Descripción breve

Diseño del Software que se desea desarrollar  
 basado en el Documento de Análisis Funcional.

Diseño

Videojuego de Combate

Iván Gutiérrez González [Analista Funcional][Jefe de proyecto]

Arturo Enrique Gutiérrez Mirandona [Analista Programador]

Jorge Andrés Echevarría [Ingeniero de desarrollo]

Víctor Bartolomé Letosa [QA]

# Tabla de contenido

[0. Tabla de contenido 1](#_Toc160577897)

[1. Diagramas de clases 3](#_Toc160577898)

[1.1. Diagrama de clases 1 3](#_Toc160577899)

[1.2. Diagrama de clases 2 5](#_Toc160577900)

[2. Diagramas de actividad 6](#_Toc160577901)

[2.1. Operativa del programa 6](#_Toc160577902)

[2.2. Registrarse 7](#_Toc160577903)

[2.3. Iniciar sesión 8](#_Toc160577904)

[2.4. Retar otros jugadores 9](#_Toc160577905)

[2.5. Combates 10](#_Toc160577906)

[3. Diagramas de secuencia 11](#_Toc160577907)

[3.1. Diagrama de secuencia 1 11](#_Toc160577908)

[3.2. Diagrama de secuencia del inicio de sesión 12](#_Toc160577909)

[4. Diagramas de estados 13](#_Toc160577910)

[4.1. Diagrama de estado del proceso del desafío 13](#_Toc160577911)

[4.2. Diagramas de estados del combate 14](#_Toc160577912)

[4.3. Diagramas de estados del combate 14](#_Toc160577913)

[4.4. Diagramas de estados del combate 15](#_Toc160577914)

[5. Diagramas de casos de uso 16](#_Toc160577915)

[5.1. Definición de actores 16](#_Toc160577916)

[5.1.1. Jugador 16](#_Toc160577917)

[5.1.2. Operador 16](#_Toc160577918)

[5.2. Diagrama de gestión de cuentas 17](#_Toc160577919)

[5.3. Diagrama de gestión de desafíos 18](#_Toc160577920)

[5.4. Diagrama de gestión de personajes 19](#_Toc160577921)

[5.5. Diagrama de consultas y reportes 20](#_Toc160577922)

[ANEXO 21](#_Toc160577923)

[Datos de ejemplo 21](#_Toc160577924)

[I. Partida 21](#_Toc160577925)

[Prototipos de pantallas 22](#_Toc160577926)

Tabla de ilustraciones

[Ilustración 1 Diagrama de clases 1 3](#_Toc160359935)

[Ilustración 2 Diagrama de clases 2 4](#_Toc160359936)

[Ilustración 3 Diagrama de actividad de operativa del programa 5](#_Toc160359937)

[Ilustración 4 Diagrama de actividad de registrarse 6](#_Toc160359938)

[Ilustración 5: Diagrama de actividad de iniciar sesión 7](#_Toc160359939)

[Ilustración 6: Diagrama de actividad de retar a otro jugador 8](#_Toc160359940)

[Ilustración 7: Diagrama de actividad de los combates 9](#_Toc160359941)

[Ilustración 8 Diagrama de secuencia 1 10](#_Toc160359942)

[Ilustración 9 Diagrama de secuencia 2 11](#_Toc160359943)

[Ilustración 10 Diagrama de estados 1 12](#_Toc160359944)

[Ilustración 11 Diagrama de estados 2 13](#_Toc160359945)

[Ilustración 12 Diagrama de casos de uso 1 15](#_Toc160359946)

[Ilustración 13 Diagrama de casos de uso 2 15](#_Toc160359947)

[Ilustración 14 Pantalla de bienvenida 16](#_Toc160359948)

[Ilustración 15 Pantalla de inicio de sesión 17](#_Toc160359949)

# Diagramas de clases

A continuación, presentamos los distintos diagramas de clases del proyecto, en ellos se podrá apreciar la distinta organización de los objetos que intervienen en el programa, así como las acciones que llevan a cabo en este, también se especificaran y explicaran las relaciones entre los distintos objetos.

## Diagrama de clases 1



Ilustración 1 Diagrama de clases 1

Diagrama de clases principal contiene las clases más importantes que intervienen en el juego, con sus dependencias, herencias, asociaciones… entre ellas. A continuación, explicaremos individualmente cada clase que aparece en la ilustración I.

* *JuegoMP*: Clase encargada de iniciar el juego.
* *GestionInicioSesion*: Clase que se encarga de la gestión y inicio sesión de los usuarios, en ella se podrán registrar tanto los usuarios como los jugadores y por supuesto iniciar sesión para jugar, tiene acceso a ficheros con la información de los usuarios que conforman el juego.
* *Partida*: Esta clase gestiona el juego, contiene la información de la partida, tanto el usuario que se ha conectado, como la información de todos los jugadores existentes en el juego, también contiene una lista con los desafíos que faltan por ser validados por los operadores, esta clase se encarga de salvaguardar los datos principales del juego con cada partida iniciada
* *Usuario*: Clase que contiene la información y de los distintos usuarios del juego, esta es la clase padre de 2 subclases que son *Operador* y *Jugador,* cada una contiene la información exclusiva de su tipo de usurario, así como las acciones que pueden realizar en el juego, recordado que el operador se encarga de gestionar el juego y el jugador de desafiar a otros jugadores.
* *Combate*: Contiene la información de los combates entre jugadores, y en ella por supuesto se realizan estos, esta clase también es manipulada por los operadores antes de que se produzca el combate.
* *Personaje*: Esta tiene la información de los personajes, y se asocia con los usuarios de tipo jugador, contiene los atributos de cada personaje, así como las acciones que se pueden realizar sobre ellos como puede ser la de elegir equipo (armas y armadura).
* *Interfaz serializable*: Mencionamos por último esta interfaz con la que se relaciona la partida, los usuarios y los personajes, esta interfaz es necesaria para hacer que la información del juego sea persistente entre una partida y otra.

## Diagrama de clases 2



Ilustración 2 Diagrama de clases 2

En el diagrama de clases que aparece en la ilustración 2 presentamos la relaciones que tiene la clase *Personaje* mencionada anteriormente con otras clases que únicamente tiene relación con esta y no con las clases antes mencionadas, nuevamente explicamos detalladamente clase a clase.

* *Personajes*: Clase que contiene la información y acciones d ellos personajes y que además de esta se heredan otras 3 clases como son la clase *Vampiro*, *Licántropo* y *Cazador*, que son tipos de personajes cada uno con acciones y características diferentes entre si pero que comparten todos los atributos y metodos que aparecen en la clase padre.
* *HabilidadEspecial*: Cada *Personaje* tiene una, esta cuenta con una serie de atributos que ayudan al *Personaje* en los combates.
* *Esbirro*: Cada personaje tiene un numero indefinido de ellos, se pueden diferenciar tres tipos, los *Demonios*, los *Humanos* y los *Ghouls* que son diferentes clases, todas heredadas de *Esbirro*.
* *Modificador*: Cada *Personaje* tiene varios modificadores unos buenos y otros malos, esos afectan a los atributos de los personajes a la hora de los combates.
* *Equipo*: Clase padre de la que se heredan 2 clases, las *Armas* y las *Armaduras*, cada *Personaje* cuenta con un conjunto de ambas y puede elegir la que desea para tener un distinto desempeño en los combates.

# Diagramas de actividad

Para entender bien el funcionamiento en ciertas partes clave de la aplicación hace falta explicarlas mediante diagramas de actividad. Estas partes clave son el funcionamiento a grandes rasgos de la aplicación, iniciar sesión, registrarse, retar a otro jugador y los combates.

## Operativa del programa

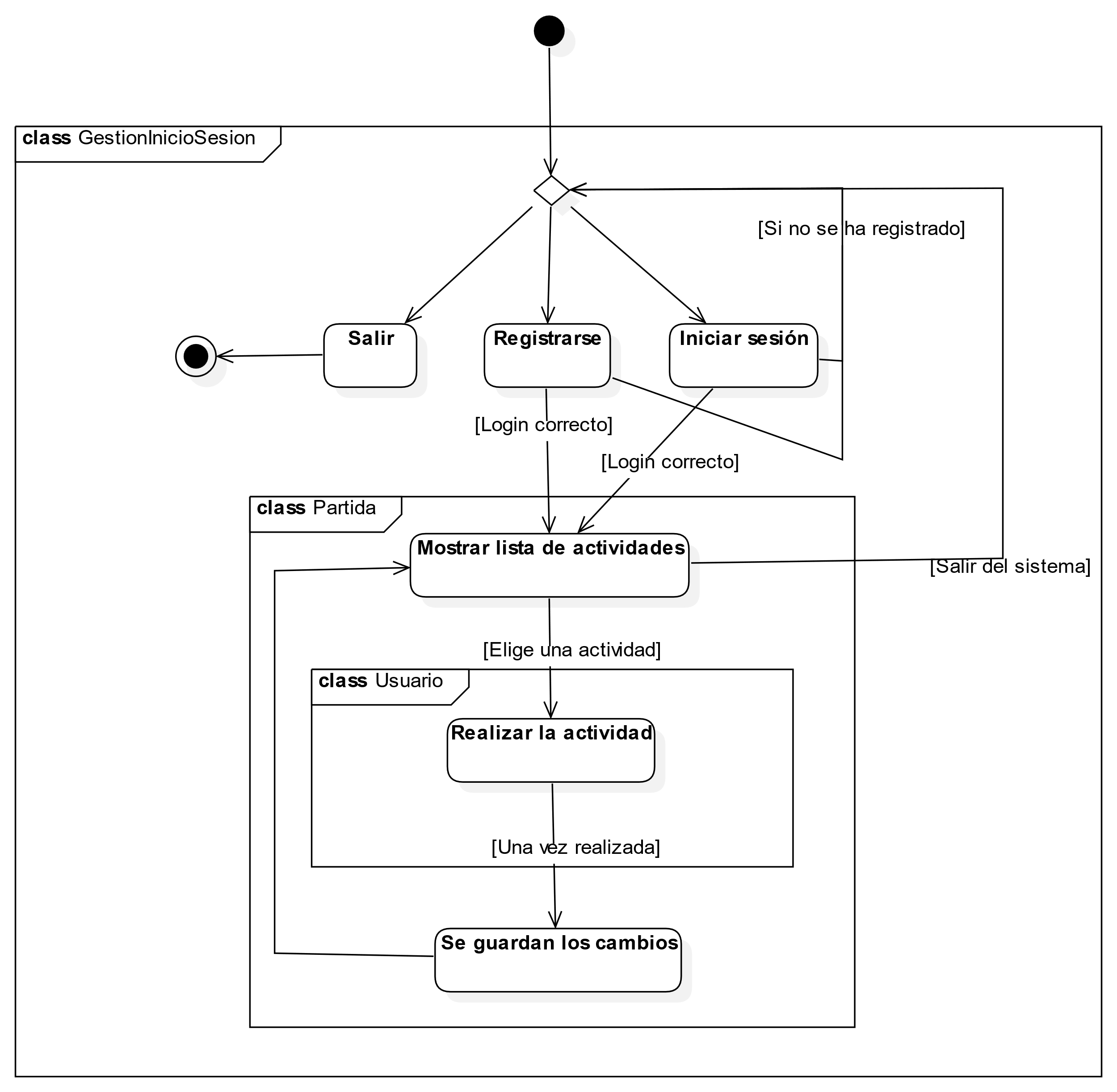


Ilustración 3 Diagrama de actividad de operativa del programa

Este diagrama de actividad representa como va a ser en general el programa en su conjunto. Primero se le mostrará un menú al usuario donde ha de iniciar sesión o registrarse y en caso de que no consiga identificarse de manera correcta o se haya equivocado en la elección del primer menú se podrá volver hacia atrás. Una vez hecho el login correcto se le mostrará una serie de actividades, distintas en función de si se es jugador u operador. Cuando elige la actividad que desea realizar se lleva a cabo y se guardan los cambios. Y así hasta que el usuario decida salir del sistema y en ese caso se cerrará sesión y volverá a la pantalla de inicio para volver a iniciar sesión o registrarse con otra cuenta o cerrar el programa.

## Registrarse

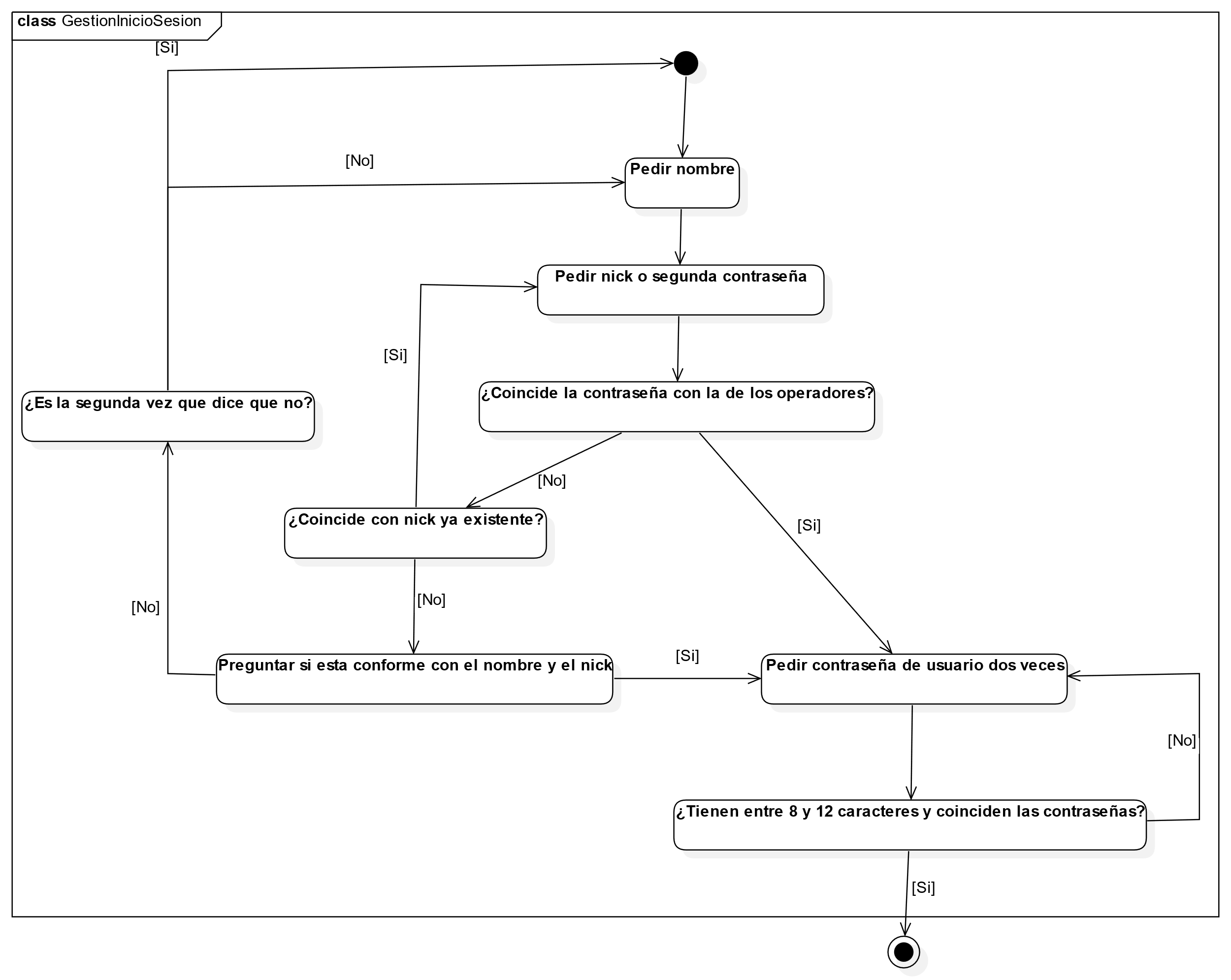


Ilustración 4 Diagrama de actividad de registrarse

Ahora pasamos a ver el diagrama de actividad que muestra como se realizará el registrarse por primera vez un usuario. Primero se pedirá el nombre y después se pedirá el nick o la contraseña especial que se habilitará para operadores, en caso de que no coincida con esta contraseña se supondrá que es un jugador y los caracteres introducidos serán su nick. Se deberá comprobar si ese nick ya existe y en tal caso volver a pedir uno valido y si no se pedirá confirmación al jugador sobre si esta conforme con su nombre y nick elegido. En caso de que se haya preguntado dos veces por la confirmación de nick y nombre se entenderá que el usuario quiere salir y se le volverá a mostrar el menú inicial. Por último, tanto para jugadores como para operadores, se les pedirá una contraseña y confirmar esa contraseña. En caso de que no cumpla con la condición de tener entre 8 y 12 caracteres o que no coincidan las dos contraseñas se volverán a pedir hasta que cumplan las condiciones.

## Iniciar sesión

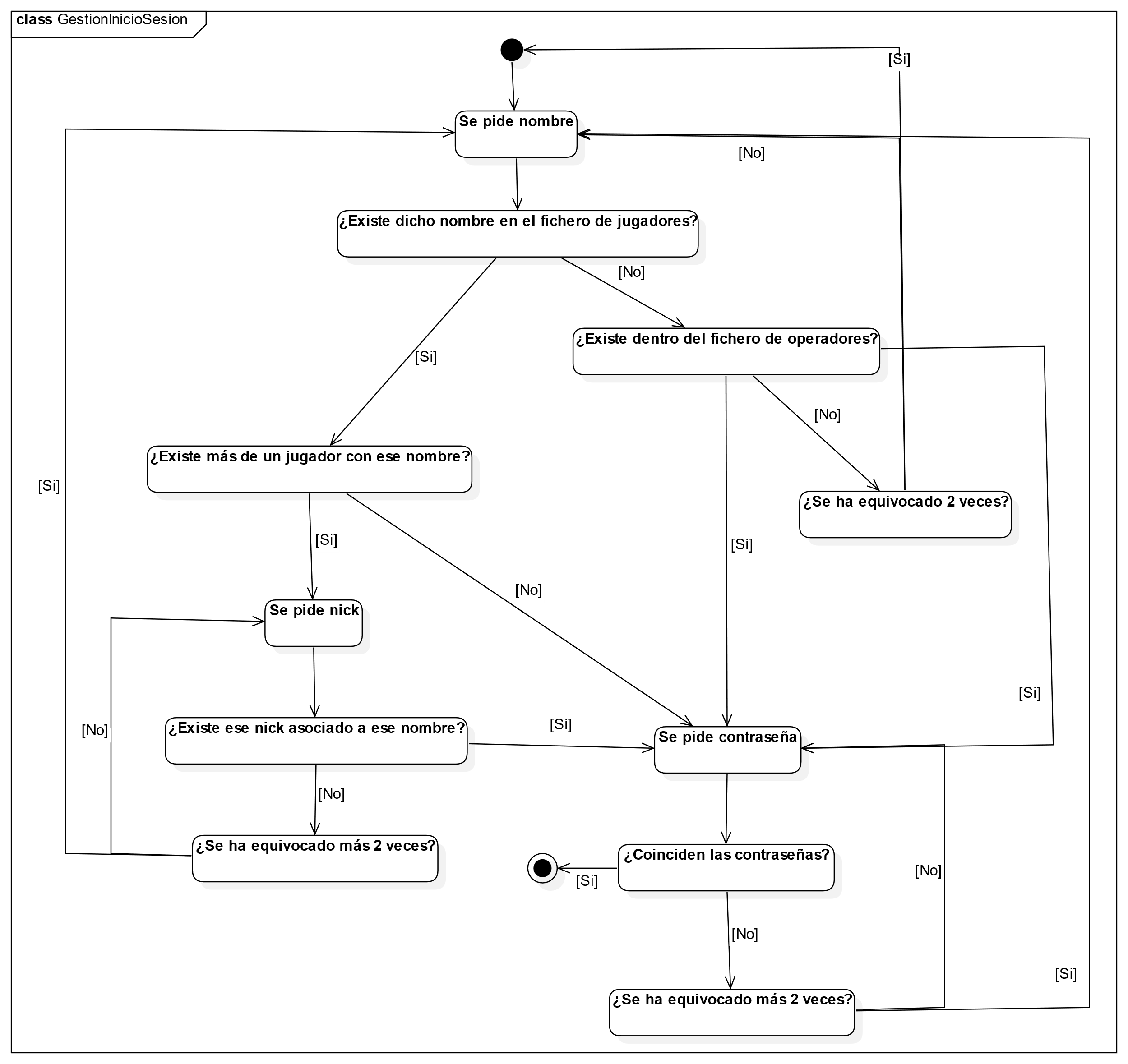


Ilustración 5: Diagrama de actividad de iniciar sesión

Además de explicar como registrarse explicamos cono iniciar sesión. Al igual que al registrarse se pedirá primero el nombre y se comprobará si existe en el fichero que guarda los datos de los jugadores, si no está aquí se buscará en el fichero de los datos de los operadores. En caso de que no exista en ninguno de los dos se comprobará si es la segunda vez que se equivoca, si es que si se le mostrará el menú inicial, y si no volverá a introducir su nombre. Cuando introduzca un nombre valido como jugador se comprobará si hay dos jugadores con el mismo nombre y se le pedirá el nick para distinguirlos. Una vez identificado que jugador u operador es se le pedirá la contraseña para poder logearse correctamente. Tanto cuando se pide el nick o la contraseña si se equivoca dos veces se volverá a pedir el nombre y si no se volverá a pedir nick o contraseña.

## Retar otros jugadores

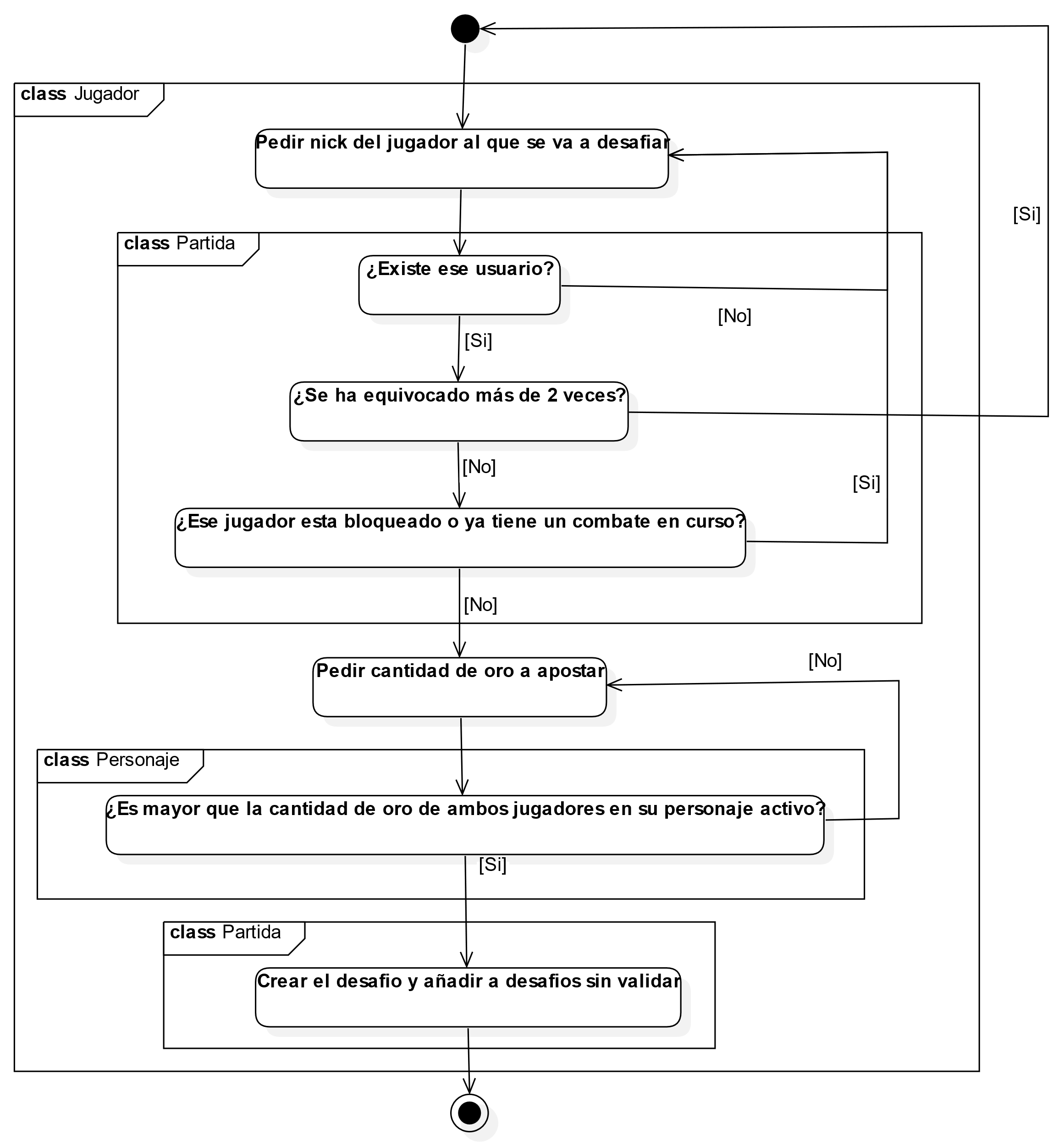


Ilustración 6: Diagrama de actividad de retar a otro jugador

Entrando en la operativa de las actividades del jugador está la de retar a otro jugador. Se empieza pidiendo el nick del jugador al que se va a desafiar y se comprueba si existe. En caso de que no exista se comprobará si es la segunda vez que se equivoca si es que si se volverá al menú del jugador y si no se volverá a pedir el nick. En caso de que exista el nick se verificará si esta bloqueado o tiene ya un combate en curso. Si es que no se pedirá otro nick y si es que si se le pedirá al jugador desafiante una cantidad de oro que quiera apostar. Después se comprobará que los personajes principales de ambos jugadores tengan esa o una mayor cantidad de oro, volviendo a pedir la cantidad de oro en caso de que no se lo puedan permitir. Y por último se creará el desafío con las características recogidas y se añadirá a la lista de desafíos sin validar a la espera de que un operador lo valide.

## Combates

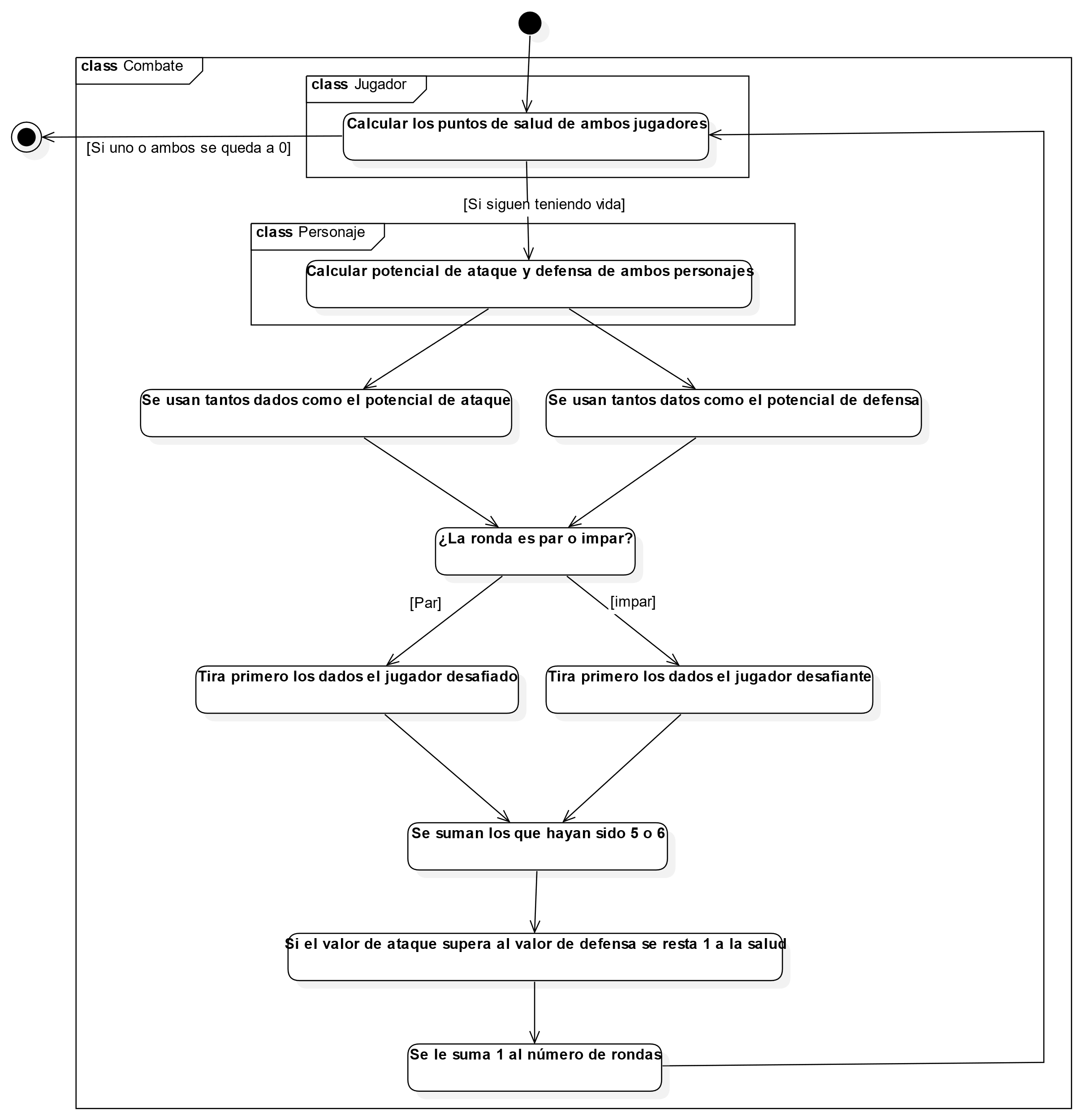


Ilustración 7: Diagrama de actividad de los combates

En el último diagrama de actividad veremos como se realizarán los combates. Los combates consistirán en un bucle que comprueba la vida total de los personajes (solo los personajes porque primero se resta a los esbirros por lo que se quedarán sin vida antes que los personajes), si se queda un jugador o ambos sin vida en su personaje se termina el combate. Si aun tienen vida se calcula el potencial de ataque y defensa de sus personajes más el de los esbirros si quedan. Si es una ronda par primero tira los dados el jugador desafiado y si es impar empieza el jugador desafiante. Después se tiran tantos dados como potencial de ataque y defensa de ambos personajes y los que salgan 5 o 6 se consideran éxito sumando su valor. Como da igual el orden al atacar empieza el jugador desafiante y después el desafiado, y si el valor de ataque supera al de defensa se le resta uno de salud a los esbirros si quedan y si no a los personajes. Y por último se aumenta en uno la ronda y se vuelve a iniciar el bucle.

# Diagramas de secuencia

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed tempus fringilla laoreet. In hac habitasse platea dictumst. Praesent non nibh non tortor eleifend vulputate. Phasellus porttitor eget erat non auctor. Vestibulum mauris diam, feugiat eu nisl eget, iaculis lacinia ex. Praesent vitae tristique nunc. Sed mi ante, consequat vitae libero vel, consectetur consectetur neque. Maecenas sodales et est ut ultrices. Quisque eget nulla aliquam, vulputate purus luctus, venenatis arcu. Duis et lorem vel nulla lacinia volutpat. Donec viverra leo nec egestas rhoncus. Duis consequat eu ante sit amet euismod. Donec in erat vitae nibh consectetur auctor. Fusce vel libero laoreet, ornare ante in, tincidunt diam. Duis id semper elit.

## Diagrama de secuencia 1



Ilustración 8 Diagrama de secuencia 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed tempus fringilla laoreet. In hac habitasse platea dictumst. Praesent non nibh non tortor eleifend vulputate. Phasellus porttitor eget erat non auctor. Vestibulum mauris diam, feugiat eu nisl eget, iaculis lacinia ex. Praesent vitae tristique nunc. Sed mi ante, consequat vitae libero vel, consectetur consectetur neque. Maecenas sodales et est ut ultrices. Quisque eget nulla aliquam, vulputate purus luctus, venenatis arcu. Duis et lorem vel nulla lacinia volutpat. Donec viverra leo nec egestas rhoncus. Duis consequat eu ante sit amet euismod. Donec in erat vitae nibh consectetur auctor. Fusce vel libero laoreet, ornare ante in, tincidunt diam. Duis id semper elit.

## Diagrama de secuencia del inicio de sesión



Ilustración 9 Diagrama de secuencia 2

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed tempus fringilla laoreet. In hac habitasse platea dictumst. Praesent non nibh non tortor eleifend vulputate. Phasellus porttitor eget erat non auctor. Vestibulum mauris diam, feugiat eu nisl eget, iaculis lacinia ex. Praesent vitae tristique nunc. Sed mi ante, consequat vitae libero vel, consectetur consectetur neque. Maecenas sodales et est ut ultrices. Quisque eget nulla aliquam, vulputate purus luctus, venenatis arcu. Duis et lorem vel nulla lacinia volutpat. Donec viverra leo nec egestas rhoncus. Duis consequat eu ante sit amet euismod. Donec in erat vitae nibh consectetur auctor. Fusce vel libero laoreet, ornare ante in, tincidunt diam. Duis id semper elit.

# Diagramas de estados

Con la finalidad de entender a detalle el funcionamiento del sistema, se proporcionarán diagramas de estado ya que estos nos proporcionan una representación visual de los diferentes estados que pueden experimentar los elementos del sistema, así como las transiciones entre estos estados, lo que facilita la comprensión del flujo de interacción y las posibles ramificaciones en cada etapa.

## Diagrama de estado del registro

Ilustración 10 Diagrama de estados 1

Al iniciar el proceso de registro de usuario estaremos en el estado de “Usuario no identificado”, al usuario en proceso de registrarse se le pedirá su nombre el cual procederá a ser guardado, entrando en el estado de “nombre guardado”. Luego procederemos a pedir un nick, en este momento el usuario entra la posibilidad de escribir un código secreto que dará acceso al registro de operador y no al de usuario, dependiendo de si dicho código es introducido por el usuario entraríamos en dos posibles estados “Jugador a registrar” o “Operador a registrar” donde ya ha sido identificado el tipo de usuario que va a ser registrado. El operador ya estará habilitado a introducir la contraseña de su gusto. En el caso del jugador, su Nick tendrá que ser validado, es decir, no puede estar duplicado. En caso de no introducir un Nick valido mas de dos veces se entrará en el estado “operación no valida” que finalmente nos llevará al inicio. En caso de Nick valido será hora de pedir validar la contraseña. En caso de que la contraseña no sea valida más de dos veces se entrará en “operación no valida” y nos llevara al inicio. En caso de que la contraseña esté validada entramos en el estado “usuario registrado” para concluir el registro.

## Diagramas de estados del inicio de sesión

Ilustración 11 Diagrama de estados 2

Al iniciar sesión estamos en el estado de “usuario no identificado”, posteriormente se le pedirá al usuario su nombre, a partir de aquí hay dos caminos. Primero se explicará que sucede si existe un jugador con dicho nombre. En caso de encontrar a un jugador con el nombre dado se entrará en el estado de “Jugador en proceso a ser identificado”, en caso de haber mas de un jugador con ese nombre se le pedirá al usuario su nick, en caso de no existir se entra en el estado “usuario no valido” que nos lleva al inicio del inicio de sesión, en caso contrario entramos en el estado “jugador existente”, posteriormente se le pedirá al jugador que introduzca su contraseña para validarla, si no es correcta entraremos en “usuario no valido” y si lo es entramos en “usuario validado” donde el usuario habrá iniciado sesión. En el caso de que el nombre dado por el usuario no exista en la lista de jugadores entraremos en el estado “operador en proceso a ser identificado”, ahí se verificará si existe en la lista de operadores el nombre introducido, si no existe entramos en “usuario no válido”, en caso contrario entramos en “operador existente” y luego se procederá a pedir y validar contraseña como al jugador para terminar de ejecutar el inicio de sesión.

## 4.3. Diagramas de estados del proceso de retar

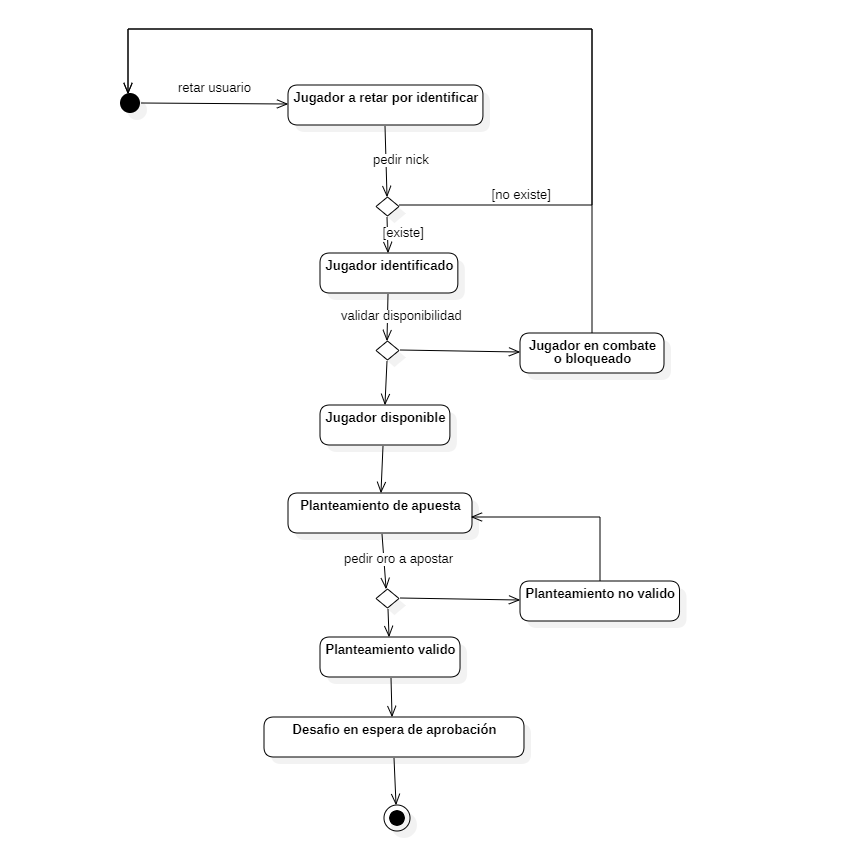


Ilustración 11 Diagrama de estados 2

Cuando un jugador reta a otro, el sistema pasará por múltiples estados. Al iniciar el proceso estaremos en el estado “Jugador por identificar” donde el sistema pedirá al usuario el nick del jugador a retar, si este jugador existe entraremos en el estado de “jugador identificado”. Luego el sistema verificará en qué estado está dicho jugador, si no esta disponible el programa estará en el estado de “Jugador en combate o bloqueado” que nos llevará al inicio de retar a un jugador, en caso contrario el sistema entra en estado “jugador disponible”. Posteriormente entramos en el estado de “planteamiento de apuesta”, donde el jugador propondrá la cantidad de oro a apostar y el sistema verificará si ambos jugadores serán capaces de pagar en caso de perder el combate, dependiendo de esto se entrará en el estado de “planteamiento valido” o “planteamiento valido”. Finalmente, el sistema entra en estado de “desafío en espera de aprobación” donde dicho desafío ha sido creado y quedará en espera a ser aprobado por un operador.

## 4.4. Diagramas de estados del combate

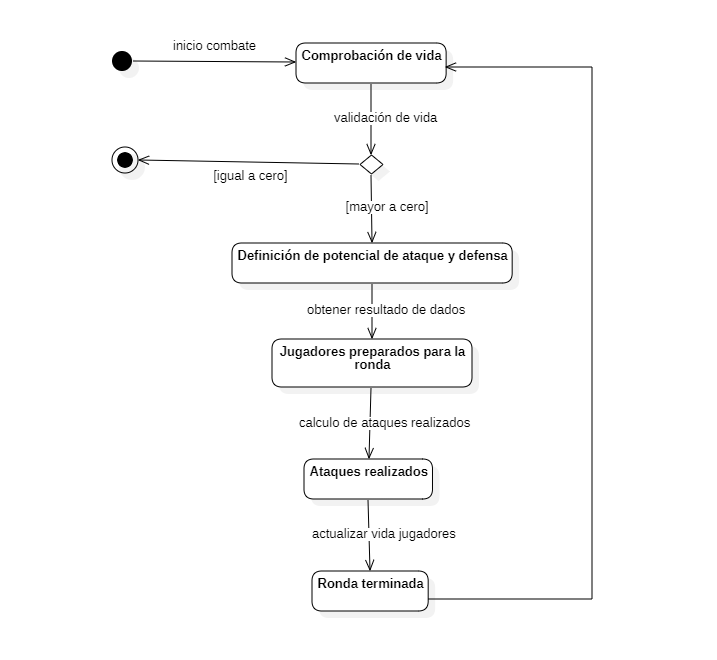


Ilustración 11 Diagrama de estados 2

El combate pasará por varios estados durante su ejecución, estos estados están atados a un bucle ya que el combate acabará cuando uno de los jugadores quede con su vida en cero. Al iniciar el bucle siempre entraremos en una fase de comprobación del estado de vida de ambos jugadores, si una de estas llega a cero se finalizará el combate. Posteriormente, el combate entrará en la fase de definición de potenciales, tanto de ataque como de defensa que posteriormente se utilizaran para verificar si los jugadores perdieron puntos de vida, esto último ocurrirá en la fase de ataques realizados. Finalmente entramos en la fase de ronda terminada, donde no queda nada más que actualizar el numero de ronda y volver a el primer estado de la ronda, es decir, el estado de comprobación de vida.

# Diagramas de casos de uso

El propósito de esta sección consistirá en establecer las acciones disponibles para los diversos usuarios dentro del sistema. Esto se detalla principalmente mediante la representación en diagramas de casos de uso.

## Definición de actores

### Jugador

El jugador, como tipo de usuario registrado que participa activamente en las actividades relacionadas con el juego, desempeñará varias acciones importantes. Primero, él tiene la capacidad de desafiar a otros jugadores, los jugadores serán retados por su Nick. Además, cuando es desafiado por otro usuario, tiene la facultad de aceptar o rechazar el desafío.

También, el jugador podrá gestionar a sus personajes. Esta gestión implica la selección y configuración del equipamiento de sus personajes, así como la creación de nuevos personajes y la eliminación de los existentes según sea necesario.

Por último, el jugador cuenta con acceso a los resultados de los combates. A través de un historial de combates, puede visualizar los detalles de estos enfrentamientos, lo que le permite analizar su rendimiento y evolución en el juego.

### Operador

El operador, es el tipo de usuario registrado que como ocupación se encargará de gestionar el sistema. En esencia el operador es un administrador del sistema, este estará encargado de validar múltiples aspectos del juego durante la duración de este. El operador se encargará de:

* Validación de desafíos: El operador verifica la autenticidad y conformidad de los desafíos propuestos por los jugadores, garantizando que estos cumplan con los requisitos establecidos.
* Bloqueo de usuarios: En caso de que algún jugador viole las normativas del juego, el operador puede bloquear su cuenta. Además, podrá desbloquear usuarios previamente restringidos, según sea necesario.
* Edición de atributos de personajes: Los operadores tienen la capacidad de modificar cualquier aspecto relacionado con los personajes, incluyendo la adición de esbirros, armas y armaduras, así como la configuración de sus habilidades y debilidades.

## Diagrama de gestión de cuentas

En el siguiente diagrama, se representa la gestión de cuentas de los diferentes usuarios en el sistema. Los usuarios tienen la opción de registrarse en el sistema como jugadores o como operadores. Al llevar a cabo el proceso de registro, a los jugadores se les asignará automáticamente un número de identificación único generado por el sistema, mientras que a los operadores no se les asignará dicho número.

Además, los usuarios tendrán la capacidad de realizar las siguientes acciones:

* Iniciar sesión en el sistema.
* Cerrar sesión para finalizar su sesión activa.
* Darse de baja en el sistema en caso de que así lo deseen, lo que implicará la eliminación de su cuenta y datos asociados.

Diagrama, Esquemático

Descripción generada automáticamente

Ilustración 13 Diagrama de gestión de cuentas

## Diagrama de gestión de desafíos

En el siguiente diagrama se ilustra la gestión de desafíos desde la perspectiva de los jugadores y los operadores. Los jugadores podrán aceptar y rechazar los desafíos planteados por otros jugadores. Además, el operador del sistema estará encargado de validar los desafíos, garantizando que cumplan con los criterios preestablecidos. Además, el operador podrá tomar medidas, como bloquear jugadores si lo considera necesario para mantener el orden y la integridad del sistema.

Diagrama, Esquemático

Descripción generada automáticamente

Ilustración 13 Diagrama de gestión de desafíos

## 5.4. Diagrama de gestión de personajes

En el diagrama que sigue, se detalla la gestión de personajes por parte de los usuarios. Los jugadores poseen la capacidad de crear y eliminar personajes, así como de equiparlos con armas y armaduras. Por otro lado, el operador del sistema puede intervenir en aspectos más detallados de los personajes, tales como la adición de esbirros, armas, armaduras, y la configuración de fortalezas y debilidades.

Diagrama

Descripción generada automáticamente

*Ilustración 14 Diagrama de gestión de personajes*

## 5.5. Diagrama de consultas y reportes

Durante la sesión activa del jugador, este podrá realizar consultas acerca de su estado en el sistema. Esto implica acceder al ranking global y revisar su historial de combates, así como la información asociada a cada uno de ellos

Diagrama

Descripción generada automáticamente

*Ilustración 15 Diagrama de consultas y reportes*

# ANEXO

## Datos de ejemplo

### Partida

* Mapa de jugadores
  + Jugador
    - Nombre: Andrés
    - Nick: ElBombas
    - Contraseña: Cuenta1]
    - Nº Registro: T15LP
    - Combates: False
    - Bloqueado: False
    - Personaje activo (el ejemplo se ve en lista de personajes)
* Lista de desafíos sin validar
  + Desafío
    - Jugador retado: Andrés
    - Jugador retador: Federico
    - Oro apostado: 20.5
    - Rondas: 1
    - Fecha: 20-01-2024
    - Ganador: null
    - Jugador con esbirros sin derrotar:
      * Andrés
      * Federico
    - Modificadores: null
    - Valido: True
* Lista de personajes
  + Personaje
    - Nombre: Vampiro
    - Habilidades especiales:
      * Nombre: Disciplina
      * Valor ATC: 1
      * Valor DFS: 2
      * Coste: 1.5
    - Armas: null
    - Armas activas:
      * Nombre: Garras
      * Modificador: 2
      * Mod DFS: 1
      * Tipo: True
    - Armaduras: null
    - Armadura activa:
      * Nombre: Caparazón
      * Modificador: 3
      * Mod ATQ: 1.5
    - Esbirros:
      * Nombre: Ghoul
      * Salud: 20
      * Dependencia: 1
    - Oro: 120
    - Salud: 10
    - Poder: 5
    - Debilidades:
      * Nombre: Roce con el agua
      * Valor: 0.1
    - Fortalezas:
      * Nombre: Ego
      * Valor: 0.5
    - Edad: 150
    - Sangre: 10

## Prototipos de pantallas

Tabla

Descripción generada automáticamente con confianza bajaEs de vital importancia que la interfaz del sistema sea fácil de entender y usar, proporcionando una experiencia de usuario intuitiva. Esto implica que la disposición de los elementos en la interfaz debe usar elementos visuales claros y descriptivos para guiar a los usuarios. La navegación y el acceso a las distintas funciones debe ser sencillo, y la interfaz debe ofrecer retroalimentación visual inmediata sobre las acciones del usuario.

Ilustración 14 Pantalla de bienvenida

*Diagrama

Descripción generada automáticamente*

Ilustración 15 Pantalla de inicio de sesión

En las ilustraciones superiores podemos observar un esquema a seguir para la orientación de la interfaz de la aplicación.

-En ***ilustración 14***se puede encontrar un espacio grande en la parte superior destinado al título y dos botones en la zona inferior que representan dos posibilidades comunes entre operadores y usuarios (iniciar sesión y registrarse).

-En ***ilustración 15*** se puede destacar en la parte superior un espacio reservado para el logo identificativo de la aplicación. Dos espacios centrales destinados a registrar los datos de los usuarios a la hora de iniciar sesión. Por último, un botón de envío del formulario en la parte inferior.