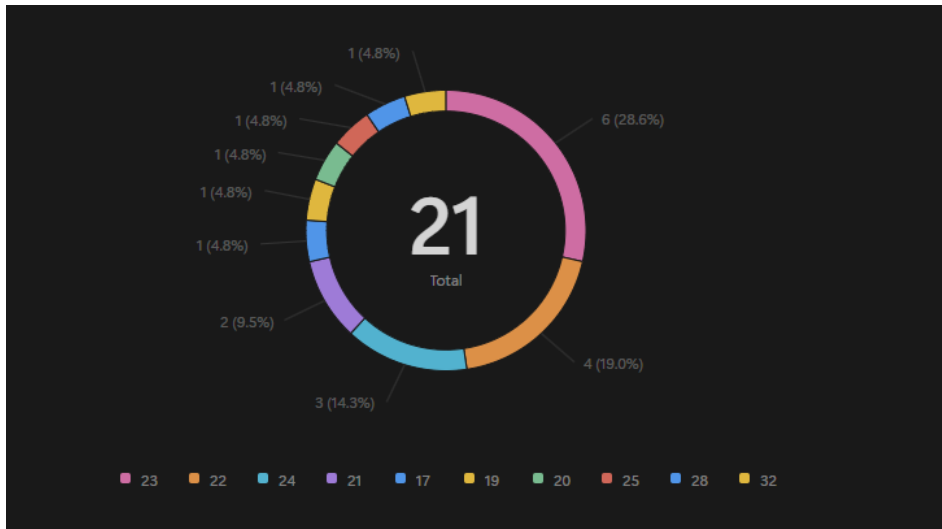
4.4. Análisis de Resultados

4.4.1. Resultados Pre-Test

1. ¿Cuál es tu edad?

Figura 13

Resultados de la pregunta 1 Pre-Test



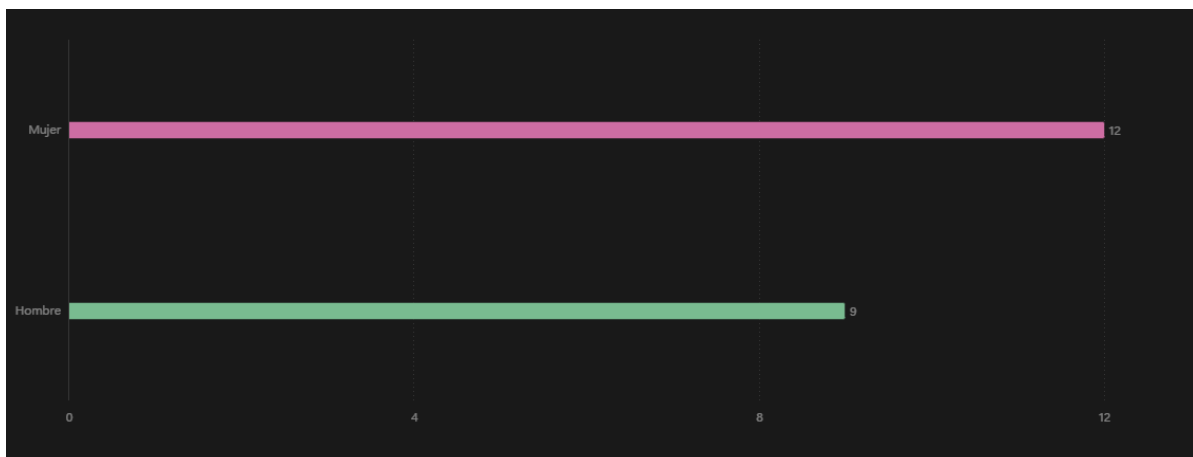
Nota. El grafico representa el porcentaje de edades que respondieron a la encuesta.

La mayoría de los participantes en esta encuesta pre-test tienen entre 21 y 24 años, con un notable predominio de la edad de 23 años. Esto sugiere que el público objetivo principal de la encuesta podría ser gente joven-adulta.

2. ¿Cuál es tu genero?

Figura 14

Resultados de pregunta 2 Post-Test



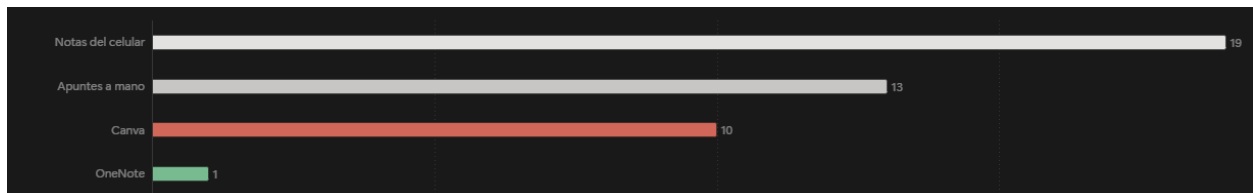
Nota. El grafico representa el número total de hombres y mujeres que respondieron a la encuesta.

El 57.1% de los participantes son mujeres (12 de 21), mientras que el 42.9% son hombres (9 de 21), indicando una mayor representación femenina en la muestra.

3. ¿Qué aplicaciones o herramientas utilizas para crear y organizar tus apuntes?

Figura 15

Resultados de la pregunta 3 Post-Test



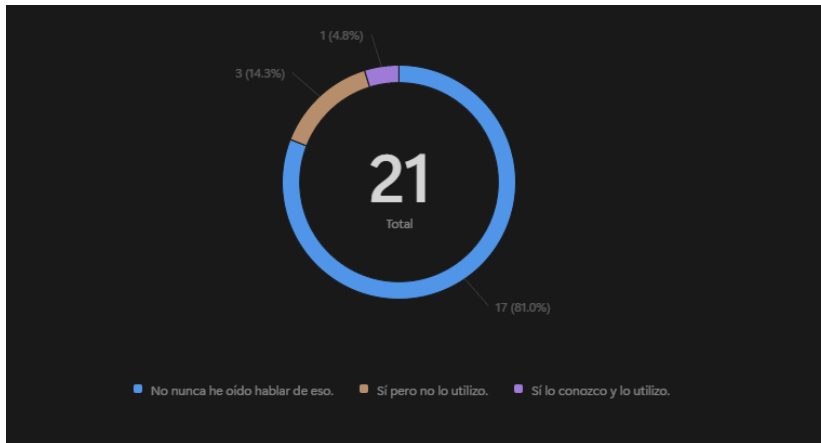
Nota. El grafico representa las aplicaciones que más utilizan las personas para realizar sus apuntes.

Los participantes prefieren métodos de toma de notas tradicionales o accesibles como las "Notas del celular" (19 usuarios) y "Apuntes a mano" (13 usuarios), mientras que herramientas digitales más especializadas como "Canva" (10 usuarios) y "OneNote" (1 usuario) son considerablemente menos populares.

4. ¿Has oído hablar del concepto de “palacio mental”?

Figura 16

Resultados de la pregunta 4 Pre-Test



Nota. El grafico representa las personas que conocen el término palacio mental

Una abrumadora mayoría de los participantes (81.0%) nunca ha oído hablar del término "Palacio Mental", indicando una baja familiaridad general con este concepto entre la muestra encuestada.

5. ¿Conoces en qué consiste la técnica de repetición espaciada?

Figura 17

Resultados de la pregunta 5 Pre-Test



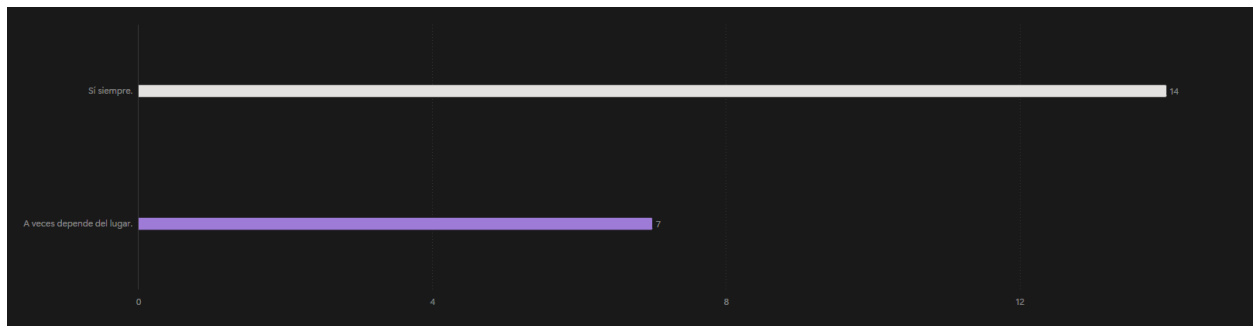
Nota. El grafico representa el número total de personas que conocen el término “Repetición Espaciada”

La mayoría de los encuestados (71.4%) no conocen la “Repetición Espaciada”, aunque un 23.8% la conoce, pero no la aplica, y solo un 4.8% la utiliza regularmente.

6. ¿Tienes acceso a internet en todo momento para consultar tus apuntes?

Figura 18

Resultados de la pregunta 6 Pre-Test



Nota. El grafico representa las personas que mantienen una conexión a internet de manera constante y las que dependen del lugar en donde se encuentren.

La mayoría de los encuestados (14 de 21) afirman tener conexión a internet siempre, mientras que una proporción menor (7 de 21) indica que su conexión depende del lugar.

7. ¿Utilizas inteligencia artificial para crear o mejorar tus apuntes?

Figura 19

Resultados de la pregunta 7 Pre-Test



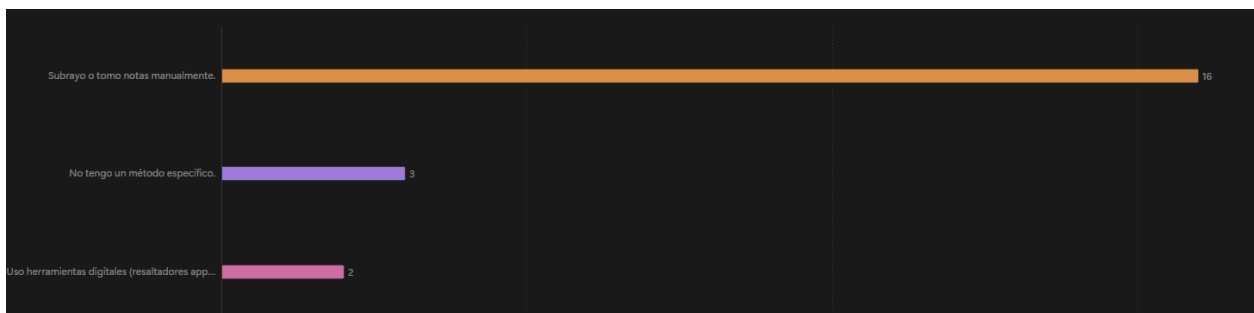
Nota. Este grafico representa a las personas que utiliza Inteligencia Artificial para crear o mejorar sus apuntes

Casi la mitad de los encuestados (47.6%) utilizan IA ocasionalmente (“A veces cuando lo necesito”), un tercio (33.3%) la usa frecuentemente, y un 19% nunca la ha empleado para este fin.

8. ¿Cómo identificas las ideas principales al crear tus apuntes?

Figura 20

Resultados de la pregunta 8 Pre-Test



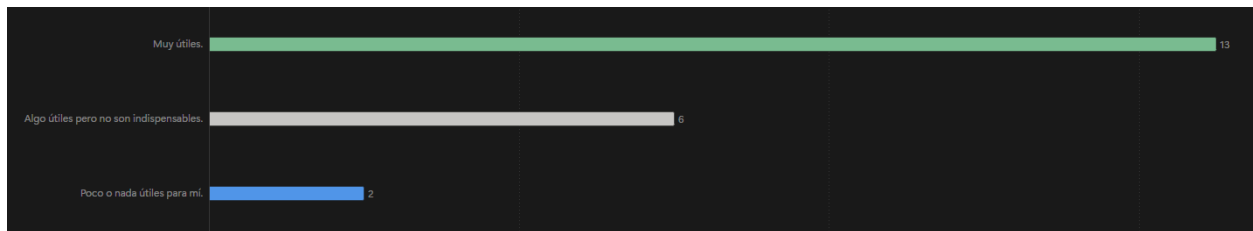
Nota. El grafico representa las formas en las que el usuario identifica las ideas principales de un texto.

La gran mayoría de los encuestados (16 de 21) prefieren métodos manuales como subrayar o tomar notas, mientras que un número considerablemente menor no tiene un método específico (3 de 21) o usa herramientas digitales (2 de 21).

9. ¿Qué tan útil te parecen las herramientas digitales para crear apuntes?

Figura 21

Resultados de pregunta 9 Pre-Test



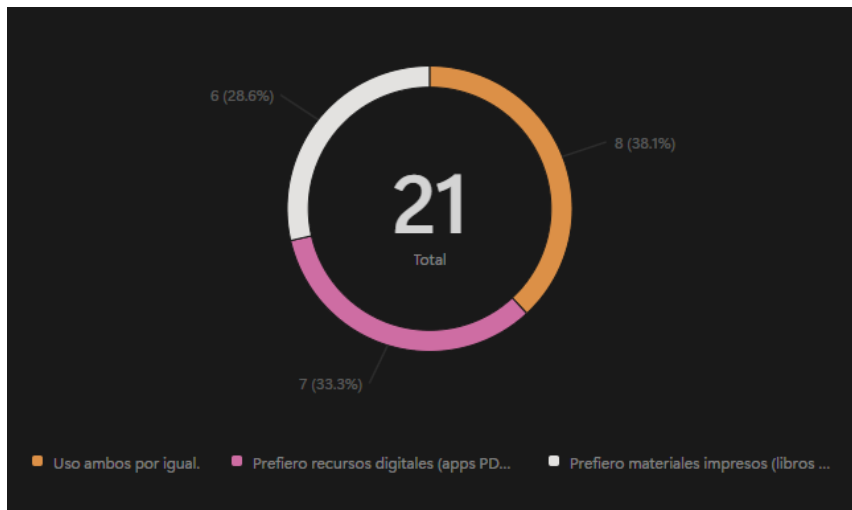
Nota. Este grafico nos indican el numero de usuarios que consideran útil utilizar Inteligencia Artificial para la toma de apuntes.

La mayoría de los usuarios (13 de 21) consideran que la inteligencia artificial es “Muy útil” para tomar apuntes, mientras que una parte significativa (6 de 21) la ve “Algo útil pero no indispensable”, y solo una minoría (2 de 21) la considera “Poco o nada útil”.

10. ¿Prefieres estudiar con recursos digitales o con materiales impresos?

Figura 22

Resultados de pregunta 10 Pre-Test



Nota. Este grafico muestra si las personas prefieren un recurso digital que en recurso físico

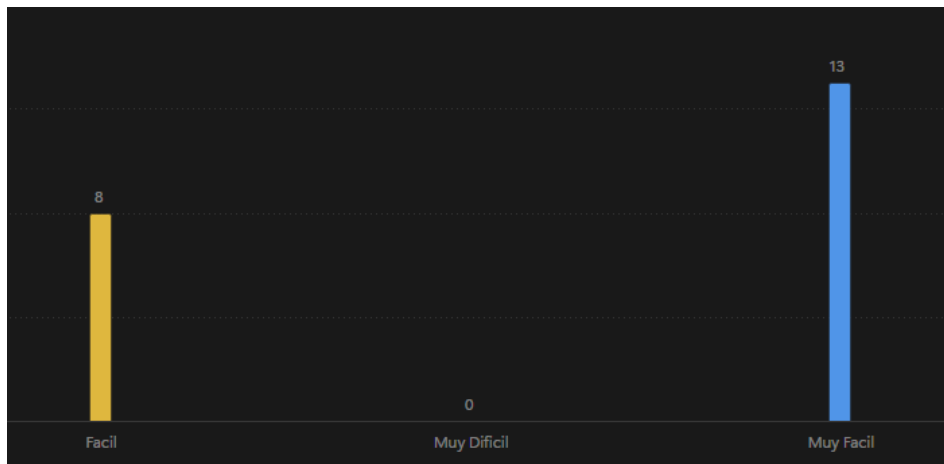
La preferencia está dividida, con un 38.1% que usa ambos por igual, un 33.3% que prefiere recursos digitales, y un 28.6% que aún se inclina por materiales impresos.

4.4.2. Resultados Post-Test

1. **¿Cómo calificarías la facilidad para navegar por las diferentes secciones de la aplicación (Presentación, Home, Diccionario y Chat)?**

Figura 23

Resultados de pregunta 1 Post-Test



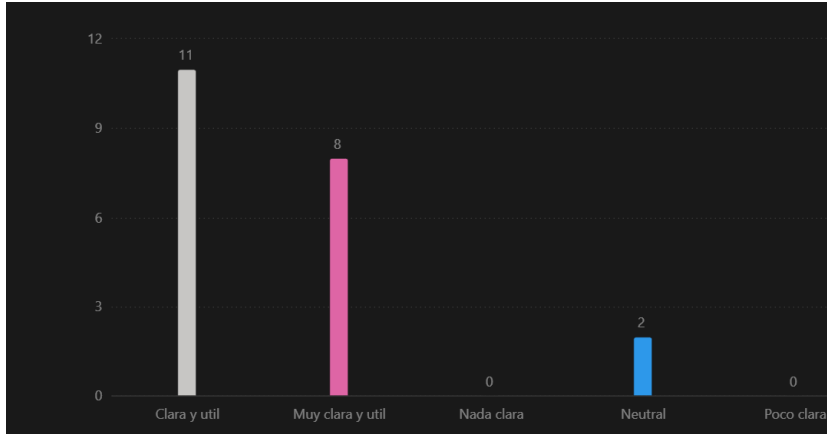
Nota. El grafico representa la percepción de los participantes sobre la facilidad de navegación dentro de la aplicación, específicamente en secciones como la Presentación, Home, Diccionario y Chat.

La mayoría de los usuarios (13 de 21 personas) calificaron la navegación como “muy fácil”, mientras que las 8 personas restantes la consideraron simplemente “fácil”, lo que refuerza la idea de que, en general, los usuarios no encontraron obstáculos importantes al interactuar con las distintas secciones.

2. ¿Consideras que la presentación de los palacios mentales y resúmenes es clara?

Figura 24

Resultados de pregunta 2 Post-Test



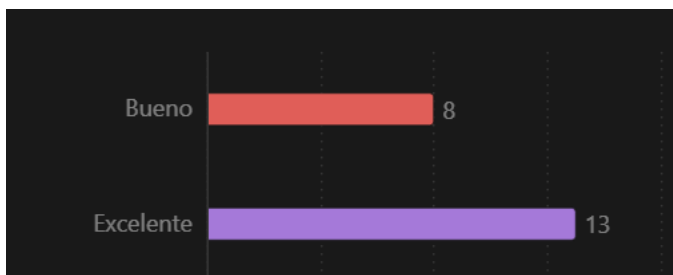
Nota. El grafico representa cómo percibieron los participantes la presentación de los palacios mentales y los resúmenes dentro de la aplicación.

El 52.4% de los encuestados (11 de 21 personas) consideró que la presentación fue “clara y útil”, mientras que un 38% (8 de 21 personas) la calificó como “muy clara y útil”. Solo un 9.5% (2 de 21 personas) se mostró neutral ante esta función. Lo cual, demuestra que la mayoría logró entender el beneficio de los palacios mentales y resúmenes que ofrece la aplicación.

3. ¿Qué te pareció el diseño general de la aplicación (colores, botones, íconos y tipografía)?

Figura 25

Resultados de la pregunta 3 Post-Test



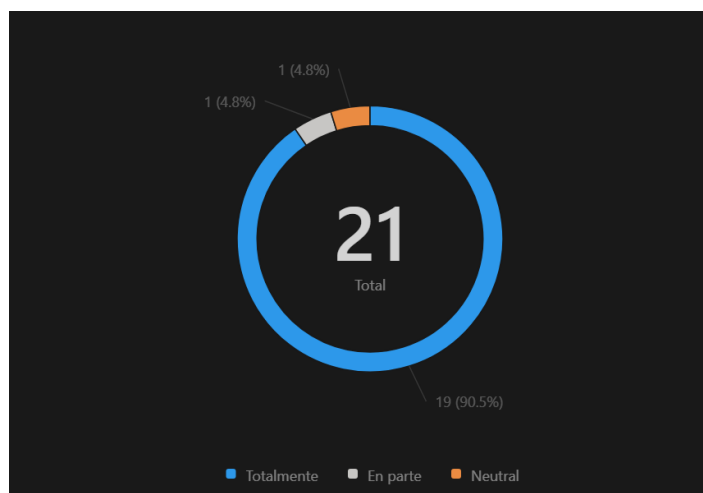
Nota. El gráfico refleja la opinión de los usuarios sobre el diseño general de la aplicación, incluyendo colores, botones, íconos y tipografía.

La mayoría, un 61.9% (13 de 21 personas), calificó el diseño como “excelente”, mientras que el 38.1% restante (8 de 21 personas) lo consideró “bueno”. Indicando que los usuarios obtuvieron una experiencia agradable y atractiva al utilizar la aplicación.

4. ¿El diseño de las pantallas facilitó encontrar los que buscabas?

Figura 26

Resultados de la pregunta 4 Post-Test



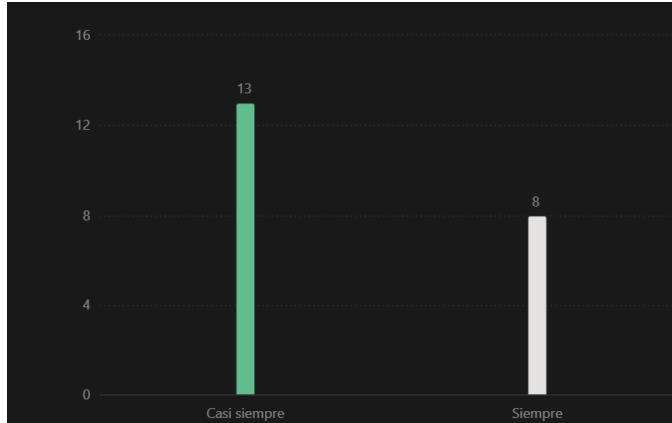
Nota. El gráfico representa la opinión de los usuarios sobre si el diseño de las pantallas facilitó encontrar lo que buscaban dentro de la aplicación.

La gran mayoría, un 90.4% (19 de 21 personas), respondió que “totalmente”, lo que indica que el diseño cumple adecuadamente con su función de orientación y usabilidad. Por otro lado, un 4.7% (1 persona) señaló que solo “en parte” y otro 4.7% se mostró neutral.

5. ¿La aplicación respondió adecuadamente a tus acciones sin errores o bloqueos?

Figura 27

Resultados de la pregunta 5 Post-Test



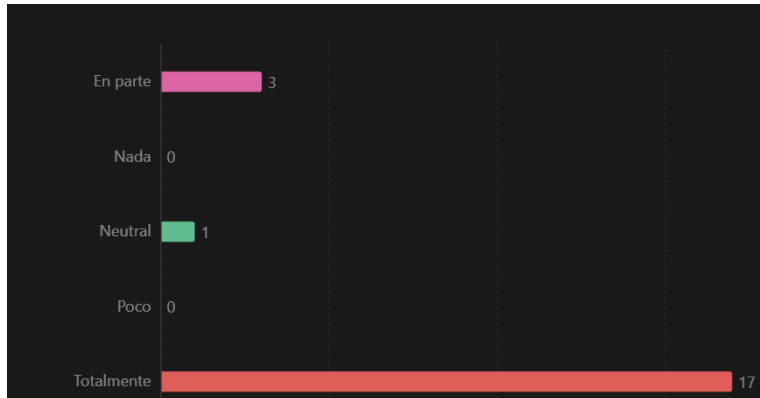
Nota. El gráfico representa la percepción de los usuarios respecto al funcionamiento técnico de la aplicación, específicamente si respondió adecuadamente a sus acciones sin errores ni bloqueos.

El 61.9% de los encuestados (13 de 21 personas) indicó que la aplicación funcionó “casi siempre” de manera adecuada, mientras que un 38.1% (8 de 21 personas) afirmó que “siempre” respondió correctamente. Esto indica que aún hay margen de mejora en la optimización del rendimiento para asegurar una experiencia completamente fluida.

6. ¿La información y funcionalidad del chat te ayudaron a comprender y repasar el contenido?

Figura 28

Resultados de la pregunta 6 Post-Test



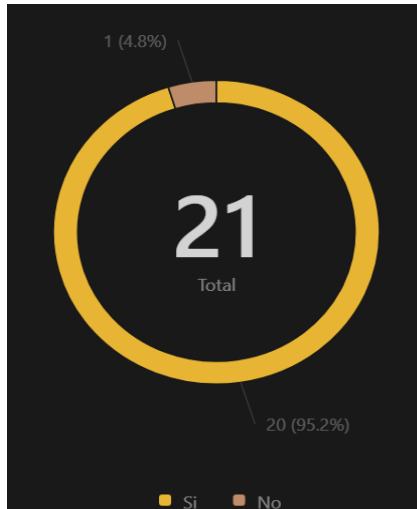
Nota. El gráfico representa la percepción de los usuarios sobre la utilidad del chat para comprender y repasar el contenido.

Un 81% de los participantes (17 de 21 personas) respondió que totalmente sintieron que el chat les ayudó en el proceso de repaso y comprensión. Por otro lado, un 14.3% (3 personas) indicó que les ayudó en parte, mientras que un 4.7% (1 persona) se mostró neutral.

7. ¿Lograste subir el documento sin inconvenientes?

Figura 29

Resultados de la pregunta 7 Post-Test



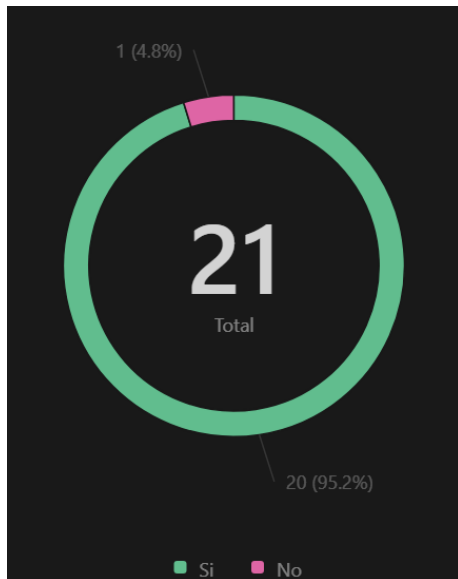
Nota. El gráfico representa la experiencia de los usuarios al subir documentos a la aplicación.

El 95.2% de los participantes (20 de 21 personas) logró subir sus documentos sin inconvenientes, lo que indica un funcionamiento confiable en este aspecto. Solo un 4.8% (1 persona) reportó dificultades para completar esta acción.

8. ¿Pudiste registrarte y crear un usuario sin problemas?

Figura 30

Resultados de la pregunta 8 Post-Test



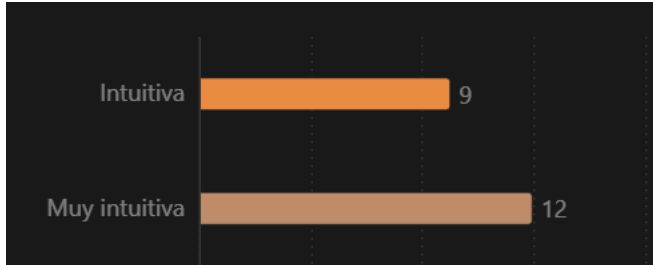
Nota. El gráfico representa los resultados sobre la facilidad para registrarse y crear un usuario en la aplicación.

Un 95.2% de los participantes (20 de 21 personas) pudo registrarse y crear su cuenta sin problemas, lo que refleja que este proceso es accesible y sencillo para la mayoría. Sin embargo, un 4.8% (1 persona) tuvo dificultades, lo que señala que podría ser necesario revisar y mejorar aspectos específicos del registro para evitar obstáculos.

9. ¿Encontraste intuitiva la interacción con el chat para consultar o repasar la información?

Figura 31

Resultados de la pregunta 9 Post-Test



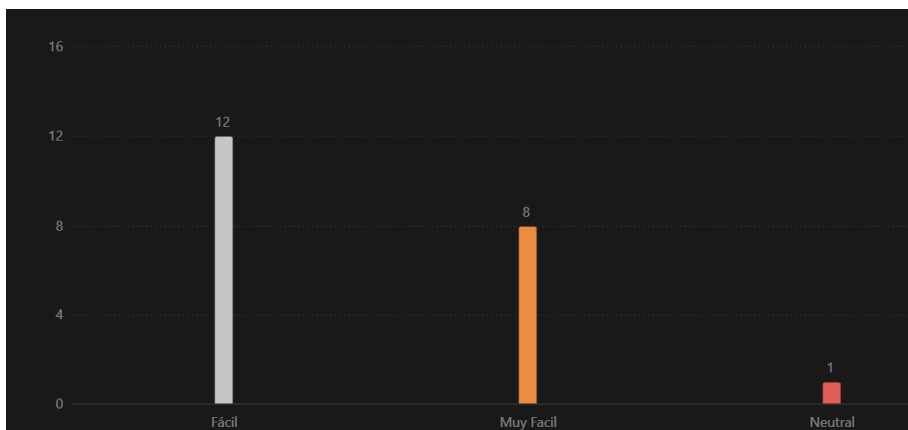
Nota. El gráfico muestra la percepción de los usuarios sobre la facilidad de uso del chat como herramienta para consultar o repasar información.

Un 57.1% de los participantes (12 de 21 personas) consideró la interacción con el chat como “muy intuitiva”, mientras que el 42.9% (9 de 21 personas) la calificó como “intuitiva”. Esto demuestra que la mayoría de los usuarios logró interactuar con el chat de forma fluida y sin complicaciones

10. ¿Cuál es tu apreciación general al usar la aplicación?

Figura 32

Resultados de la pregunta 10 Post-Test



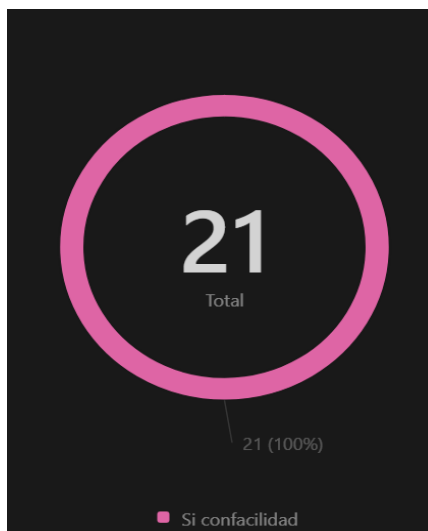
Nota. El gráfico representa apreciación general de los usuarios al utilizar la aplicación.

El 57.1% de los participantes (12 de 21 personas) calificó su experiencia como “fácil”, mientras que un 38.1% (8 de 21 personas) la consideró “muy fácil”. Solo un 4.8% (1 persona) se mostró neutral. Indicando que la mayoría de los usuarios percibió la aplicación como accesible y amigable en su uso general.

11. ¿Pudiste encontrar y gestionar fácilmente las palabras del diccionario?

Figura 33

Respuestas de la pregunta 11 Post-Test



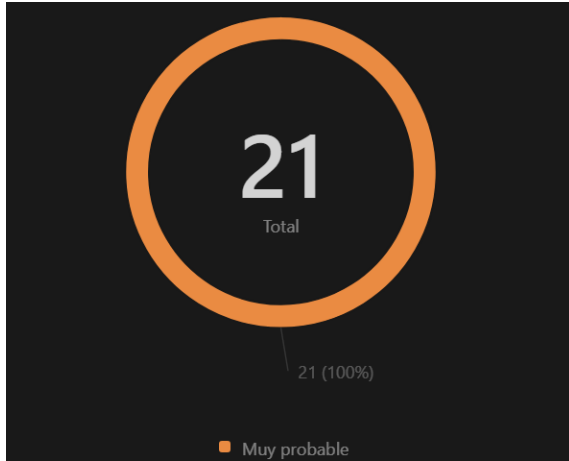
Nota. El gráfico representa la opinión unánime de los usuarios sobre la facilidad para encontrar y gestionar las palabras dentro del diccionario de la aplicación.

El 100% de los participantes (21 de 21 personas) indicó que pudieron realizar esta tarea con facilidad, lo que demuestra que esta funcionalidad está muy bien diseñada y es accesible para todos los usuarios.

12. ¿Qué tan probable es que vuelvas a usar esta aplicación en un futuro?

Figura 34

Resultados de la pregunta 12 Post-Test



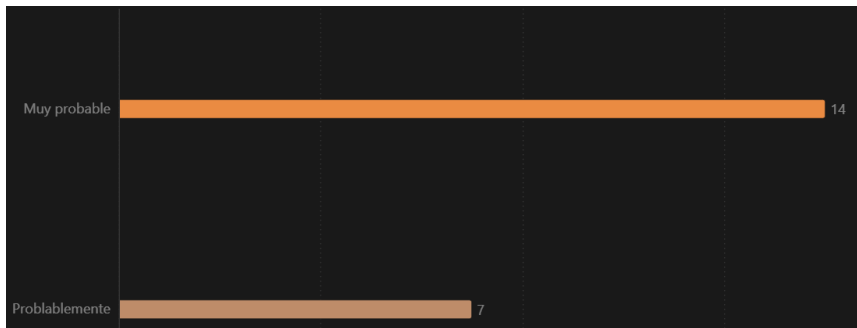
Nota. El gráfico muestra la disposición de los usuarios para volver a utilizar la aplicación en el futuro.

El 100% de los participantes (21 de 21 personas) manifestó que es muy probable que vuelvan a usar la aplicación, lo cual refleja un alto nivel de satisfacción y confianza en la misma.

13. ¿Implementarías esta nueva técnica en el futuro?

Figura 35

Resultados de la pregunta 13 Post-Test



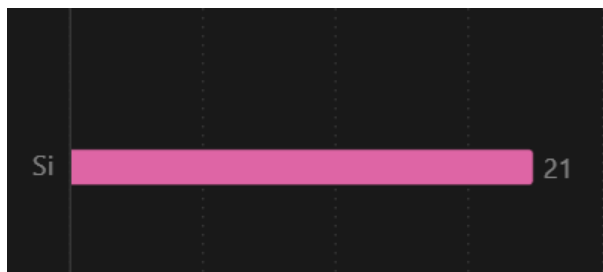
Nota. El gráfico refleja la intención de los usuarios de implementar la nueva técnica propuesta por la aplicación en su futuro estudio o trabajo.

Un 66.7% de los participantes (14 de 21 personas) expresó que “muy probablemente” implementarán esta técnica, mientras que un 33.3% (7 personas) respondió que “probablemente” lo harán. Indicando que existe una gran aceptación en cuanto a su aplicación práctica.

14. ¿Recomendarías esta aplicación a otras personas?

Figura 36

Resultados de la pregunta 14 Post-Test



Nota. El gráfico refleja la disposición total de los usuarios para recomendar la aplicación a otras personas.

El 100% de los participantes (21 de 21 personas) afirmó que sí recomendaría la aplicación, lo cual indica un nivel muy alto de satisfacción y confianza en la misma.

15. ¿Utilizaste la opción de repasar en el chat?, ¿cómo fue tu experiencia?

En general, todos los participantes afirmaron haber usado la opción de repasar en el chat, y sus experiencias fueron mayoritariamente positivas, aunque también señalaron algunos aspectos que pueden mejorarse.

Uno de los principales comentarios fue el carácter novedoso y original de la herramienta, destacando que la interacción tipo chat aporta dinamismo al estudio. Como señaló un usuario: “Fue como hablar con alguien que te ayuda a estudiar sin hacerlo aburrido”.

Otro aspecto por destacar fue la utilidad para reforzar contenidos clave, especialmente términos importantes del documento. En palabras de otro participante: “Fue útil para revisar términos clave, aunque sentí que podría profundizar más”. Esta observación revela que aún existen oportunidades para desarrollar interacciones más personalizadas según la necesidad del usuario.

Algunas respuestas también evidencian ciertas dificultades iniciales, como no saber cómo empezar o problemas técnicos menores. Por ejemplo, alguien comentó: “Sí, la usé. Me pareció útil, aunque al principio no sabía bien cómo empezar”, mientras que otra persona expresó: “Estuvo chévere esta experiencia, aunque al principio no me dejaba iniciar sesión”.

16. ¿Encontraste alguna funcionalidad difícil de usar o entender?

Las respuestas a esta pregunta muestran una experiencia mixta por parte de los participantes. Si bien algunos indicaron que no tuvieron dificultades y todo les pareció claro, otra parte sí identificó funcionalidades que les generaron confusión, sobre todo durante los primeros usos.

Una de las áreas donde más se concentraron las dudas fue en la gestión de términos. Un participante mencionó: “Al principio no entendía cómo modificar los términos”, mientras que otro señaló: “Sí, me costó identificar cómo relacionar los términos resaltados con mi documento”. Esto sugiere que, aunque la funcionalidad existe, falta una orientación más clara sobre su propósito y uso.

También hubo confusión respecto a la edición y guardado de historias. Como lo expresó un usuario: “Tuve problemas para saber cómo guardar los cambios de la historia. No sabía si se

guardaban solos o no”. Este tipo de comentario demuestra que es necesario incorporar una guía pasa a paso visible sobre cómo usar la aplicación.

Otro punto de dificultad fue el concepto de palacio mental, que generó dudas tanto en su función como en su implementación: “El término del palacio mental fue confuso, no me quedó claro cómo se hace uno. Me costó entender para qué servía en verdad”. Esto muestra que, al tratarse de una técnica poco convencional o desconocida para algunos, es necesario explicar su utilidad con ejemplos sencillos.

17. ¿Qué fue lo que más te gustó de la aplicación?

Una de las características más halagadas fue la capacidad de transformar los textos en historias, lo cual fue percibido como una funcionalidad creativa, inesperada y útil. Varios usuarios coincidieron en esto, con frases como “Me pareció genial que los textos se transformen en historias, fue algo inesperado pero útil” y “Me pareció muy cool que de mis apuntes saliera como un cuento”. Este tipo de comentarios muestra que la propuesta de presentar contenido de manera narrativa no solo fue bien recibida, sino que generó sorpresa de una manera positiva.

También hubo mención recurrente a la inteligencia artificial, tanto en su rol para generar historias como en su presencia general dentro de la aplicación. Un usuario destacó “La implementación de la inteligencia artificial”, mientras que otro valoró “La integración de IA para generar historias a partir de los documentos que subimos”.

El diseño visual también fue mencionado positivamente, tanto en términos generales como por su funcionalidad. Comentarios como “El diseño de la app” y “La temática de la app y la paleta de colores” demuestran que la apariencia fue bien recibida. Además, la forma en que se destacan los términos importantes también fue reconocida como un punto fuerte, con frases

como “Que los términos importantes estén marcados en color, así puedo ver rápido lo que debo leer” y “Que resalta lo importante. Eso me ahorra tiempo”.

18. ¿Qué fue lo que menos te gustó de la aplicación o qué mejorarías?

Uno de los comentarios más recurrentes fue la falta de una guía clara. Una persona expresó: “Faltó una guía más clara. Si no hubiera estado en el grupo de prueba, no habría sabido cómo empezar”, mientras que otra añadió: “Todo me gustó, pero aumentaría una guía paso a paso que esté a la mano”. Esta observación sugiere que la aplicación necesita acompañar al usuario en los primeros pasos para evitar confusión.

También se destacó el deseo de contar con notificaciones que recuerden cuándo repasar. Comentarios como “Me gustó todo, pero me gustaría que me notifique cuándo deba repasar” y “Lo que mejoraría es que la aplicación me dé notificaciones cuando vaya a repasar el documento” indican que los usuarios valoran funciones que faciliten el seguimiento del contenido a lo largo del tiempo.

Otra área mencionada fue la personalización del texto, especialmente en las historias generadas. Algunos usuarios expresaron: “Me hubiera gustado cambiar el tipo de letra” y “Que le puedas cambiar el tipo y tamaño de letra en la parte de las historias”.

Finalmente, se mencionaron algunas limitaciones técnicas puntuales, como lentitud al navegar entre secciones o en el proceso de registro. Frases como “Se demoró un poco en crear el usuario” y “Noté algo de lentitud cuando cambiaba a otras secciones” indican oportunidades de optimización en la velocidad de respuesta de la aplicación.

19. ¿Qué funcionalidades adicionales te gustaría ver en futuras versiones?

Una de las solicitudes más mencionadas fue la implementación de recordatorios automáticos, alineados con la técnica de repetición espaciada. Varios usuarios sugirieron ideas como “recordatorios automáticos de repaso, según lo que estudiaste antes” o “la posibilidad de crear recordatorios”, destacando la importancia de que la aplicación les ayude a mantener una rutina sin tener que hacerlo manualmente.

También surgió con una mayor integración con otras plataformas. Una persona expresó: “Integración con los calendarios del celu para planificar el repaso”.

En cuanto a la experiencia de uso, se valoró la posibilidad de agregar un tutorial con audio, ideal para quienes prefieren una guía hablada que complemente el aprendizaje visual. A esto se suma el interés por mejorar la personalización del texto, mencionando que se debería “aumentar fuentes y tamaño de letra”.

Otro aspecto interesante es que algunos sugieren la creación de “un mapa mental interactivo o una red de ideas”, lo cual va en línea con la función de palacios mentales y el aprendizaje visual que la app ya promueve.

Finalmente, se sugiere el “crear rachas” o tener “una sección donde puedas comparar tu progreso con otras personas, como un ranking o algo así”. Estas propuestas reflejan un interés por gamificar la experiencia de estudio para hacerla más dinámica.

20. ¿Tienes algún comentario adicional sobre tu experiencia con la aplicación?

Como última pregunta, uno de los comentarios más repetidos fue la apreciación de que la aplicación es “una idea muy original” o “novedosa”. Estas palabras reflejan que la propuesta

rompe con lo tradicional y despierta interés desde el primer contacto. Como dijo un participante: “Aunque fue solo una vez, sentí que es una herramienta distinta”.

También se destacó mucho el diseño visual y estético, con frases como “el diseño de la app estuvo muy bonito y colorido”, lo que confirma que la experiencia de uso no solo se evaluó desde lo funcional, sino también desde lo agradable que resulta visualmente.

Algunos participantes expresaron que, aunque fue una experiencia positiva, aún sienten que necesitan más práctica o tiempo para aprovecharla mejor. Una persona comentó: “Me gustó probarla, aunque creo que necesito más práctica para aprovecharla bien” y otra señaló: “Si tuviera más tiempo, la usaría para estudiar”.

Por otro lado, se reconoció que la aplicación tiene potencial real en contextos educativos, con observaciones como: “Me gustaría verla aplicada en entornos educativos reales” y “Creo que tiene futuro si se siguen arreglando detallitos y explican mejor algunas cosas”.

5. Conclusiones y Recomendaciones

5.1. Conclusiones

El desarrollo de esta aplicación ha hecho que implementemos conceptos ineludibles como lo son la comunicación Cliente-Servidor que para el presente proyecto se utilizó de tal manera en que podamos hacer el envío de solicitudes http y de esa manera poder comunicar nuestro *frontend* con el *backend*.

Siguiendo el mismo hilo del *backend* es importante destacar la manera en cómo se desarrollan los métodos y funciones para que toda la responsabilidad no caiga únicamente en manos de nuestra UI. Pero para ello es importante desarrollar un buen esquema de datos y saber cuándo llamar o responder a las peticiones utilizando técnicas como *throttle* o *debouncing* que son herramientas que nos apoyan de tal forma en que nosotros podamos optimizar el rendimiento y la experiencia del usuario.

Dado que el proyecto fue realizado por una tecnología de *React* el control de los componentes, así como el uso de estados y efectos se vuelven una tarea que no podemos ignorar para el correcto uso del Virtual Dom que es el concepto que implementa *React* y es lo que lo hace tan eficiente.

Sabemos además que gran parte de la lógica del proyecto ocurre en los agentes de IA que están alojados en los flujos de trabajo de *n8n* y parece prudente hacer un hincapié que la tarea de *n8n* funcionan de tal manera que su enfoque está destinado en la orquestación de tareas permitiendo de una manera más gráfica poder evidenciar lo que está ocurriendo.

Finalmente, la retroalimentación obtenida con las pruebas de usuario puede concluir que es verdad que, aunque muchas personas no tengan conocimiento de lo que es un palacio mental, se encuentra interesadas en saber cómo funciona.

5.2. Recomendaciones

Es importante considerar el entorno de ejecución de la aplicación, debido a que, si están apoyadas en herramientas como la de *Expo Go*, es fundamental constatar que los dispositivos y la máquina anfitrión estén bajo la misma red de Internet. Además, también es una tarea primordial verificar si alguno de los puertos de esa red se mantiene bloqueados por cuestiones de seguridad.

Para el trabajo con *React Native* la mejor práctica que se puede hacer es dividir el trabajo de una manera modular, de ese modo garantizamos que cada una de nuestras tareas estén bien separadas de tal modo que en caso de que queramos utilizarla para otro proyecto podemos hacer reusable este módulo.

Para el trabajo con *React* es importante dividir nuestra UI en componentes que sean más pequeños, con el fin de precautelar su mantenimiento y desarrollo.

Para *frameworks* como *NestJS* es recomendable integrar librerías como la que se implementó en el proyecto que fue Prisma con el propósito de tener un control y un tipado para nuestras bases de datos.

Finalmente, en cuanto a las pruebas con los usuarios fomentar el pensar en voz alta, resulta ser muy provechoso para poder obtener mejores resultados ya que de esa manera tu percibes la forma en como percibe la prueba el usuario y cuáles son sus pensamientos en el momento en que está realizando las actividades.

6. Referencias

Coursera Staff. (2024, August 9). *What Is React Native? Beginner's Guide + FAQ* | Coursera. <https://www.coursera.org/articles/what-is-react-native>

Day, C. (2020, July 7). *Los Orígenes del Método de los Loci - El Campeonato Mundial de la Memoria*. <https://www.worldmemorychampionships.com/the-origins-of-the-method-of-loci/>

Defeyter, M. A., Russo, R., & McPartlin, P. L. (2009). The picture superiority effect in recognition memory: A developmental study using the response signal procedure. *Cognitive Development*, 24(3), 265–273. <https://doi.org/10.1016/j.cogdev.2009.05.002>

Domínguez, I. (2023, August 23). *Por qué necesitamos contar historias* | Ethic. Ethic. <https://ethic.es/por-que-necesitamos-contar-historias>

Figueroba, A. (2017, July 11). *Repetición espaciada (técnica para memorizar): ¿cómo se usa?* <https://psicologiymente.com/inteligencia/repeticion-espaciada>

González, R. (2025, March 10). *El futuro de la IA conectada: Qué es un servidor MCP y por qué podría reemplazar los sistemas RAG - Blog de hiberus*. Hiberus. <https://www.hiberus.com/crecemos-contigo/el-futuro-de-la-ia-conectada-que-es-un-servidor-mcp/>

Jain, R. (2025, March 11). *Lynx JS vs. React Native vs. Flutter: A Comprehensive Comparison for Developers* | by Ritik jain | Medium. Medium. <https://ritiwrites.medium.com/lynx-js-vs-react-native-vs-flutter-a-comprehensive-comparison-for-developers-58db5d00a770>

Lynx JS vs Flutter vs React Native. (2025, March 15). <https://lynx-js.com/lynx-js-vs-flutter-vs-react-native-which-one-should-you-choose-in-2025/>

n8n: An Overview of the Workflow Automation Tool. (2024, June 10). DataScientest. <https://datascientest.com/en/n8n-an-overview-of-the-workflow-automation-tool>

Nicole, C. (2025, June 3). *Qué es Frontend y Backend: principal diferencia y ejemplos.* <https://platzi.com/blog/que-es-frontend-y-backend/>

¿Qué es una API de REST? (2023, June 31). Red Hat. <https://www.redhat.com/es/topics/api/what-is-a-rest-api>

¿Qué es una base de datos relacional? | IBM. (2021, October 20). IBM. <https://www.ibm.com/think/topics/relational-databases>

Raeburn, A. (2025, February 13). *¿Qué es la programación extrema (XP)? [2025] • Asana.* Asana. <https://asana.com/es/resources/extreme-programming-xp>


Schwaber-Cohen, R. (2023, May 3). *What is a Vector Database & How Does it Work? Use Cases + Examples | Pinecone.* Pinecone. <https://www.pinecone.io/learn/vector-database/>

Tyagi, A. (2024, July 10). *Desarrollo Full Stack con Postgres Next.js, Clerk y Neon.* FreeCodeCamp. <https://www.freecodecamp.org/news/nextjs-clerk-neon-fullstack-development/>

Von Essen, J. (2022, October 10). *Sherlock's Mind Palace: El arte de la técnica del palacio de la mente.* MemoryOS. <https://memoryos.com/article/sherlocks-mind-palace>

7. Anexos

A. Encuesta Pre-Test



Yunta (Pre-test)

Queremos conocer si utilizas herramientas digitales en la creación de tus apuntes. Esta información nos ayudará a diseñar mejores soluciones que se adapten a tus necesidades. Agradecemos mucho tu tiempo 🙌 y colaboración para responder estas preguntas 🙏.

This form is public. Anyone with the link can submit a response. [Change](#)

¿Cuál es tu nombre completo? 🧐*

Respondent's answer

¿Cuál es tu edad? 😎*

Respondent's answer

¿Cuál es tu genero?*

(Respondents can select up to 1)

Hombre

Mujer

Otro

+ Add option

¿Cuál es tu email? ✉️*

Respondent's answer

¿Qué aplicaciones o herramientas utilizas para crear y organizar tus apuntes?

(Respondents can select up to 3)

Notion

Obsidian

Canva

Notas del celular

Apuntes a mano

Anky

OneNote

+ Add option

¿Has oído hablar del concepto de “palacio mental”?

(Respondents can select up to 1)

- Sí lo conozco y lo utilizo.
- Sí pero no lo utilizo.
- No nunca he oído hablar de eso.

+ Add option

¿Conoces en qué consiste la técnica de repetición espaciada?*

(Respondents can select up to 1)

- Sí la aplico regularmente.
- Sí pero no la aplico.
- No no la conozco.

+ Add option

¿Tienes acceso a internet en todo momento para consultar tus apuntes?*

(Respondents can select up to 1)

- Sí siempre.
- A veces depende del lugar.
- No casi nunca.

+ Add option

¿Utilizas inteligencia artificial para crear o mejorar tus apuntes?*

(Respondents can select up to 1)

- Sí frecuentemente.
- A veces cuando lo necesito
- No nunca.

+ Add option

¿Cómo identificas las ideas principales al crear tus apuntes?*

(Respondents can select up to 1)

- Uso herramientas digitales (resaltadores apps IA).
- Subrayo o tomo notas manualmente.
- No tengo un método específico.

+ Add option

¿Qué tan útil te parecen las herramientas digitales para crear apuntes?*

(Respondents can select up to 1)

- Muy útiles.
- Algo útiles pero no son indispensables.
- Poco o nada útiles para mí.

+ Add option

¿Prefieres estudiar con recursos digitales o con materiales impresos?*

(Respondents can select up to 1)

- Prefiero recursos digitales (apps PDFs videos).
- Prefiero materiales impresos (libros apuntes en papel).
- Uso ambos por igual.

+ Add option

B. Encuesta Post-Test

Cual es tu apreciación general al usar la aplicación*

(Respondents can select up to 1)

- Muy Facil
- Fácil
- Neutral
- Dificil
- Muy Dificil

+ Add option

Como calificarias la facilidad para navegar por las diferentes seccionse de la aplicacion (Presentacion, Home, Diccionario y Chat)*

(Respondents can select up to 1)

- Muy Facil
- Facil
- Neutral
- Dificil
- Muy Dificil

+ Add option

Encontraste intuitiva la interaccion con el chat para consultar o repasar la informacion*

(Respondents can select up to 1)

- Muy intuitiva
- Intuitiva
- Neutral
- Poco Intuitiva
- Nada Intuitiva

+ Add option

Pudiste encontrar y gestionar facilmente las palabras en el diccionario? *

(Respondents can select up to 1)

- Si confacilidad
- Si con dificultad
- No

+ Add option

La aplicacion respondo adecuadamente a tus acciones sin errores o bloqueos*

(Respondents can select up to 1)

- Siempre
- Casi siempre
- Algunas veces
- Nunca

+ Add option

Que te parecio el disenio general de la aplicacion (colores, botones, iconos y tipografia)*

(Respondents can select up to 1)

- Excelente
- Bueno
- Regular
- Malo
- Muy malo

+ Add option

Consideras que la presentación de los palacios mentales y resúmenes es clara*

(Respondents can select up to 1)

- Muy clara y útil
- Clara y útil
- Neutral
- Poco clara
- Nada clara
- + Add option

La información y funcionalidad del chat, te ayudaron a comprender y repasar el contenido*

(Respondents can select up to 1)

- Totalmente
- En parte
- Neutral
- Poco
- Nada
- + Add option

El diseño de las pantallas facilitó encontrar lo que buscabas*

(Respondents can select up to 1)

- Totalmente
- En parte
- Neutral
- Poco
- Nada
- + Add option

Pudiste registrarte y crear un usuario sin problemas*

(Respondents can select up to 1)

- Si
- No
- + Add option

Lograste subir el documento sin inconvenientes*

(Respondents can select up to 1)

- Si
- No
- + Add option

Utilizaste la opción de repasar en el chat, como fue tu experiencia*

Respondent's answer

Encontraste alguna funcionalidad difícil de usar o entender*

Respondent's answer

Que fue lo que mas te gusto de la aplicacion*

Respondent's answer

Que fue lo que menos te gusto de la aplicacion o que mejorarias*

Respondent's answer

Que funcionalidades adicionales te gustaria ver en futuras versiones*

Respondent's answer

Recomendarías esta aplicación a otras personas *

(Respondents can select as many as they like)

Si

No

+ Add option

Que tan probable es que vuelvas a usar esta aplicacion en el futuro.*

(Respondents can select as many as they like)

Muy probable

Poco probable

Muy poco probable

+ Add option

Tienes algun comentario adicional sobre tu experiencia con la aplicacion*

Respondent's answer

Implementarías esta nueva técnica en el futuro?

(Respondents can select up to 1)

- Muy probable
- Probablemente
- Nada probable
- + Add option