

The last challenge

Git Gud & Git Commit-ed Incorporated



Creadores:

- Jonay Guillen Rodriguez
- Hugo Ramos Almenar
- Aytami Castro Rodríguez
- Alejandro Afonso Barber

Curso;

1º DAW B

Compañía;

- Git Gud & Git Commit-ed Incorporated

INDICE

Git Gud & Git Commit-ed Incorporated	2
INDICE	3
Resumen del proyecto:	4
Diagrama de casos de uso:	5
Requisitos Funcionales:	6
Esto son los requisitos de nuestra aplicación basados en que el usuario tenga una gran experiencia de juego	6
Requisitos no Funcionales:	7
Casos de uso:	8
Caso de uso 1:	8
Caso de uso 2:	9
Caso de uso 3:	10
Caso de uso 4:	11
Diagramas de secuencia:	12
Diagrama de secuencia 1:	12
Diagrama de secuencia 2:	13
Diagrama de secuencia 3:	14
Diagrama de secuencia 4	15
Diagramas de Flujos:	16
Diagrama de flujo 1:	16
Diagrama de flujo 2:	17
Diagramas de Estado:	18
Diagrama de secuencia 3:	18
Diagrama de secuencia 4	19
Mapa de navegación	20

-

Resumen del proyecto:

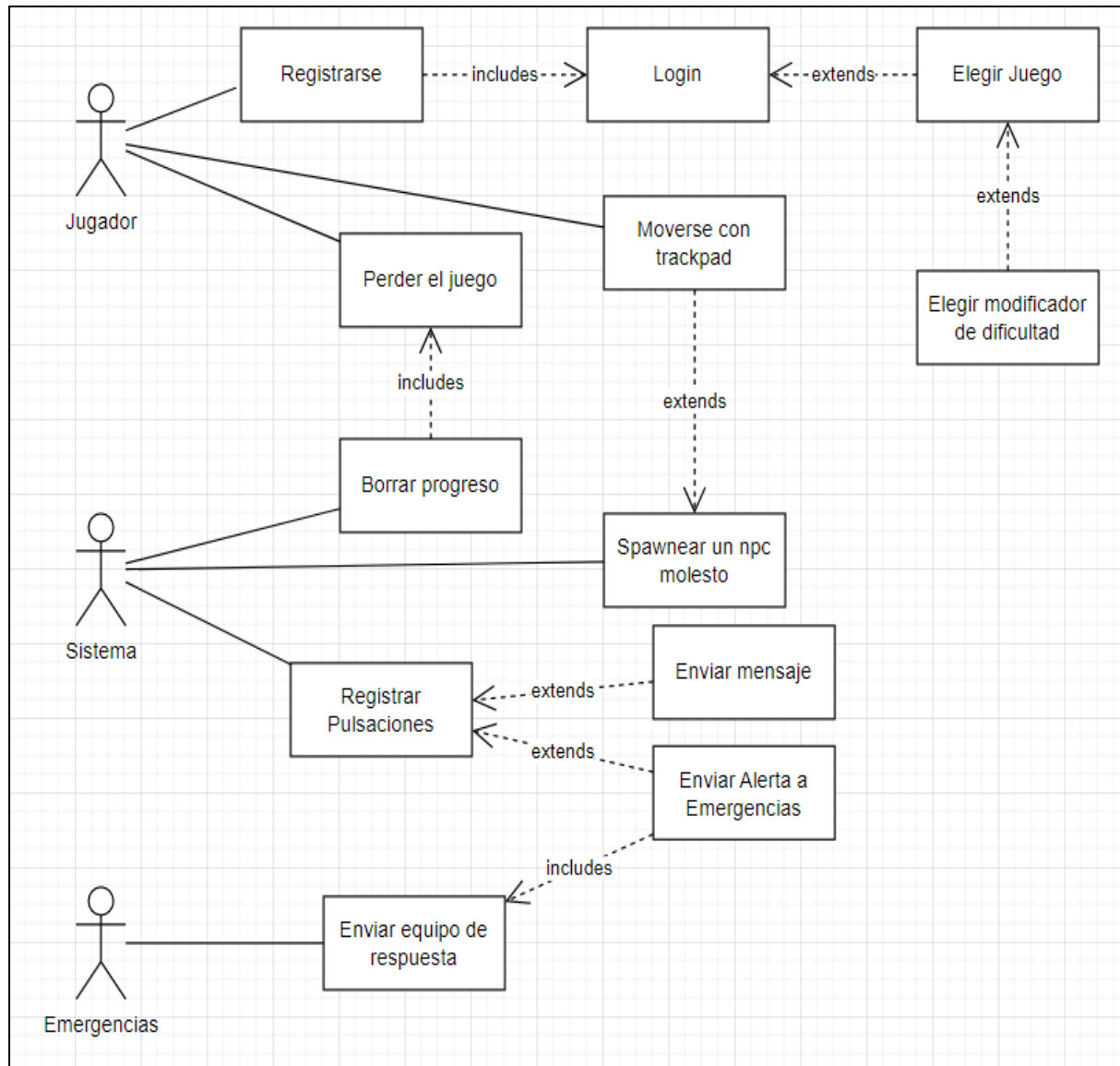
En este proyecto hablaremos sobre nuestro juego para realidad virtual, que se compone por una re-edición de tres grandes juegos, El NeonWhite un frenético juego que pondrá a prueba tu equilibrio y reflejos, Metal Hellsinger para mejorar tu capacidad de ritmo musical y tu coordinación y por último el outlast, un juego donde podrás poner a prueba tu valor y tu capacidad de resolución en los entornos más difíciles, todo esto en realidad virtual con movimiento controlado por un trackpad (como se puede ver figura 1). Reflejando el diseño de esta en los diferentes casos y esquemas,



img1: uso trackpad

Diagrama de casos de uso:

En este diagrama se puede ver reflejadas las funciones más básicas de nuestro programa y cómo se desenvuelven con el usuario.



Requisitos Funcionales:

Esto son los requisitos de nuestra aplicación basados en que el usuario tenga una gran experiencia de juego

Números:	Funcionalidad:
RF1	El usuario se registra
RF2	El usuario elige el juego entre los 3 disponibles
RF3	El juego usa un sistema de progreso en el que si el usuario pierde se elimina el progreso
RF4	El juego registra las pulsaciones
RF5	Si las pulsaciones superan 40 en 15 segundos, se da aviso y si superan 45 se llama a los servicios emergencia
RF6	El usuario se puede mover con el trackpad.
RF7	El juego tiene un NPC detestable que de manera aleatoria grita y desconcentra al jugador
RF8	El usuario puede elegir modificadores de dificultad para la partida
RF9	Cada dos niveles completados, el sistema se bloquea durante 13 minutos para que el usuario descanse
RF10	En los niveles 13, el juego añade un modificador de dificultad aleatorio.
RF11	El sistema almacenará los datos necesarios para poder contactar a Emergencias

Requisitos no Funcionales:

Nuestros principales objetivos ante el desarrollo del sistema

Números:	Funcionalidad:
RNF1	El juego estará disponible en todas plataformas menos macOS
RNF2	El sistema implementará los requisitos legales para escudarse de demandas o daños recibidos por el usuario
RNF3	El sistema contactará al servidor para cada error crítico del juego y almacenará la información del juego en ese momento
RNF4	El juego estará disponible en Inglés, Español, Galeico, Ewé, Uigur, Turcomano y Kinyarwanda (inclusive)
RNF5	El juego tendrá actualizaciones cada 8 semanas.
RNF6	El tamaño de las actualizaciones estará entre 1 y 10 GB
RNF7	El sistema almacenará las estadísticas de tiempo de juego por sesión para cada usuario

Casos de uso:

Caso de uso 1:

Realizado por: Hugo

Función: Llamada a Emergencia.

Descripción: Si las pulsaciones del usuario superan el rango de taquicardia (160 bpm) el sistema del juego lanzará una alerta a emergencias

Actores	Jugador, Sistema y Emergencias
Precondiciones	<ol style="list-style-type: none">1. El sistema está registrando las pulsaciones2. Las pulsaciones del usuario superan las 160 bpm
Postcondiciones	<ol style="list-style-type: none">1. Los servicios de Emergencias son notificados2. El juego se quedará bloqueado y al cabo de 20 minutos se cerrará el juego
Flow	<ol style="list-style-type: none">1. El sistema registra el valor de las pulsaciones por cada tick2. Si las pulsaciones superan 150 bpm el juego se pausa y se muestra una alerta en pantalla3. Si las pulsaciones superan 160 bpm el juego se bloquea y se manda una notificación a emergencias4. El usuario recibe la notificación de emergencias.5. Tras 20 minutos el juego se cierra
Flow alternativo	<ol style="list-style-type: none">1. El sistema no puede registrar pulsaciones2. El juego se para y lanza una notificación para comprobar si el usuario sigue vivo3. Si no se responde a esta notificación se inicia el protocolo PURGA4. El juego se desinstala del disco5. La información del usuario se borra de la base de datos

Caso de uso 2:

Realizado: Ale

Función: Registro web.

Descripción: El usuario se registra en la web de la empresa

Actores	Jugador, Web, Base de Datos
PreCondiciones	1. El usuario no está registrado
PostCondiciones	1. El usuario finaliza su registro. 2. Accede a las funcionalidades de usuario registrado
Flow	1. El usuario irá a la página web y hará click en el botón de registrarse. 2. El usuario rellenará un formulario. 3. El sistema comprueba que los datos introducidos son válidos. 4. El sistema enviará un mensaje de confirmación en caso de que todo sea válido.
Flow Alternativo	En el paso 3 si el nombre o contraseña no es válido se pide al usuario introducir de nuevo los datos En el paso 4, si el correo ya existe en el sistema se manda un aviso y pide al usuario introducir un correo válido. .

Caso de uso 3:

Realizado: Jonay

Función: Paimaivi

Descripción: Molestar severamente al jugador, cada acción del jugador tiene un 1,3% de generar una molestia al jugador

actores	jugador, sistema
precondiciones	que el jugador este dentro de cualquiera de los juegos
PostCondiciones	
flow	<ol style="list-style-type: none">1. el sistema registra la acción2. se evalúa el juego3. se genera un numero random 100 al 04. si el número es igual o menor a 1,3 se escoge una frase aleatoria de una lista de frases ruidos preparadas para ese juego
flow alternativo	

Caso de uso 4:

Realizado: Aytami

Función: Elegir juego

Descripción: El usuario selecciona uno de los juegos disponibles para jugar.

Actores	Jugador, Sistema de Juegos.
Precondiciones	<ol style="list-style-type: none">1. Los juegos están disponibles para elegir.2. El usuario está registrado.
Postcondiciones	<ol style="list-style-type: none">3. El usuario elige un juego y puede comenzar a jugar.
Flow	<ol style="list-style-type: none">1. El usuario accede al sistema de juegos.2. El sistema muestra los 3 juegos disponibles para elegir.3. El usuario selecciona uno de los juegos.4. El sistema registra la elección del usuario y permite comenzar a jugar el juego seleccionado.
Flow alternativo	<ol style="list-style-type: none">5. Paso 3: Si el usuario decide cancelar la elección, el caso de uso finaliza sin seleccionar ningún juego.

Diagramas de secuencia:

Diagrama de secuencia 1:

Realizado por: Hugo

Función: Llamada a Emergencia.

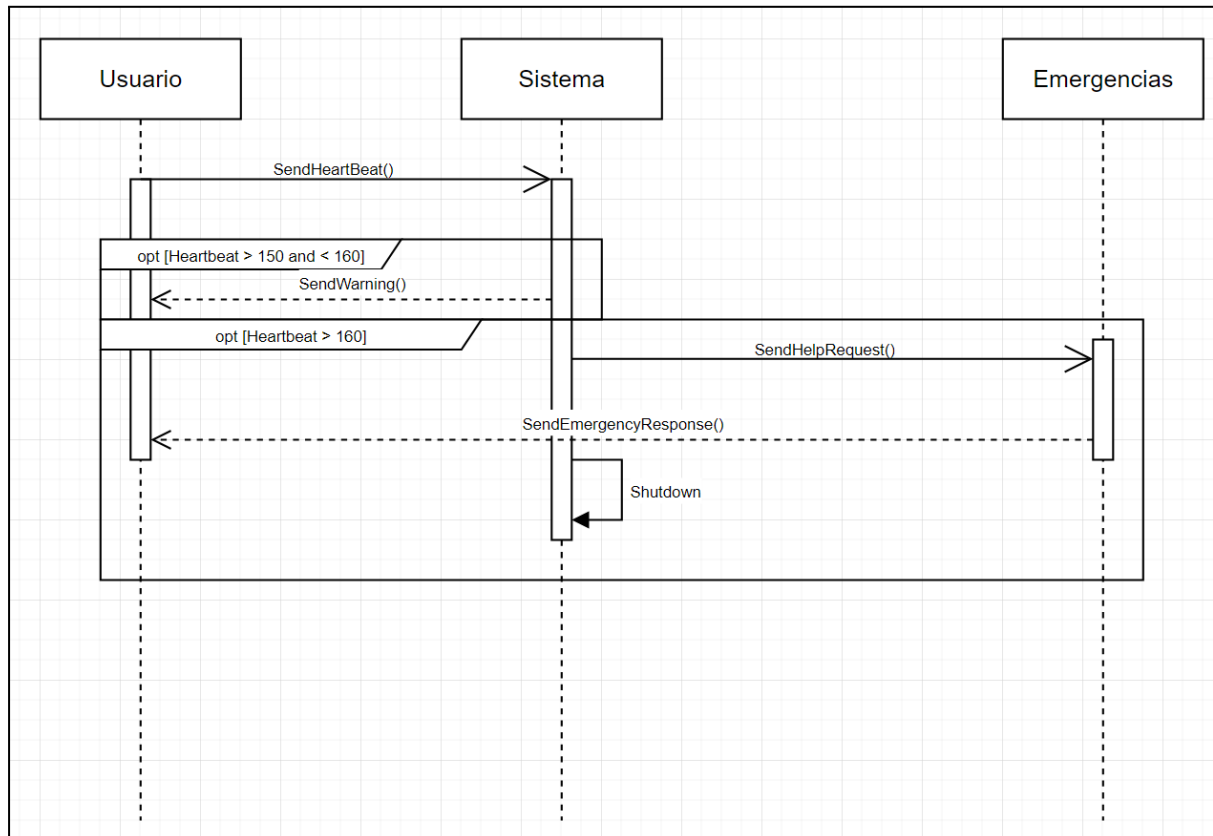


Diagrama de secuencia 2:

Realizado: Ale

Función: Registro web.

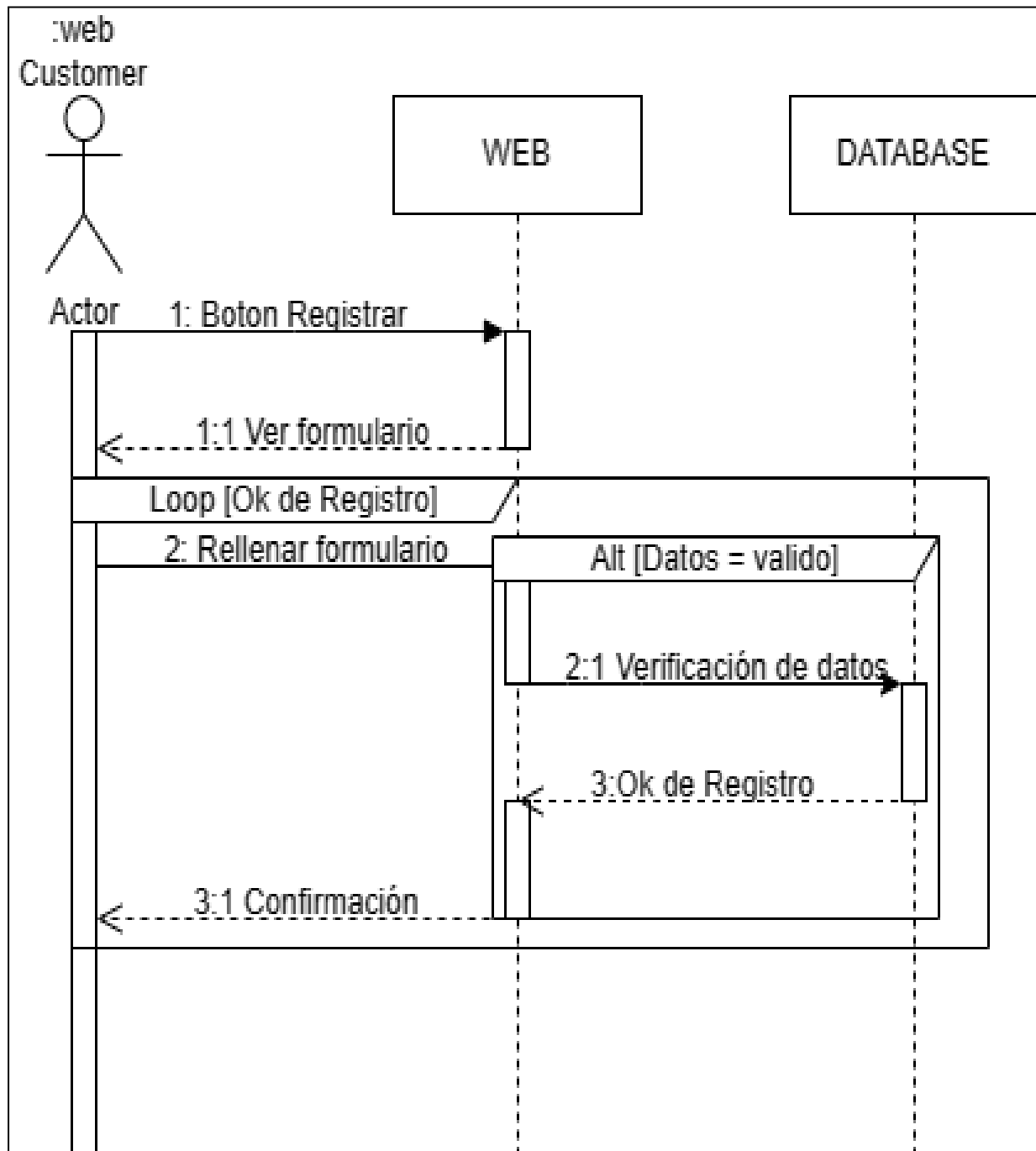


Diagrama de secuencia 3:

Realizado: Jonay

Función: Paimaivi

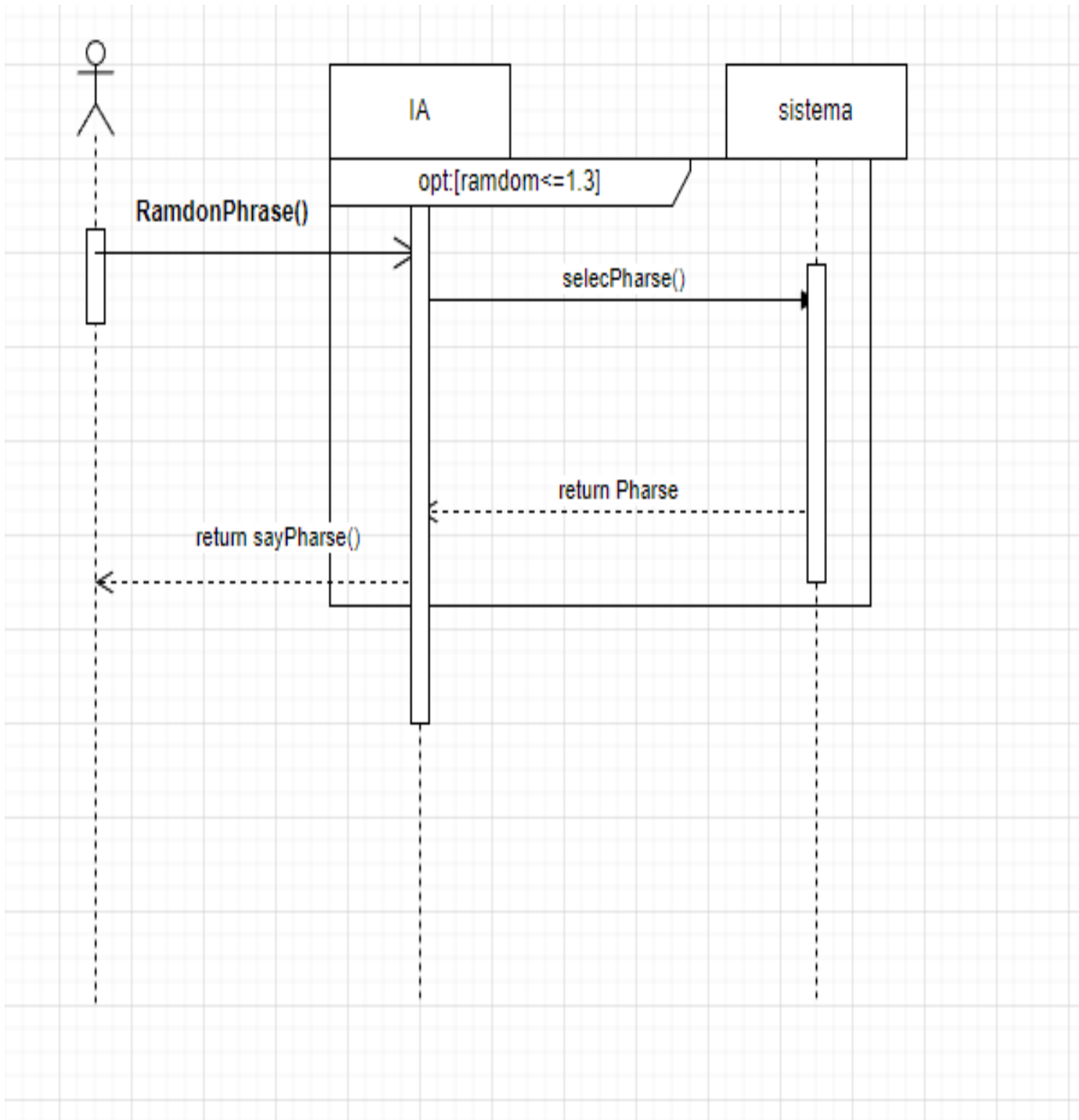
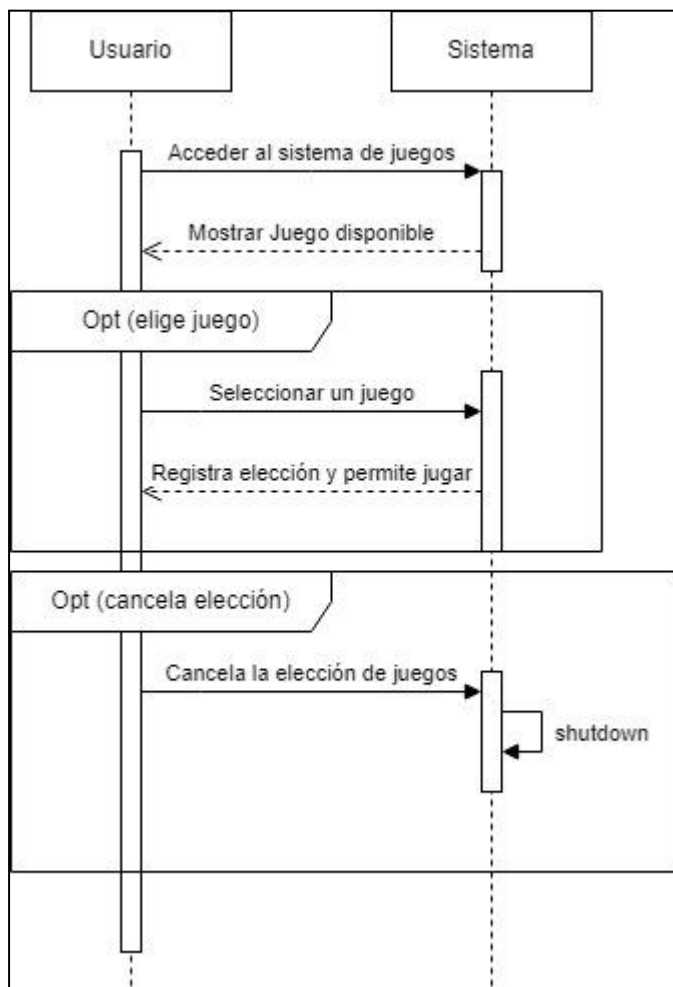


Diagrama de secuencia 4

Realizado: Aytami

Función: Elegir juego



Diagramas de Flujos:

Diagrama de flujo 1:

Realizado por: Hugo

Función: Llamada a Emergencia.

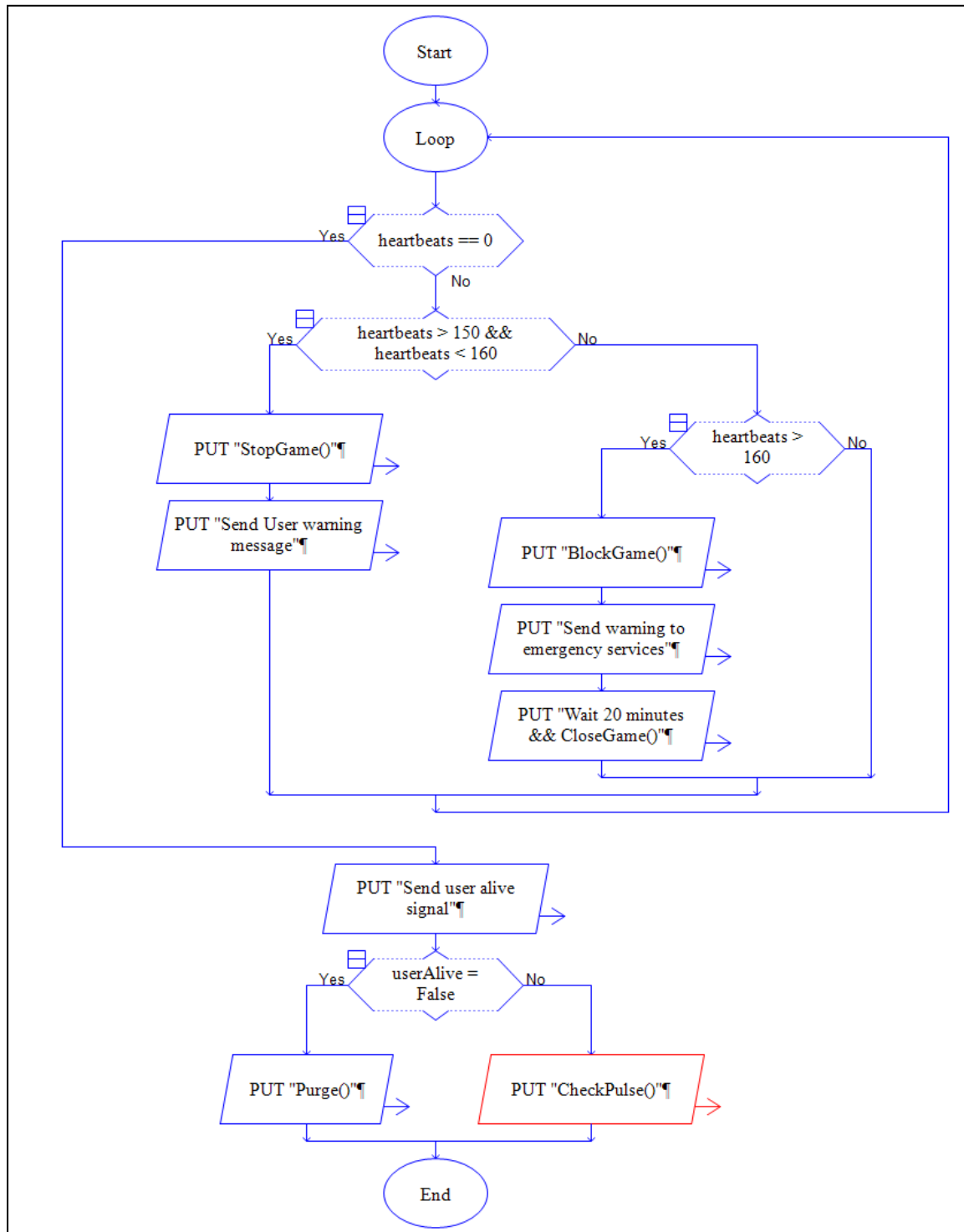
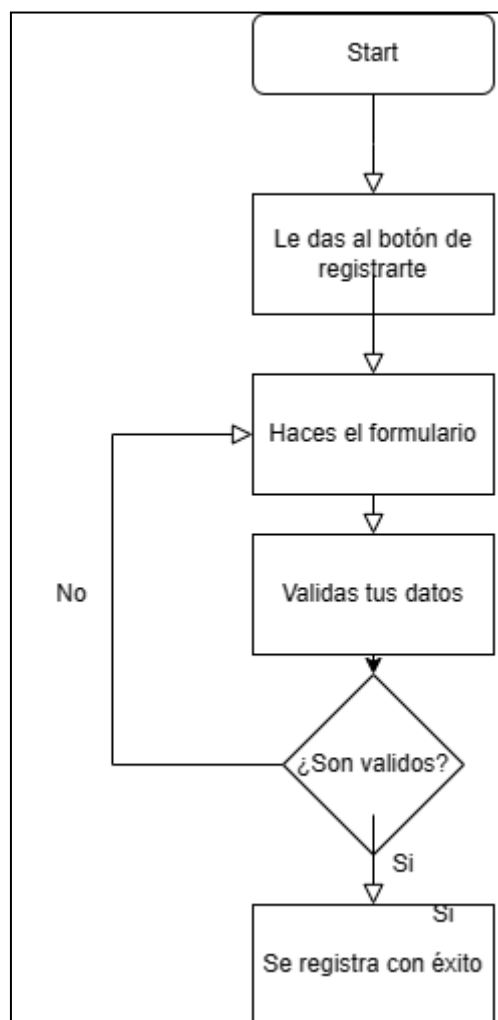


Diagrama de flujo 2:

Realizado: Ale

Función: Registro web.



Diagramas de Estado:

Diagrama de secuencia 3:

Realizado: Jonay

Función: Paimaivi

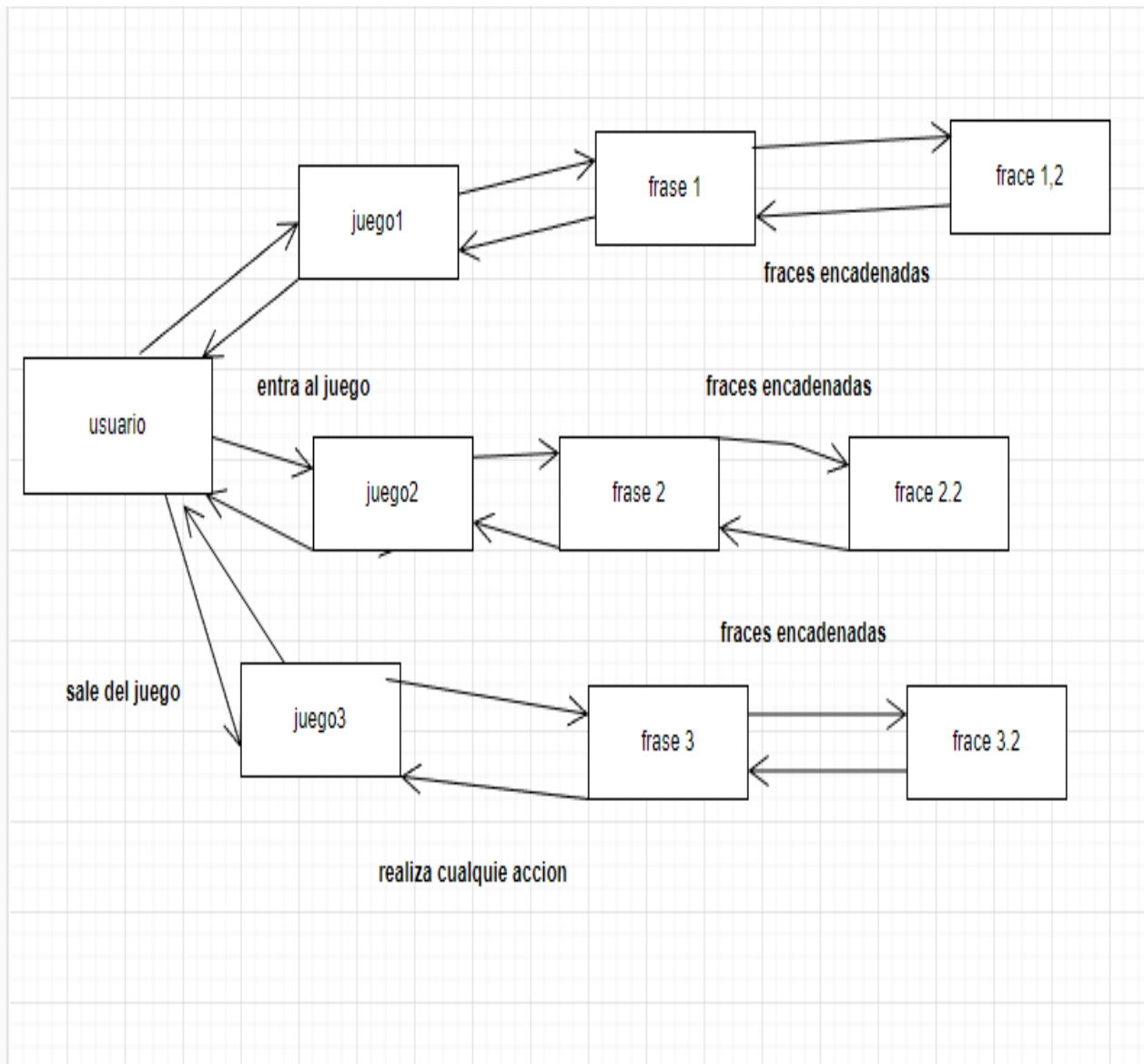


Diagrama de secuencia 4

Realizado: Aytami

Función: Elegir juego



Mapa de navegación

En este parte veremos el mapa de navegación de nuestra aplicación de control y soporte del juego

