

1. ANEXOS

1.1 Historias de usuario

- **Como** jugador, **quiero** recibir recompensas (puntos) al completar misiones **para** sentirme motivado a seguir avanzando en el juego.
- **Como** jugador, **quiero** poder invitar a mis amigos a jugar en partidas cooperativas en línea **para** disfrutar del juego en equipo y mejorar la experiencia multijugador.
- **Como** jugador, **quiero** acceder a un tutorial interactivo al comenzar el juego **para** aprender fácilmente las mecánicas básicas y poder disfrutar del juego sin dificultades.
- **Como** jugador, **quiero** ver tablas de clasificación globales y de mis amigos **para** comparar mi rendimiento y competir por las mejores puntuaciones.

1.2 Requerimientos no funcionales

Identificador:		Nombre: tutorial	
Tipo: guía	Requerimiento que lo utiliza o especializa: interfaz		¿Crítico? No
Prioridad de desarrollo: media	Documento de visualización asociado: mockups		
Entrada: N/A	Salida: N/A		
Descripción: proporcionar algunas indicaciones para que el usuario pueda usar nuestro programa sin ningún inconveniente			
Manejo de situaciones anormales: N/A			
Criterios de aceptación: <ul style="list-style-type: none">• presentar un tutorial didáctico que cualquier persona pueda comprender			

Identificador:		Nombre: personalización	
Tipo: personalizar	Requerimiento que lo utiliza o especializa: interfaz		¿Crítico? No
Prioridad de desarrollo: media	Documento de visualización asociado: mockups		
Entrada: N/A	Salida: N/A		
Descripción: permitirá al usuario personalizar un avatar a su gusto			
Manejo de situaciones anormales: N/A			
Criterios de aceptación: <ul style="list-style-type: none">• crear un avatar completamente a tu gusto			

1.3 Requerimiento funcional

Identificador:	Nombre: iniciar sesión
Tipo: funcional	Requerimiento que lo utiliza o especializa
Prioridad de desarrollo: alta	Documento de visualización asociado: mockups
Entrada: validar si el usuario y la contraseña de la persona son correctas	Salida: permitir ingresar al sistema
Descripción: se encargará de validar si el usuario y la contraseña del usuario son correctas, si son correctas se le permitirá iniciar al sistema sino se denegará el acceso	
Manejo de situaciones anormales: <ul style="list-style-type: none"> • se le mostrará un mensaje donde se le informará que tendrá 3 intentos para escribirlos correctamente y en el caso de no hacerlo se le mostrará otro mensaje informando que deberá registrarse 	
Criterios de aceptación: <ul style="list-style-type: none"> • validar y permitir el acceso de nuestros usuarios al sistema 	

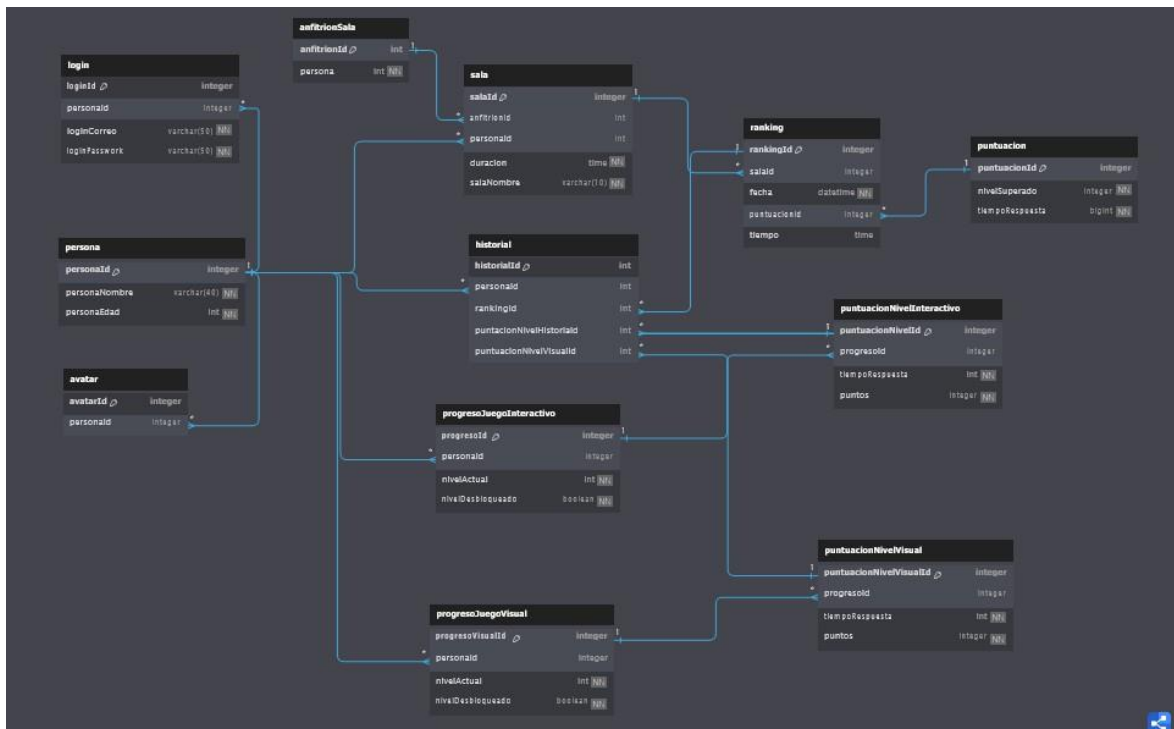
Identificador:	Nombre: registro
Tipo: funcional	Requerimiento que lo utiliza o especializa
Prioridad de desarrollo: alta	Documento de visualización asociado: mockups
Entrada: un usuario y una contraseña creada por el usuario	Salida: cuenta registrada, iniciar sesión
Descripción: permitirá a los usuarios crear una cuenta por medio de un usuario y una contraseña con la cual después de registrados deberán iniciar sesión	
Manejo de situaciones anormales: <ul style="list-style-type: none"> mostrar mensaje si el usuario o la contraseña ya está en uso mensaje de contraseña errónea sino cumple con los requerimientos necesarios 	
Criterios de aceptación: <ul style="list-style-type: none"> mantener un control del registro de todos nuestros usuarios 	

Identificador:	Nombre: categorías
Tipo: funcional	Requerimiento que lo utiliza o especializa
Prioridad de desarrollo: alta	Documento de visualización asociado: mockups
Entrada: seleccionar o crear alguna de las categorías	Salida: seleccionar nivel
Descripción: en esta opción podrás seleccionar entre las categorías que te proporcionamos o en caso de no llamarte la atención alguna también tienes la opción de tu mismo crear la tuya	
Manejo de situaciones anormales: <p>N/A</p>	
Criterios de aceptación: <ul style="list-style-type: none"> proporcionar al usuario una gran variedad de categorías 	

Identificador:	Nombre: Nivel
Tipo: funcional	Requerimiento que lo utiliza o especializa
Prioridad de desarrollo: alta	Documento de visualización asociado: mockups
Entrada: seleccionar el nivel de tu preferencia	Salida: Iniciar a jugar
Descripción: te permite elegir el nivel que desees para tu categoría, esta opción estará disponible en los dos modos de juego (solitario y con amigos)	
Manejo de situaciones anormales: <ul style="list-style-type: none"> en caso de no elegir algún nivel el sistema te seleccionara el nivel 1 automáticamente 	
Criterios de aceptación: <ul style="list-style-type: none"> generar a nuestros usuarios el espíritu de la competitividad 	

Identificador:	Nombre: solo o con amigos
Tipo: funcional	Requerimiento que lo utiliza o especializa
Prioridad de desarrollo: alta	Documento de visualización asociado: mockups
Entrada: seleccionar entre jugar solo o con amigos	Salida: seleccionar alguna categoría
Descripción: Después de registrarte, elige jugar solo o con amigos. En solitario, selecciona la categoría y nivel, y presiona "Iniciar" para comenzar. Con amigos, una persona será el administrador y a esa misma se le generará un código QR con el cual podrán unirse los demás compañeros. Una vez dentro, el administrador elige la categoría y nivel y ya solamente presiona "Iniciar" para comenzar	
Manejo de situaciones anormales: <p>N/A</p>	
Criterios de aceptación: <ul style="list-style-type: none"> generas un ambiente agradable entre tus compañeros 	

1.4 MER



1.5 Requerimiento de hardware

- **Sistema Operativo:** Windows 10/11 (64 bits), macOS 10.14 o superior, Ubuntu 20.04 (para desarrolladores que usen Linux)
- **Procesador (CPU):** Intel Core i5 o AMD Ryzen 5 (para manejar tareas básicas de desarrollo)
- **Memoria RAM:** 16 GB de RAM (para ejecutar motores de juego y herramientas de diseño)
- **Tarjeta Gráfica (GPU):** NVIDIA GTX 1060 6GB o AMD Radeon RX 580 8GB (para desarrollo y pruebas gráficas)
- **Almacenamiento:** 512 GB SSD (para tiempos de carga rápidos y manejo de archivos grandes)
- **Monitor:** Full HD (1920x1080), preferiblemente de 24" o superior para mayor área de trabajo
- **Periféricos:** Teclado, ratón, y controlador (gamepad) para pruebas de juego
- **Conexión a Internet:** Para trabajo colaborativo y manejo de archivos en la nube.

Especificaciones:https://docs.google.com/spreadsheets/d/1I7NYX9KUGElqSNc26ReDg_k9-ICdmkla/edit?usp=drive_link&oid=115741417886687263434&rtpof=true&sd=true

Requerimientos del Software

El objetivo de este proyecto es desarrollar un juego de memoria que permita a los usuarios mejorar sus habilidades de retención de información de forma didáctica y entretenida. Con una dinámica interactiva y una interfaz atractiva, se busca posicionarlo como una herramienta destacada en el mercado, promoviendo el desarrollo cognitivo mientras los jugadores disfrutan de una actividad estimulante.

Para lograr este objetivo, el aplicativo contará con una interfaz llamativa e interactiva que garantice una experiencia agradable para el usuario. La creación del software involucró el uso de diversos lenguajes de programación y herramientas, los cuales jugaron un papel clave en el desarrollo tanto del frontend como del backend. A continuación, se detallarán las tecnologías utilizadas y su contribución al proyecto.

FRONTEND	
html,js y css	Estos lenguajes de marcado y programación fueron fundamentales para lograr una interfaz atractiva e interactiva en nuestro aplicativo. Con HTML y CSS, pudimos crear un diseño visualmente llamativo y estructurado, brindando el toque estético necesario para captar la atención del usuario. Por otro lado, JavaScript nos permitió añadir funcionalidades dinámicas, mejorando significativamente la interacción del usuario con la interfaz y ofreciendo una experiencia más fluida e intuitiva.
bootstrap	Para nuestro aplicativo, era fundamental que se adaptara a cualquier dispositivo, y aquí es donde entra Bootstrap. Este framework nos facilitó enormemente esta tarea, ya que ofrece una amplia variedad de herramientas y opciones diseñadas para garantizar la compatibilidad y responsividad en diferentes tamaños de pantalla, cumpliendo exactamente con lo que necesitábamos.
BACKEND	
PostgreSQL	Para el aplicativo, necesitábamos un sistema de gestión de datos robusto que permitiera manejar y almacenar grandes volúmenes de información. Para resolver este desafío, decidimos utilizar PostgreSQL, ya que ofrece las herramientas necesarias para garantizar un manejo eficiente de los datos, cumpliendo con los requerimientos de nuestro proyecto.

Ratchet	Ratchet fue el servidor gratuito que nos permitió alojar y ejecutar nuestro aplicativo, brindándonos una solución práctica y accesible para su despliegue.
node.js	Node.js fue el encargado de permitirnos desplegar nuestro aplicativo en el servidor de Ratchet, proporcionando un entorno eficiente y escalable para gestionar las operaciones del lado del servidor. Además, fue clave para agregar la funcionalidad de multijugador a nuestro aplicativo, mejorando la experiencia interactiva de los usuarios.

