



NEURO-GAME

Tu memoria es mi misión

Presentado por:

SANTIAGO CHAPARRO

YERSON STIVEN RUBIANO



1. Resumen Ejecutivo

NeuroGame es un juego de memoria diseñado para mejorar las habilidades cognitivas de los jugadores a través de secuencias y patrones cada vez más complejos. El proyecto tiene como objetivo proporcionar una plataforma interactiva que combine diversión y ejercicios mentales, ofreciendo beneficios cognitivos. Se espera que el juego ayude a mejorar la memoria y la atención, además de proporcionar una experiencia entretenida para los jugadores. Se identifican desafíos relacionados con la falta de experiencia en parte del equipo, y se recomiendan pruebas continuas para garantizar la calidad.

2. Introducción

Esta propuesta técnica y económica detalla el desarrollo de NeuroGame, una solución interactiva que combina entretenimiento y entrenamiento cognitivo. El juego está diseñado para usuarios de todas las edades que deseen mejorar sus habilidades de memoria a través de un enfoque lúdico. En este documento, se presenta la estructura del proyecto, los recursos necesarios y un cronograma detallado para su desarrollo.



3. Descripción del Proyecto

Objetivos:

- Proporcionar una herramienta accesible y divertida para mejorar la memoria y la atención de los usuarios.
- Crear un sistema de recompensas para motivar a los jugadores a avanzar en los niveles.
- Facilitar el acceso a un modo multijugador con chat integrado para fomentar la interacción.

Requisitos:

- Registro e inicio de sesión con correo electrónico y contraseña.
- Guardar información del perfil de usuario, como partidas jugadas, monedas ganadas y compras en la tienda.
- Funcionalidad de chat en modo multijugador.
- Implementación de seguridad usando Fetch para manejar APIs de forma segura.

Características:

NeuroGame permite a los jugadores iniciar sesión y acceder a su perfil con historial de partidas y compras. En cada nivel, los jugadores ganan experiencia y monedas que pueden utilizar para personalizar su perfil con nuevas skins. El juego incluye un modo multijugador con chat para mejorar la interacción.

Mercado y Competencia:

El juego se posiciona en el creciente mercado de juegos de entrenamiento cognitivo, con competidores como Simon y Memoria. La ventaja competitiva de NeuroGame radica en su progresiva dificultad y su sistema de recompensas que incentiva la mejora continua.



4. Solución Propuesta

NeuroGame será desarrollado utilizando una arquitectura basada en capas. Las tecnologías clave incluyen:

- Frontend: HTML, Bootstrap y CSS para la interfaz de usuario.
- Backend: PHP y Fetch para comunicación segura con el servidor.
- Base de datos: SQL para el manejo de la información de los usuarios, partidas y compras.

El juego estará alojado en un servidor propio, y la seguridad será gestionada a través de Fetch para proteger las APIs. El desarrollo incluye funcionalidades clave como el registro, inicio de sesión, almacenamiento de información de partidas y la integración de un chat multijugador.

5. Plan de Trabajo

- El desarrollo de NeuroGame se llevará a cabo en las siguientes fases:
 1. Fase de diseño y mockups (junio - agosto 2024): Creación del diseño visual y wireframes en Figma.
 2. Desarrollo de frontend y backend (agosto - octubre 2024): Implementación de la interfaz, funcionalidad de usuario y bases de datos.
 3. Pruebas y ajustes (noviembre 2024): Realización de pruebas de usuario, depuración y ajustes antes del lanzamiento.
 - Utilizaremos Trello para gestionar el proyecto y realizar un seguimiento de las tareas y fases.
-



6. *Equipo de Trabajo*

- Santiago Chaparro Riaño: Programador principal, responsable del desarrollo backend y la integración de bases de datos.
 - Yerson Stiven Cuellar Rubiano: Diseñador de interfaz y experiencia de usuario, encargado de la implementación visual.
-

7. *Presupuesto*

El presupuesto se desglosa de la siguiente manera:

- **Costos de personal:**
 - Programador: 200 horas a \$15 USD/hora = \$3,000 USD.
 - Diseñador: 150 horas a \$12 USD/hora = \$1,800 USD.
 - Tester: 100 horas a \$10 USD/hora = \$1,000 USD.
 - **Licencias de software:**
 - Adobe Photoshop: \$90 USD.
 - **Infraestructura:**
 - Hosting en servidor propio: \$200 USD.
 - **Total estimado: \$6,090 USD.**
-



8. Plan de Mantenimiento y Soporte

Después del lanzamiento, se ofrecerán dos actualizaciones trimestrales para agregar nuevas funcionalidades y realizar ajustes en la jugabilidad. El soporte técnico se brindará a través de Gmail, con un tiempo de respuesta estimado de 48 horas. También se monitoreará continuamente el rendimiento del juego para corregir errores y mejorar la seguridad.

9. Cronograma de Pagos

- Inicio del proyecto (junio 2024): 20% del pago total.
 - Desarrollo del prototipo (agosto 2024): 30%.
 - Pruebas y ajustes (noviembre 2024): 30%.
 - Entrega final y lanzamiento (noviembre 2024): 20%.
-



Anexos

1. Historias de Usuario

Las historias de usuario describen interacciones simples entre el jugador (usuario) y el sistema (juego). Cada historia se escribe desde la perspectiva del usuario y detalla lo que espera lograr con el sistema.

Ejemplo de Historias de Usuario para NeuroGame:

- Registro: "Como nuevo usuario, quiero registrarme con mi correo electrónico y contraseña para acceder a todas las funcionalidades del juego."
- Iniciar Sesión: "Como jugador, quiero iniciar sesión para continuar desde donde dejé mi última partida."
- Ver perfil: "Como jugador, quiero ver mi perfil, historial de partidas y monedas ganadas para seguir mi progreso."
- Comprar en la tienda: "Como jugador, quiero usar las monedas ganadas para comprar skins u otros elementos en la tienda."
- Chat en multijugador: "Como jugador, quiero interactuar con otros jugadores en el modo multijugador a través de un chat."
- Estas historias guían el desarrollo y ayudan a priorizar funciones esenciales para los usuarios.



2. Requisitos Funcionales y No Funcionales

Los requisitos funcionales describen qué debe hacer el software para cumplir su objetivo, mientras que los requisitos no funcionales describen las características de calidad, como rendimiento y seguridad.

Requisitos Funcionales:

- Registro e inicio de sesión: El sistema debe permitir que los usuarios se registren e inicien sesión con su correo electrónico y contraseña.
- Perfil de usuario: El sistema debe guardar el historial de partidas, monedas ganadas y compras realizadas por el usuario.
- Sistema de puntuación y recompensas: El juego debe registrar la puntuación del jugador al finalizar cada partida y asignar recompensas en monedas.
- Multijugador y chat: Los usuarios deben poder interactuar con otros jugadores en tiempo real a través de un chat integrado en el modo multijugador.

Requisitos No Funcionales:

- Seguridad: El sistema debe implementar el manejo de APIs usando Fetch para asegurar las comunicaciones y proteger los datos del usuario.
- Rendimiento: El tiempo de carga del juego no debe exceder los 3 segundos en conexiones estándar de internet.
- Compatibilidad: El juego debe funcionar en portátiles, laptops y teléfonos inteligentes con acceso a internet.

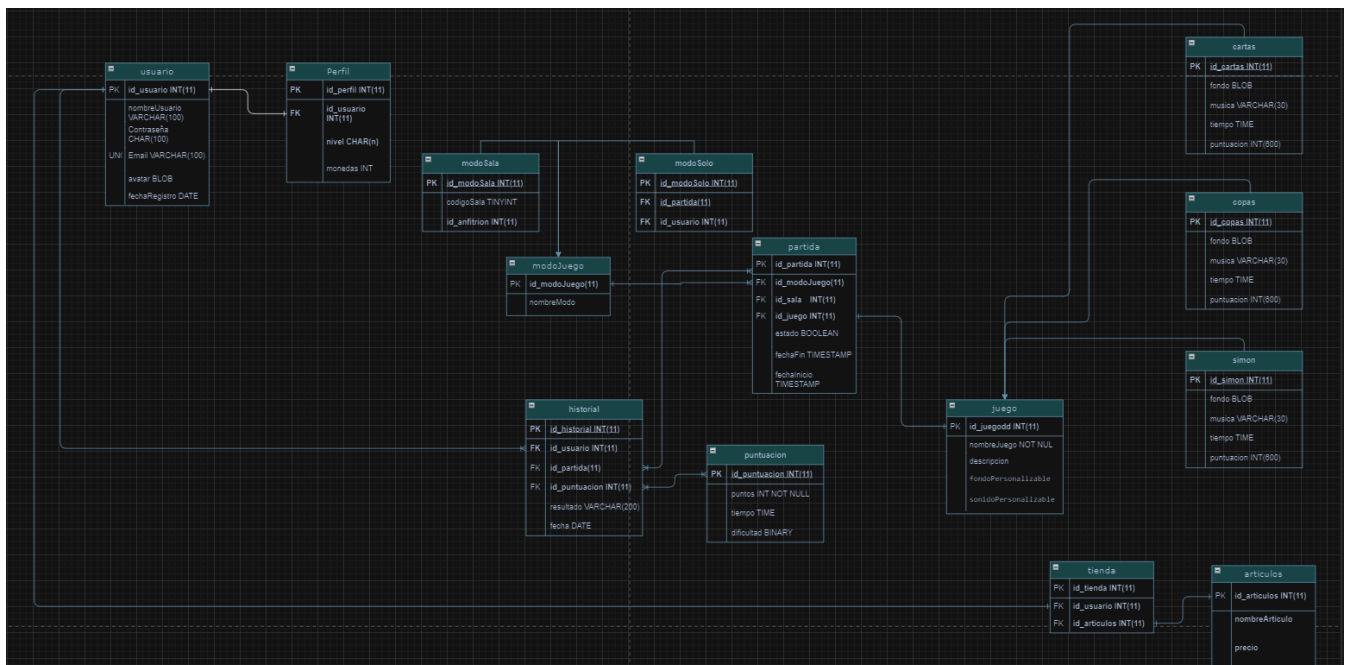


3. Modelo Entidad-Relación (MER)

El MER es un diagrama que muestra las relaciones entre las entidades del sistema, en este caso, el juego NeuroGame. Ya mencionaste que tienes un MER listo, pero aquí te doy una idea de las principales entidades que deben estar incluidas:

- Usuario: Información del jugador (nombre, correo, contraseña).
- Partida: Detalles de cada partida jugada (fecha, puntuación, duración).
- Moneda: Monedas acumuladas por el jugador.
- Tienda: Productos disponibles para la compra (skins u otros objetos).
- Historial de compras: Relación entre el usuario y los productos adquiridos.

Este diagrama ayuda a visualizar cómo se relacionan las distintas tablas en tu base de datos.



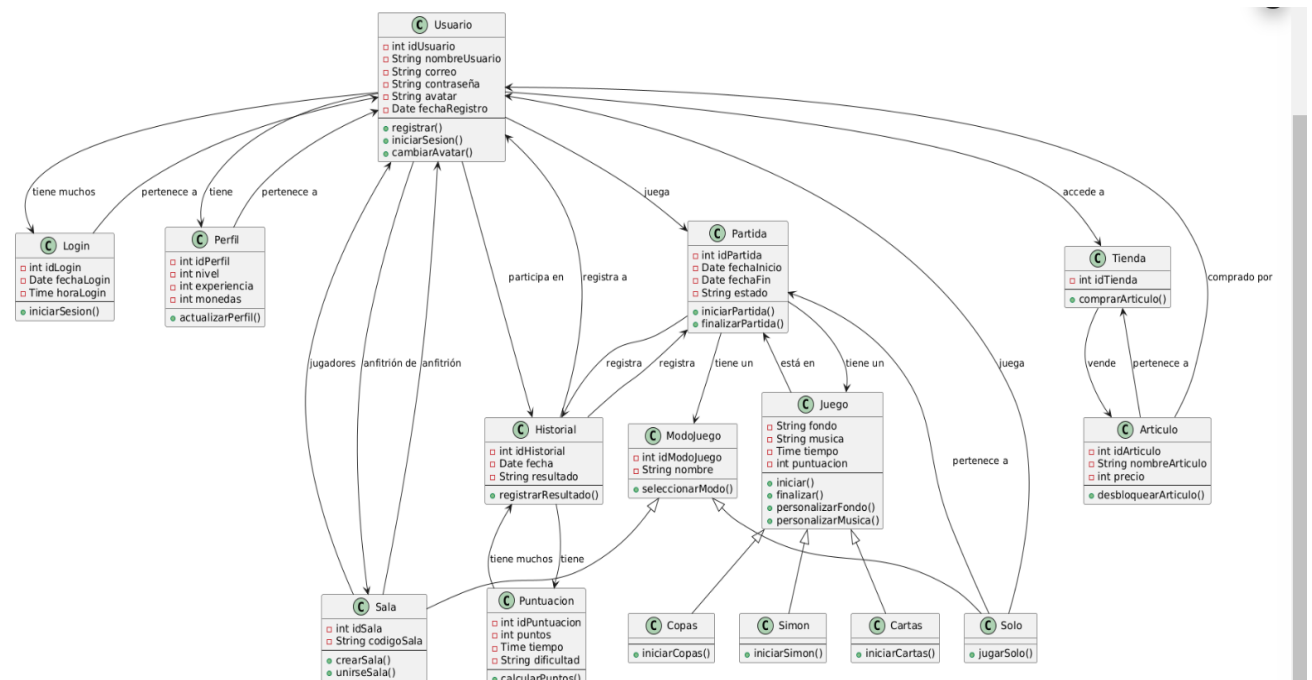


4. Diagrama de Casos de Uso

El diagrama de casos de uso muestra las interacciones principales entre el usuario y el sistema desde una perspectiva de alto nivel. No tienes este diagrama aún, pero podrías tener los siguientes casos de uso:

1. Registrarse/Iniciar sesión: El usuario se registra o inicia sesión en el juego.
2. Jugar una partida: El usuario selecciona un nivel y juega una partida de memoria.
3. Ver el perfil: El usuario accede a su perfil para ver su progreso, historial de partidas y compras.
4. Comprar en la tienda: El usuario utiliza monedas ganadas para adquirir objetos en la tienda.
5. Chat en modo multijugador: El usuario interactúa con otros jugadores en el modo multijugador.

Este diagrama visualiza los procesos más importantes y sus interacciones, ayudando a diseñar el flujo del juego.



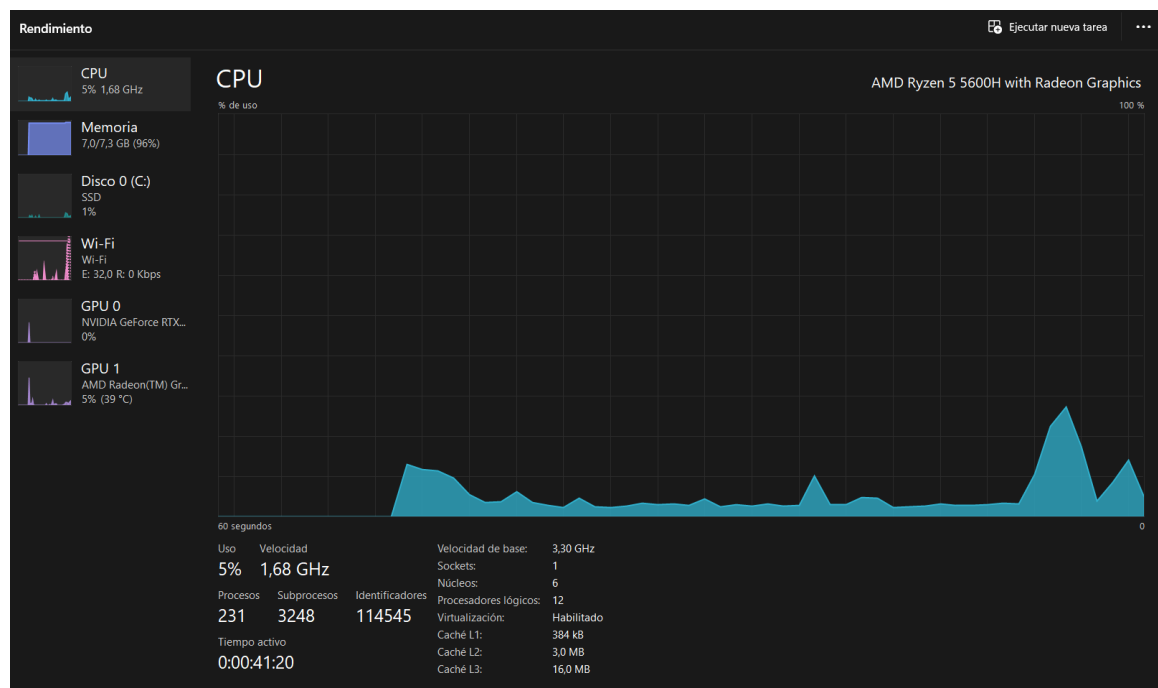


5. Requisitos de Hardware y Software

Estos requisitos describen las especificaciones mínimas y recomendadas para que el juego funcione correctamente. Pueden incluir:

Requisitos de Hardware:

- Portátil o laptop: Requisitos mínimos de memoria RAM (2GB mínimo, 4GB recomendado) y procesador (Intel Core i3 o superior).
- Teléfonos inteligentes: iOS 12 o Android 8.0 en adelante.
- Conexión a Internet: Velocidad mínima de 1 Mbps.





Requisitos de Software:

- Frontend: HTML, Bootstrap, y CSS.
- Backend: PHP, SQL para la base de datos.
- Seguridad: Implementación de Fetch para manejo seguro de APIs.

Estos requisitos aseguran que el juego pueda ejecutarse sin problemas en los dispositivos de los usuarios y que se mantenga seguro.





6. Riesgos y Mitigación

Aquí identificarás los posibles riesgos que pueden afectar el éxito del proyecto y cómo planeas enfrentarlos.

Ejemplo de riesgos para NeuroGame:

- Retrasos en el desarrollo: Mitigación mediante la capacitación de los miembros del equipo y el uso de Trello para gestionar tareas y plazos.
- Problemas técnicos: Implementación de pruebas regulares y revisión de código para identificar y corregir errores antes de las fases críticas.

Riesgos y Mitigación

El principal riesgo identificado es la falta de experiencia en parte del equipo, lo que podría causar retrasos en el desarrollo. Para mitigar este riesgo, se recomienda realizar capacitaciones periódicas y dividir el trabajo de manera más eficiente, utilizando herramientas de gestión de proyectos como Trello para priorizar las tareas críticas.

