UNIVERSIDAD TECNOLOGICA DE AGUASCALIENTES

INGENIERÍA EN DESARROLLO Y GESTIÓN DE SOFTWARE

DESARROLLO DE UNA APLICACIÓN WEB BASADA EN UN CASO DE ESTUDIO

MATERIA:

DESARROLLO WEB INTEGRAL

PROFESOR:

ROBERTO CORREA BRAVO

NOMBRE DEL ALUMNO:

ANA PATRICIA JUÁREZ RAMÍREZ

GRUPO:

IDGS 9A-11

# AGUASCALIENTES, AGS. 06 DE JULIO DE 2025

**SITIO WEB DE JUEGOS INTERACTIVOS**

**Implementación del Caso de Estudio**

**Descripción del proyecto:**

El proyecto nace de la necesidad de ofrecer un espacio web gratuito, accesible y sin registros, donde cualquier persona pueda jugar títulos clásicos y de lógica. En un entorno digital donde la mayoría de los sitios de juegos requieren crear cuentas o tienen publicidad invasiva, se busca desarrollar una plataforma sencilla, sin distracciones, centrada en la experiencia del juego y el entretenimiento sano.

**Objetivo:**

Desarrollar un sitio web funcional que permita a los usuarios acceder de manera inmediata a una variedad de juegos clásicos y de lógica sin necesidad de crear cuentas, con una interfaz limpia, amigable y optimizada para distintos dispositivos.

**Alcance:**

**Incluye:**

* + Acceso inmediato a juegos sin registro.
  + Juegos implementados directamente en el navegador.
  + Interfaz intuitiva y responsiva.
  + Accesibilidad desde PC, tablet o móvil.
  + Sección de instrucciones para cada juego.

**No incluye:**

* + Registro o inicio de sesión.
  + Almacenamiento de progreso.
  + Juego multijugador en línea.
  + Integración con redes sociales.

**Requerimientos funcionales:**

* 1. El sistema debe permitir al usuario seleccionar un juego desde la página principal.
  2. El sistema debe iniciar el juego directamente en el navegador sin redirecciones externas.
  3. El sistema debe mostrar una opción de instrucciones.
  4. El sistema debe permitir reiniciar la partida en cualquier momento.
  5. El sistema debe tener un diseño responsivo que se adapte a distintos tamaños de pantalla.

1. **Requerimientos no funcionales:**
   1. Usabilidad: Interfaz clara, intuitiva y sin elementos que confundan al usuario.
   2. Rendimiento: Los juegos deben cargarse en menos de 3 segundos en conexiones promedio.
   3. Seguridad: No se manejan datos sensibles.
   4. Compatibilidad: Soporte para los navegadores modernos.

**Planificación Ágil:**

El siguiente plan presenta la organización del desarrollo de una plataforma web de juegos. Se utiliza metodología Scrum, considerando un único producto que integra todos los juegos.

Backlog de Producto

La tabla siguiente muestra las tareas del producto, ordenadas por prioridad. Para cada elemento se indica una breve descripción de la tarea, la duración estimada en semanas y los objetivos que se quieren alcanzar:

|  |  |  |  |
| --- | --- | --- | --- |
| Sprint | Tareas | Duración Estimada | Objetivos |
| Sprint 1 (Semanas 1–2) | - Configuración inicial del proyecto y repositorio.  - Estructura inicial de la interfaz.  - Diseño y desarrollo de la UI del menú principal.  - Desarrollo completo de Ahorcado y Gato (lógica, frontend y pruebas). | 2 semanas | Crear la base del proyecto, establecer las rutas y los componentes básicos.  Completar los juegos Ahorcado y Gato.  Tester verifica que ambos juegos funcionen correctamente. |
| Sprint 2 (Semanas 3–4) | - Implementación de Memorama y Trivia.  - Diseño de la UI de Memorama y Trivia.  - Implementación de la lógica de emparejamiento de cartas en Memorama.  - Implementación del módulo de preguntas y puntuación en Trivia.  - Pruebas de QA de ambos juegos. | 2 semanas | Completar e integrar Memorama y Trivia. Asegurar que ambos juegos sean funcionales y que el usuario pueda navegar por todos los juegos en el menú. |
| Sprint 3 (Semanas 5–6) | - Desarrollo de Juego de Matemáticas.  - Desarrollo de Buscaminas.  - Pruebas de QA exhaustivas en ambos juegos. | 2 semanas | Desarrollar e integrar Juego de Matemáticas y Buscaminas. Asegurar que ambos juegos sean jugables de forma adecuada, y pasen las pruebas de funcionalidad y jugabilidad. |
| Sprint 4 (Semanas 7–8) | - Diseño y desarrollo de 2048.  - Implementación del Simulador de Decisiones.  - Pruebas de QA de ambos juegos. | 2 semanas | Desarrollar e integrar 2048 y Simulador de Decisiones. Asegurar que ambos juegos sean funcionales, con el flujo de decisiones y las combinaciones en 2048 bien implementados. |
| Sprint 5 (Semanas 9–10) | - Diseño de la UI de Sudoku.  - Implementación de la lógica para generación de puzzles en niveles fáciles, medios y difíciles.  - Pruebas de QA de Sudoku. | 2 semanas | Completar Sudoku con las funcionalidades de generación de puzzles en tres niveles de dificultad, junto con pruebas de funcionalidad. |
| Sprint 6 (Semanas 11–12) | - Diseño de la interfaz de Solitario.  - Implementación completa del Solitario.  - Pruebas de integración y pruebas de QA para toda la plataforma. | 2 semanas | Desarrollar e integrar Solitario. Asegurar que todos los juegos funcionen correctamente juntos, con la funcionalidad de inicio, reinicio, y deshacer movimientos implementada. |

**Sprints Planificados**

Sprint 1: Se centra en la configuración inicial del proyecto y la implementación de dos juegos sencillos, Ahorcado y Gato. Estos serán los primeros juegos funcionales disponibles en la plataforma.

Sprint 2: Los juegos Memorama y Trivia se añaden con interfaces y lógicas más complejas, permitiendo al jugador disfrutar de otros tipos de entretenimiento mientras se asegura que los juegos anteriores sean funcionales.

Sprint 3: El Juego de Matemáticas y Buscaminas agregan desafíos lógicos y estratégicos, mejorando la diversidad de la plataforma.

Sprint 4: 2048 y el Simulador de Decisiones enriquecen la experiencia con juegos de lógica numérica y narrativa.

Sprint 5: El Sudoku, un juego clásico de lógica, es una de las funcionalidades más complejas, por lo que se dedica un sprint completo para su desarrollo.

Sprint 6: Finalmente, Solitario se agrega al final, lo que requiere un diseño y desarrollo más extenso debido a la mecánica de cartas, movimientos y reglas. También se realiza una integración de todos los juegos para garantizar la estabilidad de la plataforma.

**Diseño de Arquitectura y Patrones:**

**Justificación de la arquitectura seleccionada:**

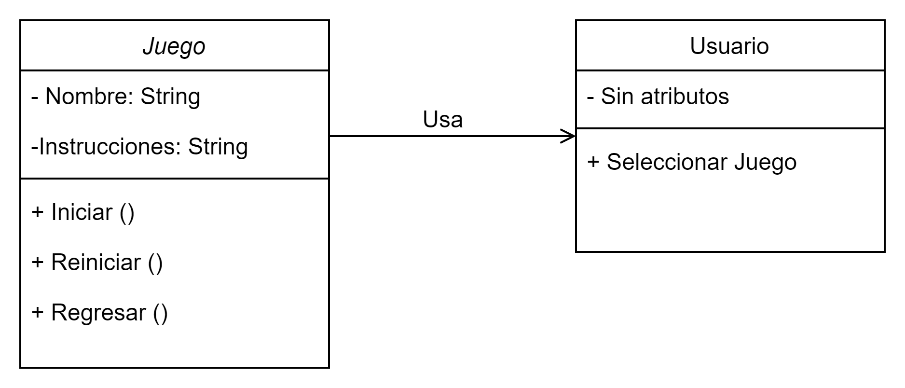
Se opta por una arquitectura cliente-servidor monolítica, donde todo el contenido se sirve desde un único backend básico o incluso como sitio estático. Esta elección se justifica por la simplicidad del sistema, la baja necesidad de recursos y la facilidad de implementación y mantenimiento. No se requiere una arquitectura compleja como microservicios, ya que el sitio no manejará lógica pesada ni bases de datos.

**Modelado:**

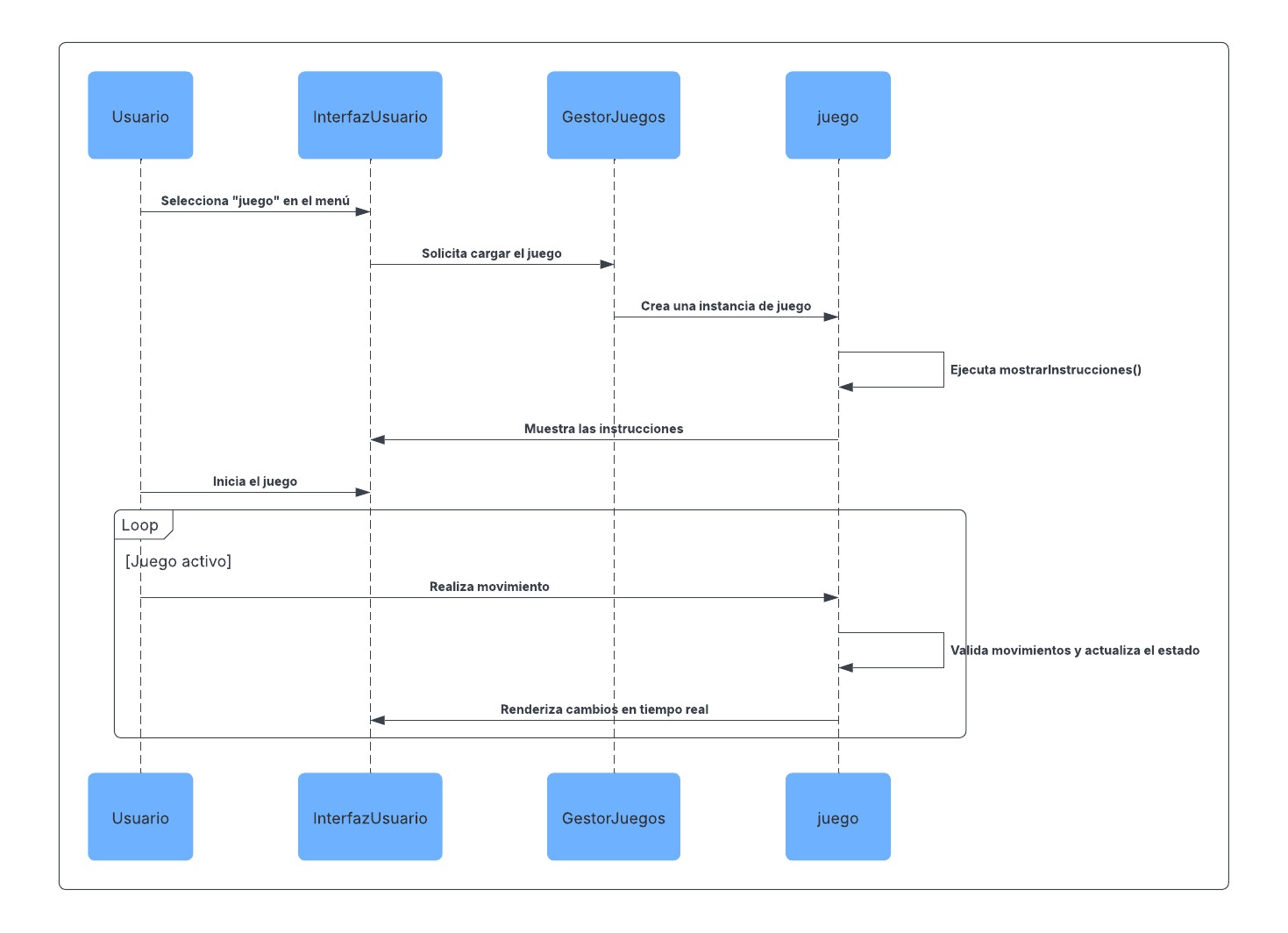
Diagrama de casos de uso:



Diagrama de clases:



diagramas de secuencia:



**Desarrollo con Frameworks:**

**Frontend**

* **Bootstrap 5**

Framework CSS para el diseño responsive y componentes UI. Usado para el navbar, grid system, cards, botones y utilidades de espaciado

* **Font Awesome 6**

Biblioteca de íconos vectoriales. Usado para los íconos de navegación, botones y elementos decorativos

**Backend**

* **Firebase**

Usado para el hosting del frontend, además de esta no se utilizaron más frameworks para backend ya que no requiere conexión a bases de datos y los juegos se ejecutan en el cliente

**Seguridad**

* No se incorporan mecanismos de seguridad avanzados porque no maneja datos sensibles ni requiere autenticación.

**Integración de Servicios Web:**

**Servicio web propio:**

En un modelo de arquitectura cliente-servidor monolítica, la lógica de la aplicación está contenida en una única aplicación. Esto significa que tanto el frontend (interfaz de usuario) como el backend (lógica) se encuentran juntos en una sola unidad.

En este tipo de arquitectura, no es necesario usar una API propia si todos los componentes de la aplicación están ejecutándose dentro de la misma estructura.

**Servicio web de terceros:**

Para el Solitario Spider, hemos integrado una API externa especializada en juegos de cartas que se encarga de toda la lógica del juego.

**Pruebas y Validación:**

|  |  |
| --- | --- |
| Variable | Juego Sudoku |
| Nombre de la prueba | Verificación de la jugabilidad del Sudoku. |
| Descripción | Probar la jugabilidad del juego, incluyendo la inserción de números, validación y finalización. |
| Objetivo | Asegurar que el usuario pueda jugar correctamente, insertar valores, validar el Sudoku y completarlo sin errores. |
| Pasos a seguir | 1. Iniciar el juego. 2. Insertar un número en las celdas del tablero. 3. Verificar si el número es válido. 4. Rellenar el tablero con números válidos. 5. Validar que el Sudoku se complete correctamente. |
| Resultado esperado | El tablero se debe completar correctamente, sin errores, y al final debe mostrar un mensaje de victoria. |
| Resultado obtenido |  |
| Aprobado | Si |

|  |  |
| --- | --- |
| Variable | Juego Trivia |
| Nombre de la prueba | Verificación de la jugabilidad de Trivia. |
| Descripción | Probar la correcta visualización de las preguntas, las opciones de respuesta y la actualización de la puntuación. |
| Objetivo | Validar que las preguntas se muestren correctamente, las respuestas sean seleccionadas y se actualice la puntuación. |
| Pasos a seguir | 1. Iniciar el juego. 2. Responder a una pregunta. 3. Verificar si la puntuación se actualiza correctamente. 4. Continuar con las siguientes preguntas. 5. Verificar el puntaje final después de responder todas las preguntas. |
| Resultado esperado | El juego debe avanzar correctamente entre preguntas, sumando puntos y mostrando el puntaje final al finalizar. |
| Resultado obtenido |  |
| Aprobado | Si |

|  |  |
| --- | --- |
| Variable | Juego de Matemáticas |
| Nombre de la prueba | Verificación de la jugabilidad del Juego de Matemáticas. |
| Descripción | Probar que las operaciones matemáticas sean aleatorias, validas y que el sistema calcule correctamente las respuestas. |
| Objetivo | Asegurarse de que las operaciones se generen correctamente y que la puntuación se actualice al responder correctamente. |
| Pasos a seguir | 1. Iniciar el juego. 2. Responder a una operación matemática. 3. Verificar si la respuesta es correcta. 4. Verificar si la puntuación se actualiza al responder correctamente. |
| Resultado esperado | La operación debe mostrarse correctamente, la respuesta debe ser validada y la puntuación debe sumarse adecuadamente. |
| Resultado obtenido |  |
| Aprobado | Si |

|  |  |
| --- | --- |
| Variable | Juego Memorama |
| Nombre de la prueba | Verificación de la jugabilidad del Memorama. |
| Descripción | Probar la correcta funcionalidad del emparejamiento de cartas y la visualización de las tarjetas volteadas. |
| Objetivo | Validar que el emparejamiento de cartas funcione correctamente y que el usuario pueda completar la tarea sin errores. |
| Pasos a seguir | 1. Iniciar el juego. 2. Voltear dos cartas. 3. Verificar si las cartas coinciden. 4. Continuar emparejando las cartas. 5. Verificar que el juego termine cuando todas las cartas estén emparejadas. |
| Resultado esperado | Las cartas deben emparejarse correctamente, y al finalizar el juego, se debe mostrar un mensaje de victoria. |
| Resultado obtenido |  |
| Aprobado | Si |

|  |  |
| --- | --- |
| Variable | Juego Ahorcado |
| Nombre de la prueba | Verificación de la jugabilidad del Ahorcado. |
| Descripción | Probar la correcta funcionalidad del juego de Ahorcado, incluyendo la selección de letras y la visualización de ahorcado. |
| Objetivo | Asegurar que el usuario pueda adivinar las letras correctamente y que el juego termine al fallar o completar la palabra. |
| Pasos a seguir | 1. Iniciar el juego. 2. Adivinar letras para completar la palabra. 3. Verificar si la palabra se completa. 4. Fallar algunas letras y observar si el ahorcado se completa. |
| Resultado esperado | El juego debe permitir adivinar letras, mostrar el progreso y terminar correctamente cuando el jugador gane o pierda. |
| Resultado obtenido |  |
| Aprobado | Si |

|  |  |
| --- | --- |
| Variable | Juego Gato |
| Nombre de la prueba | Verificación de la jugabilidad del juego Gato. |
| Descripción | Comprobar la funcionalidad de la lógica de las jugadas y las condiciones de victoria. |
| Objetivo | Asegurarse de que los jugadores puedan jugar y que las condiciones de victoria (línea de 3) sean detectadas. |
| Pasos a seguir | 1. Iniciar el juego. 2. Realizar movimientos de X y O. 3. Verificar si algún jugador gana con tres en línea. |
| Resultado esperado | El juego debe detectar correctamente cuando un jugador gana y terminar el juego. |
| Resultado obtenido |  |
| Aprobado | Si |

|  |  |
| --- | --- |
| Variable | Juego Decisiones |
| Nombre de la prueba | Verificación de la jugabilidad del juego de Decisiones. |
| Descripción | Asegurar que las decisiones se procesen correctamente y que el flujo del juego cambie según las elecciones del jugador. |
| Objetivo | Validar que las elecciones afecten el flujo del juego y que el resultado final sea el esperado según las decisiones. |
| Pasos a seguir | 1. Iniciar el juego. 2. Elegir una opción en cada decisión. 3. Verificar que las elecciones cambien el curso de la historia. |
| Resultado esperado | El juego debe mostrar diferentes resultados según las decisiones tomadas y terminar de manera diferente dependiendo de las elecciones. |
| Resultado obtenido |  |
| Aprobado | Si |

|  |  |
| --- | --- |
| Variable | Juego Solitario |
| Nombre de la prueba | Verificación de la jugabilidad del Solitario. |
| Descripción | Comprobar que las cartas se muevan correctamente entre las pilas y que las reglas del juego sean validadas correctamente. |
| Objetivo | Asegurarse de que las cartas puedan moverse y apilarse correctamente según las reglas del solitario. |
| Pasos a seguir | 1. Iniciar el juego. 2. Mover cartas entre las pilas. 3. Verificar que el juego detecte cuando se haya completado. |
| Resultado esperado | El juego debe permitir mover cartas entre pilas y detectar cuando se completan las secuencias. |
| Resultado obtenido |  |
| Aprobado | Si |

|  |  |
| --- | --- |
| Variable | Juego 2048 |
| Nombre de la prueba | Verificación de la jugabilidad del 2048. |
| Descripción | Validar que el juego funcione correctamente, permitiendo deslizar las fichas y combinar números hasta alcanzar 2048. |
| Objetivo | Asegurarse de que las fichas se deslicen correctamente y se combinen al alcanzar valores iguales. |
| Pasos a seguir | 1. Iniciar el juego. 2. Mover las fichas usando las teclas de dirección. 3. Verificar que las fichas se combinen correctamente. |
| Resultado esperado | El juego debe permitir mover las fichas y combinar números hasta llegar a 2048. |
| Resultado obtenido |  |
| Aprobado | Si |

|  |  |
| --- | --- |
| Variable | Juego Buscaminas |
| Nombre de la prueba | Verificación de la jugabilidad de Buscaminas. |
| Descripción | Comprobar que el tablero del buscaminas funcione correctamente, con minas, banderas y revelado de casillas. |
| Objetivo | Validar que el usuario pueda marcar banderas, descubrir casillas y que el juego termine cuando se encuentra una mina o el tablero se limpia. |
| Pasos a seguir | 1. Iniciar el juego. 2. Marcar algunas minas con banderas. 3. Revelar casillas y verificar el estado del juego. |
| Resultado esperado | El juego debe permitir marcar banderas, descubrir casillas y terminar correctamente. |
| Resultado obtenido |  |
| Aprobado | Si |

|  |  |
| --- | --- |
| Variable | Página de Inicio (Index) |
| Nombre de la prueba | Verificación del funcionamiento del menú principal. |
| Descripción | Probar que el menú principal muestre correctamente los enlaces a los diferentes juegos y páginas de información. |
| Objetivo | Validar que el menú permita navegar a los diferentes juegos y páginas de contacto e información. |
| Pasos a seguir | 1. Iniciar la página de inicio. 2. Verificar que los enlaces a los juegos funcionen correctamente. 3. Verificar que los enlaces de "Información" y "Contacto" redirijan a las páginas correspondientes. |
| Resultado esperado | El menú debe funcionar correctamente, llevando al usuario a las páginas de juegos, información y contacto. |
| Resultado obtenido |  |
| Aprobado | Si |

|  |  |
| --- | --- |
| Variable | Páginas de Información y Contacto |
| Nombre de la prueba | Verificación de la página de Información y Contacto. |
| Descripción | Probar que las páginas de "Información" y "Contacto" se carguen correctamente y muestren el contenido esperado. |
| Objetivo | Validar que las páginas de "Información" y "Contacto" se carguen correctamente y muestren la información correcta. |
| Pasos a seguir | 1. Navegar a la página de información. 2. Verificar que el contenido se muestre correctamente. 3. Navegar a la página de contacto y verificar que los datos se muestren correctamente. |
| Resultado esperado | Las páginas deben cargarse correctamente, mostrando el contenido de "Información" y los datos de "Contacto". |
| Resultado obtenido |  |
| Aprobado | Si |

**Repositorio y Entrega:**

**Manual de usuario**

La página de Juegos Interactivos está disponible para todos los usuarios y se puede acceder desde cualquier dispositivo con conexión a internet. No se requiere registro para disfrutar de los juegos, ya que la plataforma es de acceso libre para todos los visitantes.

***Navegación***

*Página Principal*

La página principal muestra un resumen de todos los juegos disponibles en la plataforma, junto con los botones de acceso rápido a cada uno de ellos. Desde aquí puedes explorar los distintos juegos interactivos. Además, encontrarás los enlaces a las páginas de contacto e información.

*Sudoku*

En esta sección podrás disfrutar del clásico juego de Sudoku. El objetivo es llenar la cuadrícula 9x9 con números del 1 al 9, sin repetir números en filas, columnas y subcuadrículas. Además, podrás elegir entre diferentes niveles de dificultad (fácil, medio, difícil).

*Trivia*

Participa en un juego de trivia con preguntas sobre diversos temas. Responde correctamente para acumular puntos. Esta sección incluye preguntas de opción múltiple, y al finalizar, se muestra el puntaje obtenido.

*Juego de Matemáticas*

Pon a prueba tus habilidades matemáticas con un juego de operaciones matemáticas. Responde correctamente a las preguntas y trata de obtener la mejor puntuación posible.

*Memorama*

En este juego tendrás que emparejar cartas iguales en el menor tiempo posible. El tablero se muestra con todas las cartas boca abajo, y deberás recordar las posiciones para hacer coincidir las cartas, además tienes un temporizador.

*Ahorcado*

El clásico juego del ahorcado. Intenta adivinar la palabra secreta de diferentes categorías antes de que se complete la figura del ahorcado. Puedes adivinar letras y ver si forman parte de la palabra.

*Gato*

Juega a este clásico juego de tres en raya con otro jugador. El objetivo es formar una línea de tres de tus símbolos en el tablero, ya sea en horizontal, vertical o diagonal.

*Decisiones*

Este es un juego narrativo en el que debes tomar decisiones que influirán en el curso de la historia. Elige tus respuestas con cuidado y explora diferentes finales.

*Solitario Spider*

El objetivo es organizar las cartas en secuencias completas, del Rey al As, del mismo palo. Utiliza las cartas disponibles en el tablero y las del mazo para completar las secuencias

*2048*

El objetivo de este juego es combinar fichas con el mismo número hasta llegar al número 2048. Desliza las fichas para combinar y sumar puntos.

*Buscaminas*

El clásico juego de Buscaminas donde debes revelar casillas del tablero, evitando las minas. Usa los números para deducir la ubicación de las minas.

***Información Adicional***

La plataforma está diseñada para ser intuitiva y fácil de usar. Si en algún momento no encuentras lo que buscas o necesitas ayuda adicional, puedes utilizar los botones de "Instrucciones" disponibles en cada juego para obtener más detalles sobre cómo jugar.

Si necesitas más asistencia, puedes visitar la sección de Contacto donde encontrarás métodos de contacto para resolver cualquier duda o inconveniente.

**Repositorio de GitHub:**

<https://github.com/AnaPatriciajuarez/juegos-interactivos>

**Enlace a la aplicación en funcionamiento:**

<https://juegosinteractivos-cdc8f.web.app/>