

## Sistema de Compilación de Juegos Arcade (PixelArcade)

por  
**ROBOTICSO . CO**

Dueño del documento:

ROBOTICSO . CO

ROBOTICSO . CO

Integrantes:

- Becerra Sipiran, Cledy Elizabeth
- Camero Gonzalez, Ian Gabriel
- Oviedo Sivincha, Massiel
- Ramos Villena, Sergio Leandro
- Villanueva Borda, Harold Alejandro

Versión

VERSIÓN	FECHA	DESCRIPCIÓN	CONTROLADOR
1.0	11-07-2022	Documentación del Proyecto	Yessenia Deysi Yari Ramos

# ÍNDICE

<b>1.INTRODUCCIÓN</b>	<b>3</b>
<b>2. PROPÓSITO DEL DOCUMENTO</b>	<b>4</b>
<b>3. TICTACTOE</b>	<b>5</b>
3.1.Descripción	5
3.2.Instrucciones de Juego	5
3.3.Casos de Uso	6
3.3.Diagrama de Secuencia	7
<b>4.SNAKE</b>	<b>8</b>
4.1.Descripción	8
4.2.Instrucciones de Juego	9
4.3.Casos de Uso	9
4.3.Diagrama de Secuencia	10
<b>5. AHORCADO</b>	<b>11</b>
5.1.Descripción:	11
5.2.Instrucciones de Juego	11
5.3.Casos de Uso	12
5.3.Diagrama de Secuencia	13
<b>6. CONFIGURACIÓN DE ENTORNO</b>	<b>14</b>

## 1.INTRODUCCIÓN

Elm es un lenguaje de programación funcional y puramente funcional que se utiliza principalmente para el desarrollo de aplicaciones web front-end. Aunque no es el lenguaje más comúnmente utilizado para el desarrollo de juegos, es posible implementar un sistema de arcade de juegos en Elm.

Por ello al implementar Pixel Arcade tuvimos en cuenta la administración del ciclo de vida del juego, la representación y manipulación de gráficos, la gestión de la entrada del usuario y la lógica del juego. Es importante destacar que implementar un sistema de arcade de juegos completo en Elm puede requerir conocimientos más avanzados del lenguaje y la plataforma. Sin embargo, Pixel Arcade ofrecerá 3 juegos basicos conocidos como Tic-Tac-Toe, Snake y Ahorcado. Estos juegos clásicos te proporcionarán horas de entretenimiento y desafío.

### 1. Tic-Tac-Toe

Descripción: Tic-Tac-Toe, también conocido como "Gato" o "Tres en Raya", es un juego de estrategia para dos jugadores.

Objetivo: El objetivo del juego es conseguir colocar tres fichas en línea, ya sea horizontal, vertical o diagonalmente, antes que el oponente.

Reglas básicas: Los jugadores se turnan para colocar sus fichas en un tablero de 3x3.

El primer jugador que complete una línea gana el juego, y si no hay más movimientos posibles y no hay ganador, el juego termina en empate

### 2. Snake

Descripción: Snake es un juego en el que controlas una serpiente que crece a medida que consume alimentos.

Objetivo: El objetivo del juego es evitar chocar contra las paredes o contra tu propia cola mientras recoges los alimentos dispersos por el tablero.

Reglas básicas: La serpiente se mueve continuamente en una dirección. El jugador debe cambiar de dirección para guiar a la serpiente hacia los alimentos. Cada vez que la serpiente consume un alimento, crece y se suma puntos. Si la serpiente colisiona con una pared o con su propia cola, el juego termina.

### 3. Ahorcado

Descripción: Ahorcado es un juego de adivinanzas en el que debes descubrir una palabra oculta antes de que se complete el dibujo de un ahorcado.

Objetivo: El objetivo del juego es adivinar correctamente la palabra oculta antes de que se dibuje por completo el ahorcado.

Reglas básicas: Se muestra una serie de guiones que representan las letras de la palabra oculta. El jugador debe adivinar las letras ingresándolas una por una. Si la letra adivinada está en la palabra, se revelará su posición en los guiones. Si la letra no está en la palabra, se dibujará una parte del ahorcado. Si se completa el dibujo del ahorcado antes de adivinar la palabra, el juego termina.

Cada juego dentro de Pixel Arcade proporciona una experiencia única y desafíos emocionantes. Consulta la documentación individual de cada juego para obtener instrucciones detalladas sobre cómo jugar y cómo interactuar con el sistema.

## 2. PROPÓSITO DEL DOCUMENTO

El propósito de este documento es proporcionar a los usuarios una guía completa y clara sobre cómo utilizar y comprender el funcionamiento de los juegos incluidos en el sistema Pixel Arcade. Esta documentación tiene los siguientes propósitos principales:

- **Instrucciones de juego:** La documentación proporciona instrucciones detalladas sobre cómo jugar cada uno de los juegos incluidos en el sistema Pixel Arcade. Explica las reglas, el objetivo, las mecánicas de juego y cualquier aspecto relevante que los jugadores necesiten saber para disfrutar plenamente de los juegos.
- **Guía de uso:** La documentación sirve como una guía de uso del sistema Pixel Arcade, explicando cómo navegar por la interfaz, seleccionar juegos, ajustar la configuración y realizar cualquier acción necesaria para interactuar con el sistema. Proporciona una descripción de los controles y la forma de interactuar con los juegos utilizando Elm.
- **Configuración del entorno:** Si es necesario configurar el entorno de desarrollo o implementación para ejecutar el sistema Pixel Arcade, la documentación incluirá instrucciones claras sobre cómo configurar y preparar el entorno adecuadamente. Esto puede incluir la instalación de Elm y cualquier otra dependencia necesaria.
- **Explicación de la arquitectura:** Elm sigue una arquitectura de modelo-vista-actualización (Model-View-Update) para desarrollar aplicaciones. La documentación puede proporcionar una descripción de esta arquitectura y cómo se aplica en el sistema Pixel Arcade mediante diagramas de casos de uso y diagramas de secuencias. Esto ayudará a los desarrolladores a comprender la estructura del código y cómo extender o modificar los juegos existentes.

En resumen, la documentación del sistema arcade de juegos Pixel Arcade en Elm tiene como objetivo principal brindar a los usuarios una referencia completa y detallada que les permita comprender, jugar y disfrutar plenamente de los juegos incluidos en el sistema. También sirve como una guía para los desarrolladores interesados en comprender la implementación en Elm como lenguaje funcional.

### 3. TICTACTOE

El Tic-Tac-Toe es un juego de estrategia para dos jugadores que se ha vuelto popular en todo el mundo. También conocido como "Gato" o "Tres en Raya", el objetivo del juego es colocar tres fichas en línea, ya sea horizontal, vertical o diagonalmente, antes que tu oponente.

En Pixel Arcade, hemos implementado una versión emocionante y atractiva del Tic-Tac-Toe utilizando el lenguaje funcional Elm. Con una interfaz de usuario intuitiva y controles simples, podrás sumergirte en este clásico juego y desafiar a tus amigos.

#### 3.1.Descripción

El juego Tic-Tac-Toe implementado en Elm es una versión del clásico juego de tres en raya. El juego se representa mediante una cuadrícula de 3x3, donde los jugadores (representados por X y O) colocan sus fichas en las casillas. Se cuentan con las siguientes funciones principales:

- *Model*: El modelo del juego se define como un registro llamado Model, que contiene la información sobre el tablero de juego (board), el jugador actual (currentPlayer) y el tamaño del tablero (boardSize).
- *Inicialización*: La función initHelper se encarga de inicializar el modelo del juego con un tablero vacío y el tamaño especificado. Por defecto, el tamaño del tablero se establece en 3x3.
- *Actualización*: La función update define cómo se actualiza el modelo en respuesta a los mensajes recibidos. Por ejemplo, cuando se produce un clic en una casilla del tablero (Clicked), se actualiza el tablero y se cambia al siguiente jugador.
- *Lógica del juego*: La función computeWinner determina si hay un ganador en el juego actual. Comprueba si algún jugador ha completado una línea horizontal, vertical o diagonal en el tablero.
- *View*: La función view define cómo se muestra el juego en la interfaz de usuario. Utiliza elementos de la biblioteca Element de Elm para representar las casillas del tablero y otros elementos visuales, como los botones para aumentar o disminuir el tamaño del tablero.

#### 3.2.Instrucciones de Juego

El juego Tic-Tac-Toe en el sistema Pixel Arcade está diseñado para permitir a los jugadores interactuar utilizando clics del mouse. Esto significa que los usuarios pueden utilizar el cursor del mouse para seleccionar las casillas del tablero en las que desean colocar sus fichas.

Para jugar al Tic-Tac-Toe con el mouse, sigue estos pasos:

- Iniciar el juego  
Abre el juego Tic-Tac-Toe desde el menú principal del sistema Pixel Arcade.
- Tablero de juego

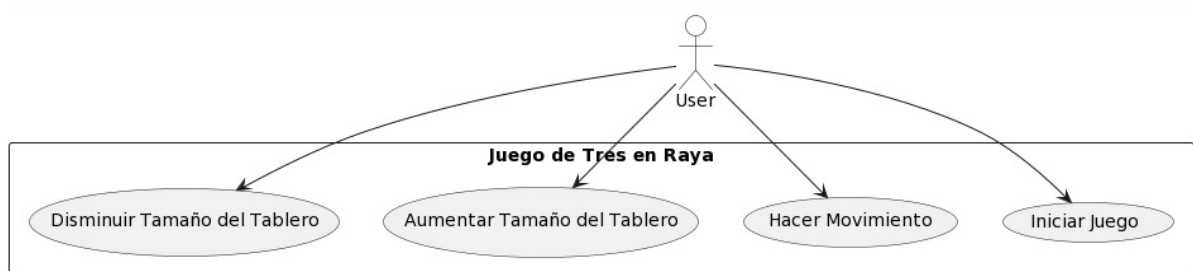
En la pantalla del juego, verás un tablero de 3x3, dividido en nueve casillas. Cada casilla representa una posición en la que puedes colocar tu ficha.

- **Selección de Casilla**  
Para colocar tu ficha en una casilla específica, coloca el cursor del mouse sobre esa casilla y haz clic izquierdo.
- **Turnos de juego**  
El juego alternará automáticamente los turnos entre los jugadores. El primer jugador colocará fichas X y el segundo jugador fichas O.
- **Resultado del juego**  
El juego determinará automáticamente si se ha alcanzado un resultado, como una victoria, un empate o si el juego continúa. Se mostrará un mensaje en pantalla indicando el resultado.

X	O	X	The winner is: X  +  -
X	O	O	
X			

Ten en cuenta que únicamente podrás colocar tu ficha en las casillas validas y el juego se encargará de validar los movimientos.

### 3.3.Casos de Uso



#### Descripción:

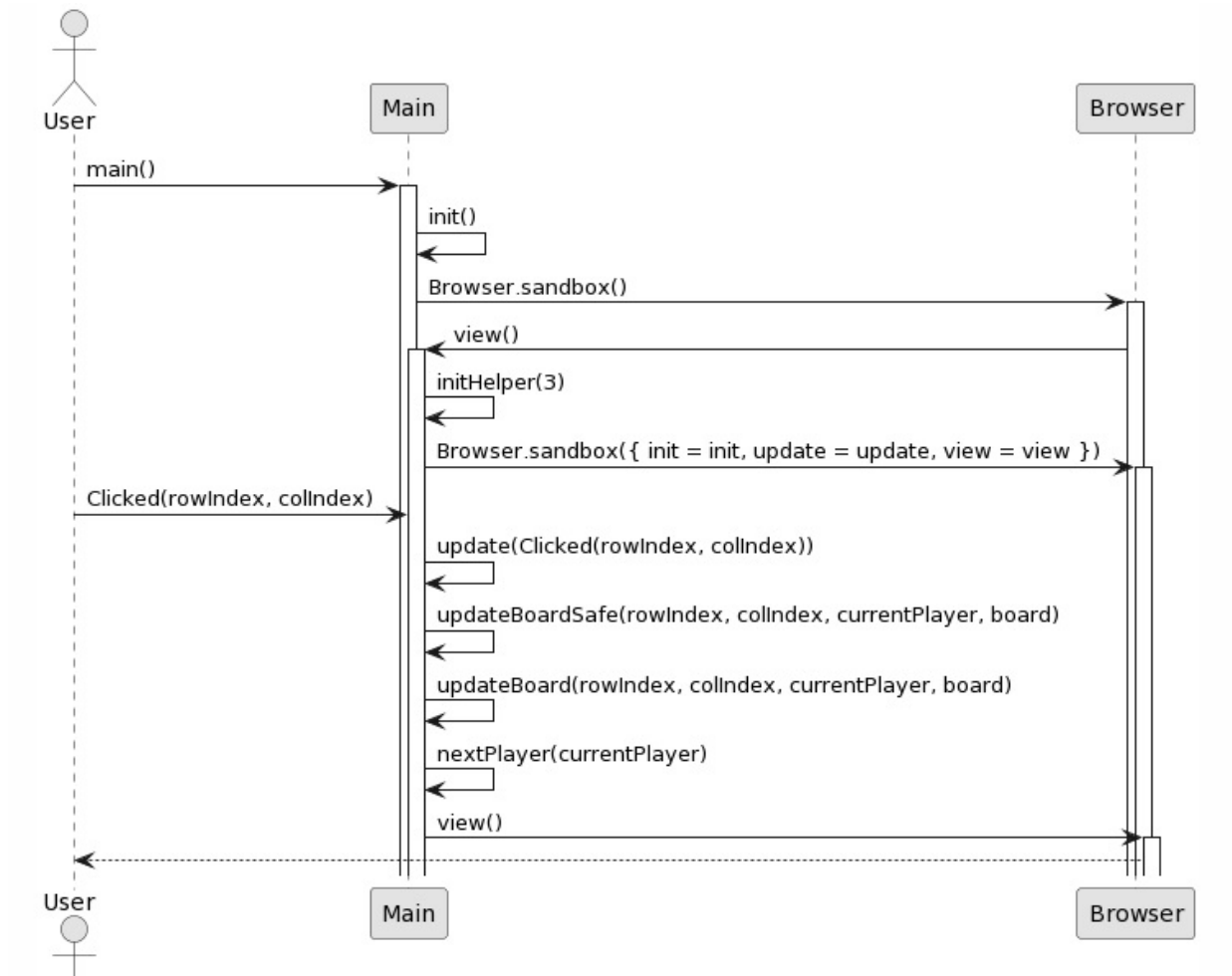
**Actor:** Tanto si es una persona, como un sistema, se representa con el dibujo de una figura humana esquemática, en este caso el actor principal será el Usuario que jugara Tictactoe.

**Sistema:** El sistema al que se refiere el caso de uso tiene forma de rectángulo, representa el entorno en el cual se desarrollan los casos de uso.

**Caso de uso:** Se muestra como una elipse que suele incluir un texto describiendo brevemente el proceso.

1. Iniciar Juego.
2. Hacer Movimiento.
3. Aumentar Tamaño del tablero.
4. Disminuir Tamaño del tablero.

### 3.3.Diagrama de Secuencia



## 4.SNAKE

Pixel Arcade presenta una versión clásica del juego Snake, donde controlarás una serpiente hambrienta y deberás guiarla a través de un tablero lleno de deliciosos alimentos. Tu objetivo es comer la mayor cantidad posible de alimentos mientras evitas chocar contra las paredes y tu propia cola.

En este juego, te enfrentarás al desafío de mantener a la serpiente en movimiento constante. A medida que consumes alimentos, la serpiente crecerá en longitud, lo que aumentará la dificultad.

### 4.1.Descripción

El juego Snake implementado en Elm permite a los jugadores controlar la serpiente utilizando las teclas de dirección (Up, Down, Left, Right) para cambiar la dirección de movimiento. Se cuentan con las siguientes funciones principales:

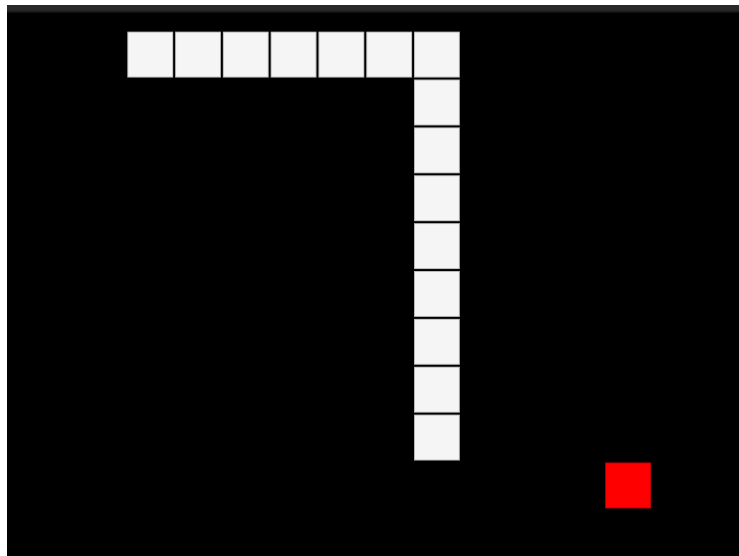
- *type alias GameState*: Define el estado del juego, que incluye la información de la serpiente y la ubicación de los alimentos.
- *type alias Snake*: Representa la serpiente en el juego, con una dirección y una lista de segmentos que forman su cuerpo.
- *type alias Location*: Representa una ubicación en el tablero del juego, con coordenadas x e y.
- *initialState*: Define el estado inicial del juego, incluyendo la serpiente y la ubicación inicial del alimento.
- *snakeDirectionFromKeyCode*: Un diccionario que mapea códigos de tecla a direcciones de serpiente. Permite obtener la dirección de la serpiente a partir de una tecla presionada por el jugador.
- *update*: La función update maneja los mensajes que se reciben y actualiza el estado del juego en consecuencia. En este caso, se procesan los mensajes de KeyDown para cambiar la dirección de la serpiente.
- *advanceSnakeOneStep*: Esta función avanza la serpiente un paso en el juego. Actualiza la ubicación de la cabeza de la serpiente según su dirección actual y verifica si ha comido el alimento. Además, maneja los límites del tablero y el crecimiento de la serpiente.
- *view*: La función view define la representación visual del juego en HTML. Utiliza SVG para dibujar el tablero, la serpiente y el alimento.
- *subscriptions*: Esta función establece las suscripciones para el juego, como capturar eventos de teclado (onKeyDown) y generar un mensaje periódico (Time.every) para mover la serpiente.
- *main*: La función principal del juego, que configura el programa Elm y define la inicialización, la vista, la actualización y las suscripciones.

En general, este código implementa las funcionalidades básicas para controlar la serpiente en el juego Snake utilizando las teclas de dirección. Cada vez que se produce un cambio de dirección, se actualiza el estado del juego y se avanza la serpiente en consecuencia.

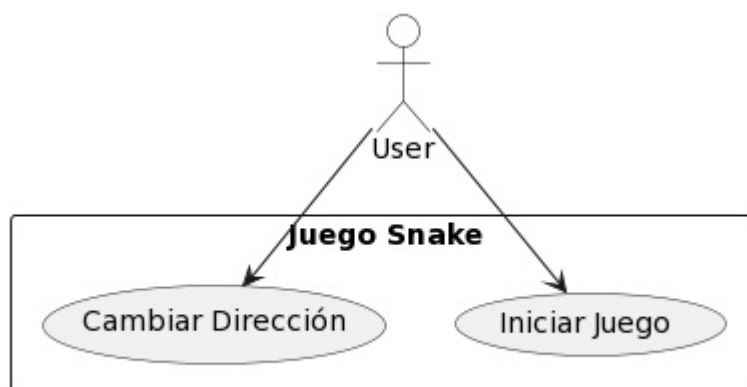


#### 4.2.Instrucciones de Juego

- La serpiente comienza con una longitud inicial y se mueve continuamente en una dirección predeterminada.
- El jugador puede cambiar la dirección de la serpiente utilizando las teclas de flecha (arriba, abajo, izquierda, derecha) o las teclas W, A, S, D.
- Cada vez que la serpiente come un bloque rojo, crece un segmento y se suma puntos al puntaje.
- El juego termina si la serpiente colisiona consigo misma o con los bordes del tablero.
- El objetivo es obtener la puntuación más alta posible evitando colisiones y comiendo tantos bloques rojos como sea posible.



#### 4.3.Casos de Uso



**Descripción:**

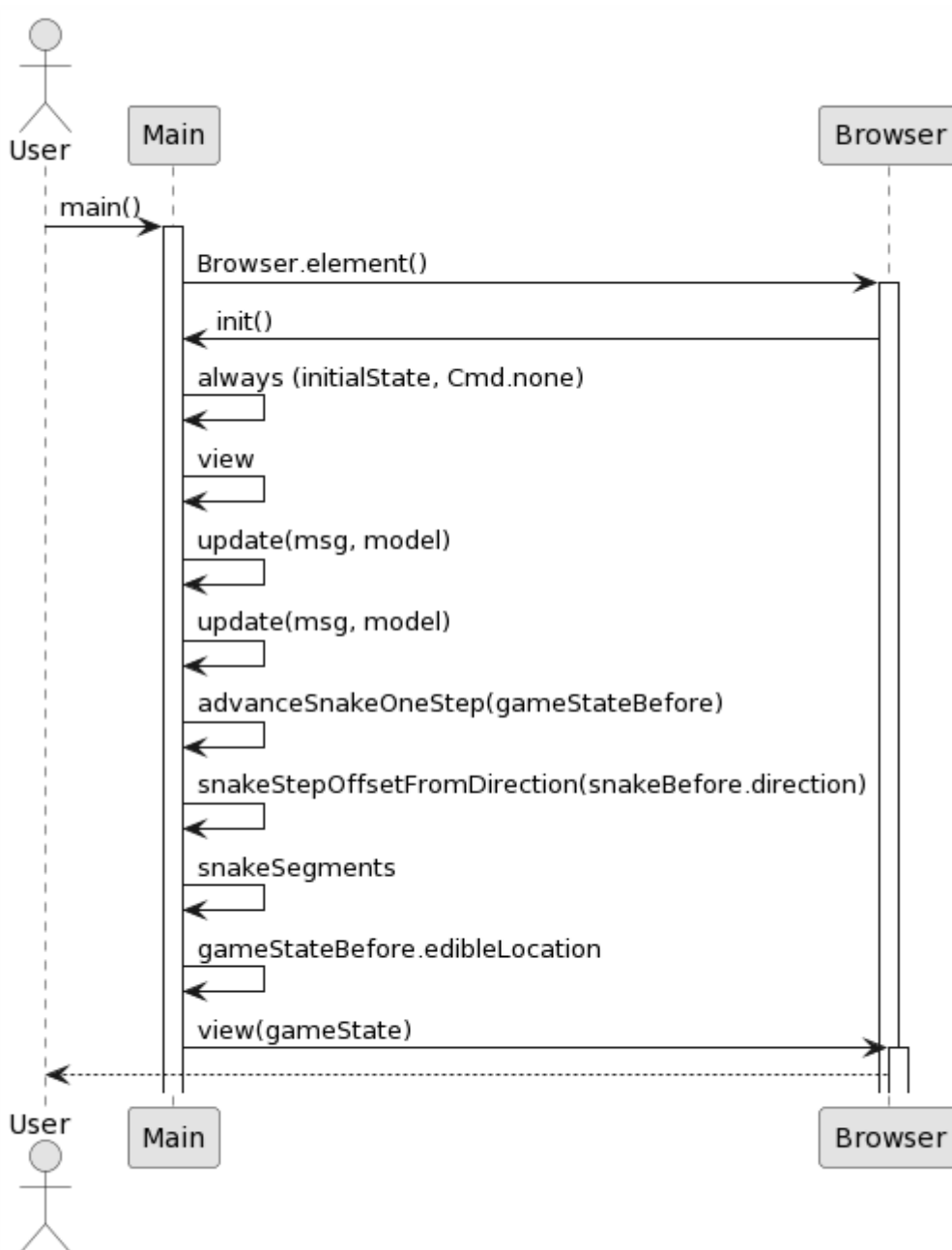
**Actor:** Tanto si es una persona, como un sistema, se representa con el dibujo de una figura humana esquemática, en este caso el actor principal sera el Usuario que jugara Snake.

**Sistema:** El sistema al que se refiere el caso de uso tiene forma de rectángulo, representa el entorno en el cual se desarrollan los casos de uso.

**Caso de uso:** Se muestra como una elipse que suele incluir un texto describiendo brevemente el proceso.

1. Iniciar Juego.
2. Cambiar direccion.

#### 4.3.Diagrama de Secuencia



## 5. AHORCADO

### 5.1.Descripción:

El juego del Ahorcado consiste en adivinar una palabra oculta letra por letra. Por cada suposición incorrecta, se añade una parte del cuerpo al "colgado". El objetivo del juego es adivinar la palabra antes de que se complete la figura del colgado.

- *Model*: El modelo del juego se define como un registro llamado Model. Contiene información sobre el estado del juego, como el progreso de la palabra a adivinar, el número de intentos fallidos, el estado del juego (jugando, ganado o perdido), la lista de palabras a adivinar, etc.
- *Msg*: Los mensajes del juego se definen como una enumeración llamada Msg. Incluye mensajes como GuessedChar (suposición de letra), Restart (reiniciar el juego), NewRandomNumber (generar un nuevo número aleatorio), NextWord (pasar a la siguiente palabra), KeyPressed (tecla presionada).
- *Init*: La función init inicializa el estado del juego cuando se inicia por primera vez. Configura el modelo inicial y emite comandos para generar un nuevo número aleatorio.
- *Update*: La función update es responsable de actualizar el modelo en función de los mensajes recibidos. Maneja la lógica del juego, como evaluar las suposiciones de letra, cambiar el estado del juego y generar nuevas palabras aleatorias.
- *View*: La función view se encarga de representar el juego en la interfaz de usuario. Define cómo se muestran los elementos del juego, como el estado de la palabra a adivinar, el dibujo del colgado, el teclado virtual y los mensajes de resultado.
- *Subscriptions*: La función subscriptions se encarga de suscribirse a eventos, en este caso, se suscribe a los eventos de presionar teclas para capturar las suposiciones del jugador.

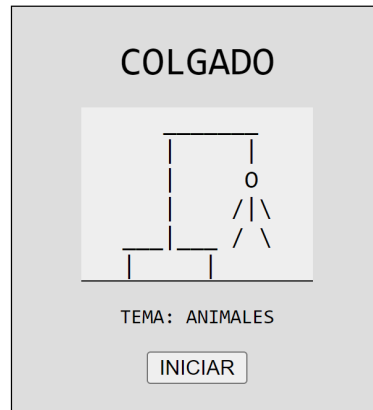
Estas son las funciones principales del juego Ahorcado implementado en Elm utilizando el código proporcionado. El juego permite a los jugadores adivinar palabras ocultas, con un límite de intentos incorrectos antes de que se complete la figura del colgado.

### 5.2.Instrucciones de Juego

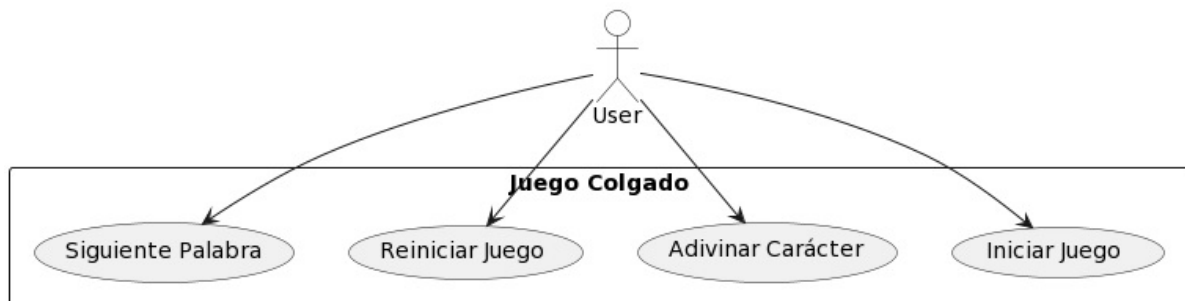
- El juego comienza con un estado inicial en el que el juego no ha comenzado. El usuario puede hacer clic en el botón "INICIAR" para comenzar el juego. Cuando se inicia el juego, se elige una palabra aleatoria de la lista de palabras proporcionada. La palabra se enmascara, mostrando solo la primera y última letra.
- El jugador puede hacer conjeturas al hacer clic en las teclas en el teclado visualizado en la pantalla. Si el jugador adivina correctamente una letra de la palabra, se revela en su posición correspondiente en la palabra enmascarada. Si la conjetura es incorrecta, se muestra una parte adicional del dibujo del ahorcado.
- El jugador continúa haciendo conjeturas hasta que adivina la palabra completa o hasta que se muestra el dibujo completo del ahorcado. Si el jugador adivina la palabra completa, se

muestra un mensaje de "CORRECTO". Si se muestra el dibujo completo del ahorcado, se muestra un mensaje de "INCORRECTO" junto con la palabra correcta.

- Después de finalizar una partida, el jugador puede hacer clic en el botón "SIGUIENTE" para iniciar una nueva partida con una palabra aleatoria diferente.



### 5.3.Casos de Uso



#### Descripción:

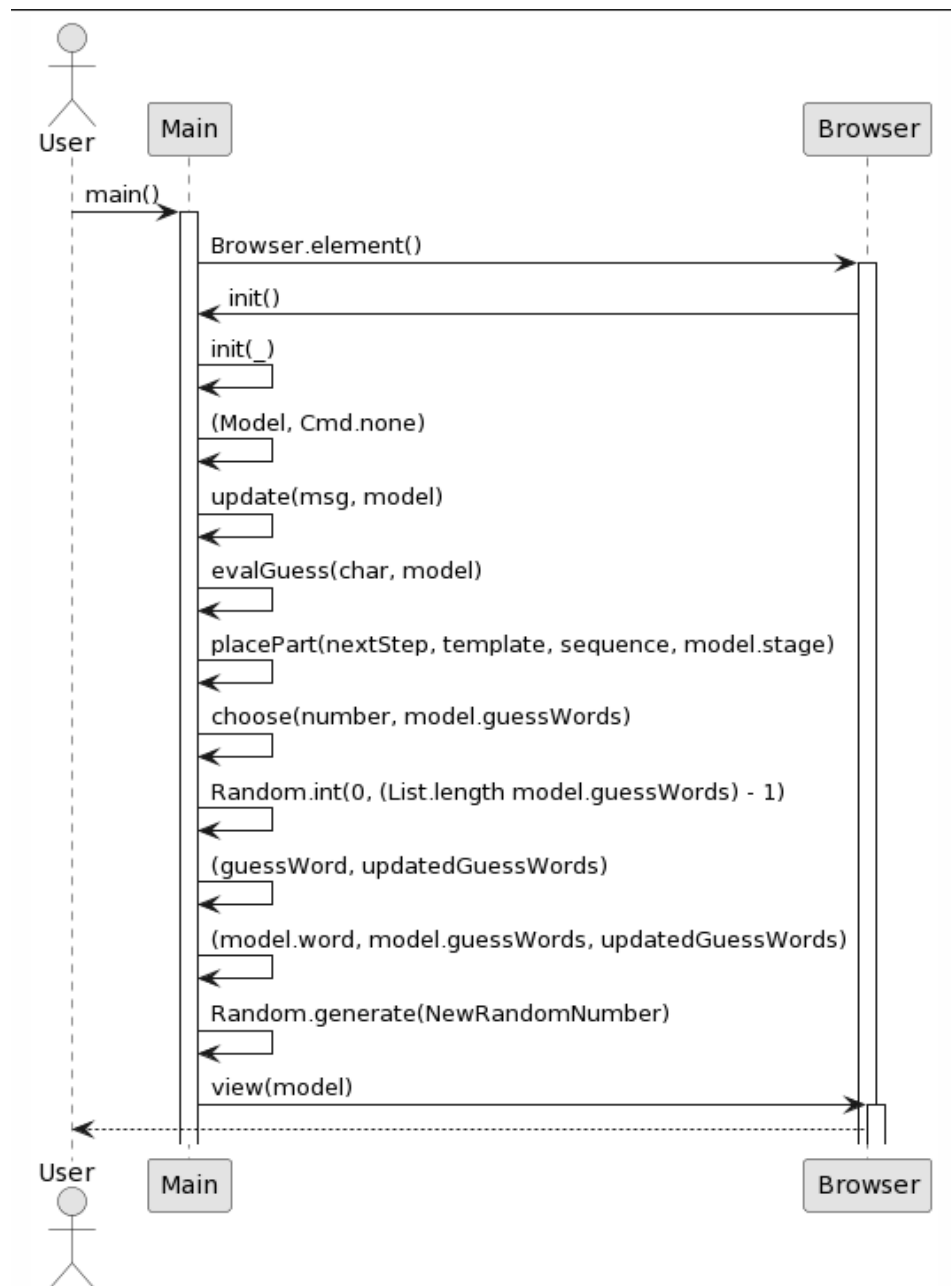
**Actor:** Tanto si es una persona, como un sistema, se representa con el dibujo de una figura humana esquemática, en este caso el actor principal sera el Usuario que jugara Ahorcado.

**Sistema:** El sistema al que se refiere el caso de uso tiene forma de rectángulo, representa el entorno en el cual se desarrollan los casos de uso.

**Caso de uso:** Se muestra como una elipse que suele incluir un texto describiendo brevemente el proceso.

1. Iniciar Juego.
2. Adivinar caracter.
3. Reiniciar juego.
4. Siguiete Palabra.

### 5.3.Diagrama de Secuencia



## 6. CONFIGURACIÓN DE ENTORNO

1. Instala Elm:
2. Visita el sitio web oficial de Elm en <https://elm-lang.org/> y sigue las instrucciones de instalación para tu sistema operativo específico.
3. Crea una nueva carpeta para tu proyecto:
4. Crea una carpeta en tu sistema de archivos donde deseas almacenar tu proyecto Elm.
5. Inicializa un nuevo proyecto Elm:
6. Abre una terminal o línea de comandos en la carpeta de tu proyecto.
7. Ejecuta el siguiente comando para inicializar un nuevo proyecto Elm:
  - a. `elm init`
8. Instala paquetes:
  - a. Utiliza el gestor de paquetes de Elm, llamado "elm" para instalar los paquetes que necesitas en tu proyecto. Por ejemplo, si deseas instalar el paquete "elm/http" para hacer solicitudes HTTP, ejecuta el siguiente comando:
    - i. `elm install elm/http`
9. Crea un archivo de código fuente:
10. Crea un nuevo archivo con la extensión ".elm" en tu carpeta de proyecto y escribe tu código Elm en ese archivo.
11. Compila y ejecuta tu proyecto:
12. Utiliza el siguiente comando para compilar y ejecutar tu proyecto Elm
  - a. `elm reactor`
  - b. Esto iniciará el servidor de desarrollo de Elm en tu máquina local.
13. Accede a tu aplicación Elm:
14. Abre un navegador web y visita la siguiente URL: <http://localhost:8000/>.
15. Verás una lista de archivos en tu proyecto Elm. Haz clic en el archivo principal de tu aplicación para abrirlo y verlo en el navegador.