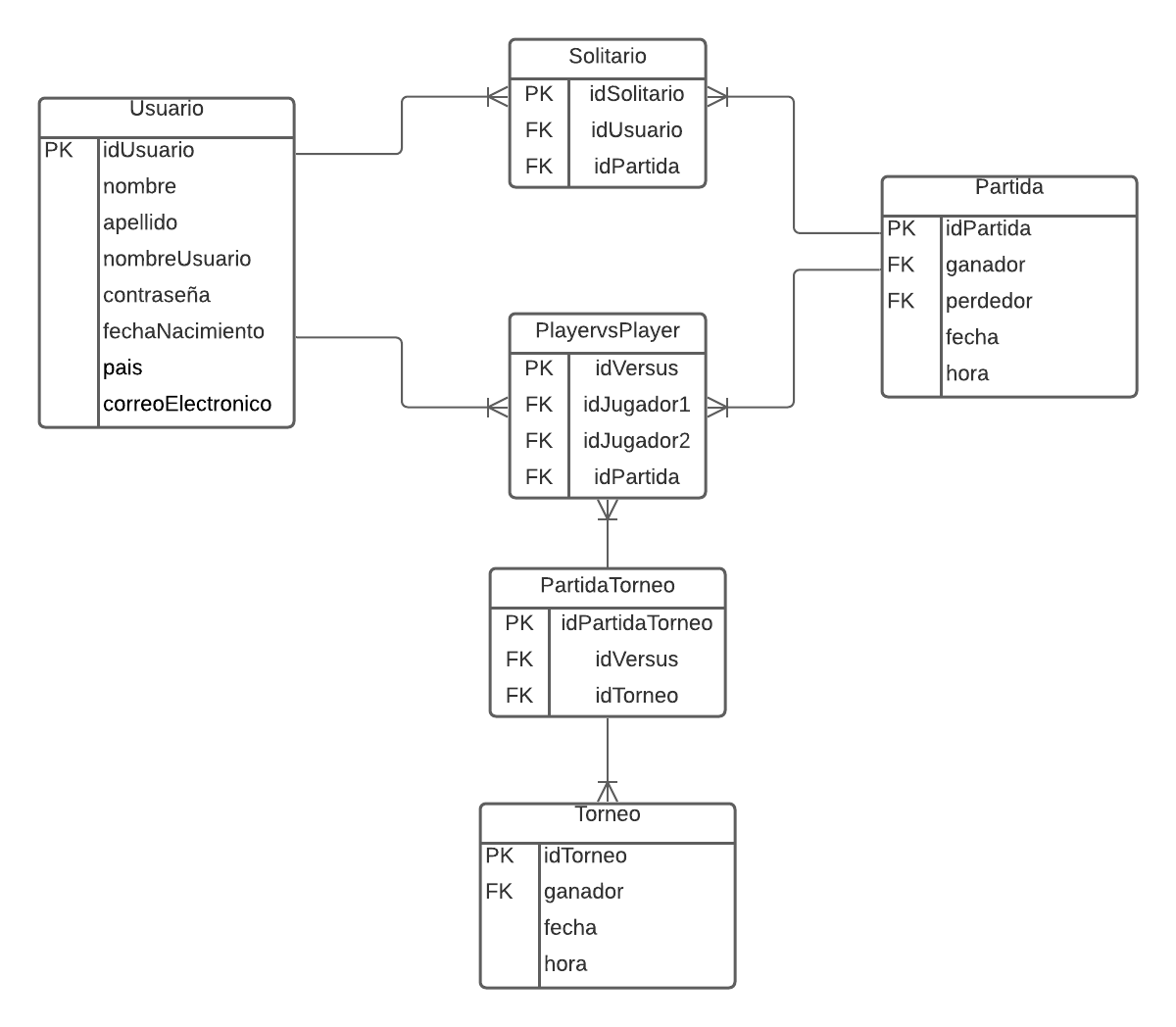
**IGamesGT: Aplicación web “Othello”**

1. **Definición de la solución**
   1. **Objetivo General:** Desarrollar una aplicación web para el juego de mesa Othello, aplicación web que ofrece funcionalidades pensadas para que el usuario se lleve una mejor experiencia en el juego.
   2. **Objeticos específicos:**
      1. Promover el juego de mesa Othello, que tenga un mayor alcance y que se popularice más.
      2. Agregar funcionalidades que facilitan el acceso al juego (Partidas On Line).
      3. Conseguir que el jugador tenga una mejor experiencia con el juego de mesa, por medio de las múltiples funcionalidades que tendrá la plataforma.
   3. **Alcance del proyecto:** La aplicación web inicialmente estará disponible en el idioma español, por lo que se espera que se llegue a todas las personas de habla hispana y de manera secundaria a las personas con un nivel de español básico ya que el juego será bastante intuitivo.

Por otra parte, los modos de juego aseguran un alcance mayor, con la participación de múltiples jugadores en los torneos que se pueden crear o bien un alcance menor con la opción de juego Player vs Player. Esto debido a que si un jugador de habla no hispana, conoce las reglas del juego puede hacer uso de la aplicación web ya que será bastante intuitivo el uso de la misma.

* 1. **Requerimientos iniciales del sistema**
     1. **Requerimientos funcionales (Ref#).**
        1. El sistema en su página de inicio mostrará un breve párrafo hablando del juego y de la plataforma misma.
        2. El sistema permitirá el registro de usuarios, que posteriormente tendrán acceso a la aplicación web.
        3. El sistema tendrá un sistema de autentificación, donde le pedirá al usuario sus credenciales que previamente indicó en el registro.
        4. El sistema será capaz de crear partidas de diferentes características: Player vs Maquina, Player vs Player.
        5. El sistema será capaz de crear torneos, para que puedan participar múltiples jugadores.
        6. El sistema será capaz de cargar partidas mediante archivos XML que contienen los datos/progreso de la partida.
        7. El sistema será capaz de guardar partidas en un archivo XML que contendrá los datos/progreso de la partida.
        8. El sistema será capaz de correr el juego de mesas, Othello, con todas las validaciones posibles: Reglas de juego, movimientos permitidos, etc.
        9. El sistema llevará un registro de las partidas del jugador: partidas ganadas, perdidas, empatadas.
        10. El sistema le permitirá al administrador gestionar la información de los usuarios en la base de datos
        11. El sistema será capaz de generar reportes con base a las estadísticas de las partidas del jugador.
        12. El sistema le permitirá al usuario cerrar sesión sin que se pierdan sus datos o progresos.
     2. **Atributos del sistema.** 
        + El sistema se enfocará y asegurará la agilidad entre las acciones generadas por el usuario tales como: Acceso a aplicación, registro de usuario, creación de partidas, movimientos en el tablero. Sin comprometer un tiempo de espera bastante largo y tedioso en la finalización de estas acciones a modo de brindar una experiencia buena.
        + La interfaz de usuario buscará la simplicidad y efectividad, tener una aplicación web bastante intuitiva. Sin saturación visual (gráficos y colores).
        + Diseño web responsivo para al aplicación y compatibilidad con todos los navegadores: Opera, Firefox, Safari, Google Chrome.

1. **Glosario inicial.**
   1. **Othello:** Othello o yang es un juego entre dos personas, que comparten 64 fichas iguales, de caras distintas que se van colocando por turnos en un tablero dividido en 64 casillas.
   2. **Tablero:** Espacio donde se desarrolla el juego, cuenta con 64 casillas, cada una identificada por un número de fila y una letra por columna.
   3. **Ficha:** Elemento de juego que se emplea en el tablero para el desarrollo del juego de mesa, existen dos grupos; un grupo de color blanco y uno de color negro.
   4. **Registro:** Proceso de recolección de los datos del usuario para su control en la base de datos del juego, se pide información básica: nombre, apellido, fecha de nacimiento, etc.
   5. **Credencial:** Datos del usuario (nombre de usuario/email y contraseña) que se necesitan para la autentificación en el sistema y posteriormente tener acceso a su cuenta en el sistema.
   6. **Usuario:** Player o Jugador, cliente del sistema que en el momento de su registro se le concedió una cuenta para tener acceso a la aplicación web.
   7. **Torneo:** Competencia entre usuarios de la aplicación web, se organizan múltiples partidas en donde el ganador de cada partida es el que pasa a la siguiente ronda hasta obtener un ganador definitivo.
   8. **Aplicación web:** Programa desarrollado específicamente para ser colgada en la internet, esto por medio de servidores permiten que nuestro sistema se ejecute sin problemas, ahorrándole al usuario la descarga e instalación de software.
   9. **Interfaz:** Parte visual de la aplicación web con la que va a interactuar el usuario con el sistema, ejemplos: Botones, cajas de texto, tablas, labels, etc.
   10. **Reporte:** Presentación de los datos, de todas sus partidas, del usuario en tablas y gráficos según sea el criterio que se tiene que seguir para mostrar las estadísticas en pantalla, ejemplos: partidas ganadas, perdidas, empatadas, etc.
   11. **Archivo XML:** Documento plano que contiene información basada en múltiples filas y etiquetas de cierre y apertura.
2. **Modelo Relacional.**

****

**Script de la base de datos.**

CREATE DATABASE Othello

USE Othello

CREATE TABLE PlayervsPlayer (

idVersus int IDENTITY,

idJugador1 int not null,

idJugador2 int not null,

idPartida int not null,

PRIMARY KEY (idVersus)

);

CREATE TABLE Usuario (

idUsuario int IDENTITY,

nombre varchar(40) not null,

apellido varchar(40) not null,

nombreUsuario varchar(0) not null,

contraseña varchar(35) not null,

fechaNacimiento date,

pais varchar(25) not null,

correoElectronico varchar(40) not null,

PRIMARY KEY (idUsuario)

);

CREATE TABLE Solitario (

idSolitario int IDENTITY,

idUsuario int not null,

idPartida int not null,

PRIMARY KEY (idSolitario)

);

CREATE TABLE Torneo (

idTorneo int IDENTITY,

ganador varchar(40) not null,

fecha date,

hora datetime,

PRIMARY KEY (idTorneo)

);

CREATE TABLE Partida (

idPartida int IDENTITY,

ganador varchar(40) not null,

perdedor varchar(40) not null,

fecha date,

hora datetime,

PRIMARY KEY (idPartida)

);

CREATE TABLE PartidaTorneo (

idPartidaTorneo int IDENTITY,

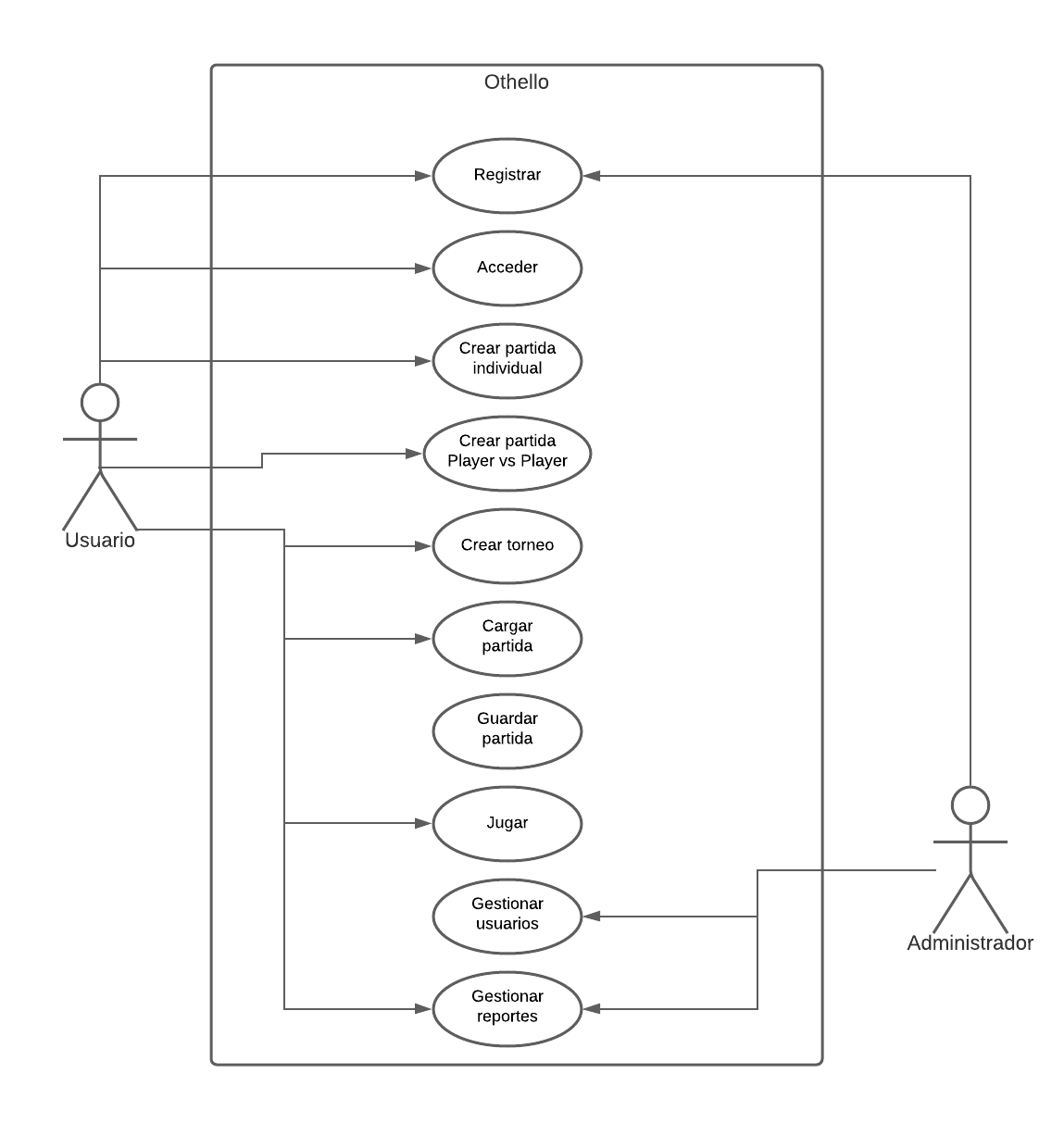
idVersus int not null,

idTorneo int not null,

PRIMARY KEY (idPartidaTorneo)

);

1. **Diagrama de casos de uso.**

****

1. **Casos de uso de alto nivel.**
   1. **Caso de uso:** Registrar

**Actores:** Usuario o administrador

**Tipo:** Primario, real.

**Descripción:** Consiste en la recolección y almacenamiento de la información de un usuario en la base de datos del sistema.

* 1. **Caso de uso:** Acceder

**Actores:** Usuario

**Tipo:** Primario, real.

**Descripción:** Con una cuenta previamente creada y con sus credenciales, el usuario procede a autentificarse y acceder al sistema.

* 1. **Caso de uso:** Crear partida individual.

**Actores:** Usuario

**Tipo:** Primario, real.

**Descripción:** El usuario tendrá la opción de jugar en solitario contra la máquina virtual.

* 1. **Caso de uso:** Crear partida Player vs Player.

**Actores:** Dos Usuarios

**Tipo:** Primario, real.

**Descripción:** El usuario tendrá la opción de jugar en una partida contra otro usuario registrado en el sistema por medio de una invitación o con la partida abierta.

* 1. **Caso de uso:** Crear torneo.

**Actores:** Entre 4 a 16 Usuarios

**Tipo:** Primario, real.

**Descripción:** A modo de involucrar más jugadores, se puede crear torneos o competencias donde participan los usuarios que estén registrados en el sistema.

* 1. **Caso de uso:** Cargar partida

**Actores:** Usuario

**Tipo:** Primario, real.

**Descripción:** El sistema cargará todo el progreso de alguna partida que fue guardada con anterioridad por medio de un archivo XML.

* 1. **Caso de uso:** Guardar partida

**Actores:** Usuario

**Tipo:** Primario, real.

**Descripción:** El sistema guardará todo el progreso de la partida que quiera el usuario si se quiere salir antes de la culminación de la partida, los datos se almacenaran en un archivo XML.

* 1. **Caso de uso:** Jugar

**Actores:** Usuario

**Tipo:** Primario, real

**Descripción:** Luego de haberse creado alguna partida, de unirse a una partida o ingresar a una competencia, la funcionalidad principal se ejecutará con todas las validaciones y reglas de juego Othello.

* 1. **Caso de uso:** Gestionar usuarios

**Actores:** Administrados

**Tipo:** Primario, esencial

**Descripción:** El administrador podrá gestionar la base de datos eliminando, registrando o modificando los usuarios en el sistema.

* 1. **Caso de uso:** Gestionar reportes

**Actores:** Administrador o usuario

**Tipo:** Primario

**Descripción:** el sistema desplegará los datos estadísticos, respecto a las partidas del usuario, mostrándolos en tablas y graficas para una mejor visualización.

1. **Casos de uso expandidos.**
   1. **Caso de uso:** Registrar

**Actores:** Usuario o administrador

**Propósito:** Que los usuarios se registren en el sistema

**Resumen:** el sistema contará con un modulo que contiene campos de texto que el usuario deberá llenar con sus datos para poder tener una cuenta y tener acceso a las funcionalidades de la aplicación web.

**Tipo:** Primario, real.

**Referencias cruzadas:** Funciones R1.2

**Curso normal de eventos.**

1. El usuario o administrador entra a la plataforma y busca el módulo de registro
2. Llena todos los campos de textos con la información que se pide para la creación de la cuenta
3. Envía los datos, el sistema verificará que no falte ningún campo obligatorio
4. La cuenta será creada.

**Cursos alternos.**

1. Línea 1: El usuario o administrador no tienen acceso a internet, necesitaran recargar la página o esperar a que se reestablezca su servicio.
2. Línea 3: No llenó algún campo obligatorio, el sistema le pedirá que verifique sus datos.
   1. **Caso de uso:** Acceder

**Actores:** Usuario

**Propósito:** Que el usuario tenga acceso a las funcionalidades del sistema.

**Resumen:** Con una cuenta previamente creada, el sistema le pedirá que ingrese sus credenciales en los campos correspondientes para la autentificación y que éste tenga acceso al sistema.

**Tipo:** Primario, real.

**Referencias cruzadas:** Funciones R1.3

**Curso normal de eventos.**

1. El usuario o administrador entra a la plataforma y busca el módulo de acceso
2. Llena los campos correspondientes con sus credenciales
3. Envía los datos para que el sistema haga la autentificación
4. Entra al sistema

**Cursos alternos.**

1. Línea 1: El usuario no tienen acceso a internet, necesitara recargar la página o esperar a que se reestablezca su servicio.
2. Línea 3: Dejó algún campo vacío, el sistema le pedirá que verifique sus datos
3. Línea 4: Sus datos no coinciden, acceso denegado y tiene que repetir los pasos para la autentificación.
   1. **Caso de uso:** Crear partida individual.

**Actores:** Usuario

**Propósito:** Que el usuario tenga acceso a una partida y pueda jugar

**Resumen:** El usuario tendrá la opción crear una partida en solitario para jugar contra la máquina virtual.

**Tipo:** Primario, real.

**Referencias cruzadas:** Funciones R1.4

**Curso normal de eventos.**

1. El usuario elige la opción de crear partida
2. Selecciona la opción partida en solitario
3. Se despliega el tablero para jugar contra la maquina

**Cursos alternos.**

1. Línea 1,2 y 3: El usuario no tiene acceso a internet, necesitará recargar la pagina o esperar que se reestablezca su servicio
   1. **Caso de uso:** Crear partida Player vs Player.

**Actores:** Dos Usuarios

**Propósito:** Que el usuario tenga acceso a una partida contra otro jugador

**Resumen:** El usuario tendrá la opción crear una partida, Player vs Player, éste podrá mandar la invitación o dejar la partida abierta para que otro usuario del sistema acceda a la partida.

**Tipo:** Primario, real.

**Referencias cruzadas:** Funciones R1.4

**Curso normal de eventos.**

1. El usuario elige la opción de crear partida
2. Selecciona la opción partida: Player vs Player
3. Se despliega el tablero para jugar contra el jugador que se haya provisto para la partida

**Cursos alternos.**

1. Línea 1,2 y 3: El usuario no tiene acceso a internet, necesitará recargar la página o esperar que se reestablezca su servicio
   1. **Caso de uso:** Crear torneo.

**Actores:** Entre 4 a 16 Usuarios

**Propósito:** Crear una competencia que involucre muchos jugadores agregados al sistema

**Resumen:** Para involucrar a mas jugadores del sistema y tener muchas más partidas, la opción del torneo permite que entre 4 y 16 jugadores del sistema puedan enfrentarse entre si

**Tipo:** Primario, real

**Referencias cruzadas:** Funciones R1.5

**Curso normal de eventos.**

1. El usuario elige la opción de crear torneo
2. El usuario fija el número de participantes que estarán en el torneo
3. Se enviarán invitaciones a los jugadores o bien se deja el torneo abierto
4. Se hace el emparejamiento de los jugadores
5. Se despliega el tablero para jugar contra el jugador que se haya provisto para la partida

**Cursos alternos.**

1. Línea 1,2 y 5: El usuario no tiene acceso a internet, necesitará recargar la página o esperar que se reestablezca su servicio
2. Línea 3 y 4:
   1. **Caso de uso:** Cargar partida

**Actores:** Usuario

**Propósito:** Recuperar el progreso de una partida pasada

**Resumen:** El sistema cargará todo el progreso de alguna partida que fue guardada con anterioridad por medio de un archivo XML. Las fichas se posicionarán en el tablero tal y como estaba cuando se procedió a guardar la partida

**Tipo:** Primario, real.

**Referencias cruzadas:** Funciones R1.6

**Curso normal de eventos.**

1. El usuario elegirá la opción de cargar partida
2. Se desplegará el explorador de archivos para buscar el archivo XML
3. Selecciona el archivo y el sistema procesará la información para enviarlo al tablero

**Cursos alternos.**

1. Línea 2: el explorador de archivos no se despliega, el usuario deberá recargar la página o bien revisar su conexión de internet.
2. Línea 3: La información procesada es errónea, se tiene que revisar el archivo que se cargó o bien recargar la página.
   1. **Caso de uso:** Guardar partida

**Actores:** Usuario

**Propósito:** Guardar el progreso de una partida que por cierta razón no se puede completar en su momento

**Resumen:** El sistema guardará todo el progreso de la partida que quiera el usuario si se quiere salir antes de la culminación de la partida, los datos se almacenaran en un archivo XML. La posición de las fichas, los movimientos realizados se guardarán asegurando la recuperación del 100% de la información

**Tipo:** Primario, real.

**Referencias cruzadas:** Funciones R1.7

**Curso normal de eventos.**

1. El usuario elegirá la opción de guardar partida
2. Se desplegará el explorador de archivos para elegir el lugar donde se guardará el archivo XML y el nombre que se le dará
3. Se creará un archivo XML con toda la información de la partida.

**Cursos alternos.**

1. Línea 2: el explorador de archivos no se despliega, el usuario deberá recargar la página o bien revisar su conexión de internet.
2. Línea 3: El archivo XML no se crea, el usuario deberá recargar la página o bien revisar su conexión a internet
   1. **Caso de uso:** Jugar

**Actores:** Uno o varios usuarios

**Propósito:** Que el usuario pueda disfrutar del juego de mesa de manera On Line

**Resumen:** Luego de haberse creado alguna partida, de unirse a una partida o ingresar a una competencia, la funcionalidad principal se ejecutará con todas las validaciones y reglas de juego Othello. En este apartado el usuario ya podrá disfrutar del objetivo principal del sistema, jugar.

**Tipo:** Primario, real

**Referencias cruzadas:** Funciones R1.8, R1.9

**Curso normal de eventos:**

1. El usuario ingresa a una partida
2. Hace sus movimientos en el tablero de juegos
3. Pierde, gana o empata la partida

**Cursos alternos.**

1. Línea 1 y 2: La página no se despliega, el usuario no tiene acceso a internet o debe recargar la página.
   1. **Caso de uso:** Gestionar usuarios

**Actores:** Administrados

**Propósito:** Llevar el control de los usuarios que se almacenan en la base datos

**Resumen:** El administrador podrá gestionar la base de datos eliminando usuarios, registrando usuarios nuevos o modificar la información de los usuarios existentes en el sistema, esto para tener actualizada la información tanto de los usuarios como de la base de datos.

**Tipo:** Primario, real

**Referencias cruzadas:** Funciones R1.10

**Curso normal de eventos.**

1. El administrador elegirá una opción entre: Registrar, modificar o eliminar
2. Según su elección en la línea 1, procederá a buscar al usuario para aplicarle los cambios correspondientes
3. Se actualizará la base de datos

**Cursos alternos.**

1. Línea 1 y 2: no se le muestra en pantalla las opciones que debería visualizar, deberá recargar la pagina o revisar su conexión a internet
2. La base de datos no se actualiza, deberá revisar la conexión entre el sistema
   1. **Caso de uso:** Gestionar reportes

**Actores:** Administrador o Usuario

**Propósito:** Tener una mejor visualización de las estadísticas del jugador

**Resumen:** el sistema desplegará los datos estadísticos de un usuario en específico, respecto a las partidas del usuario (Ganadas, perdidas o empatadas), mostrándolos en tablas y graficas para una mejor visualización y apreciación.

**Tipo:** Primario, esencial

**Referencias cruzadas:** Funciones R1.11

**Curso normal de eventos.**

1. Se elegirá el tipo de reporte que se quiere visualizar
2. En pantalla se desplegará toda la información que se solicita

**Cursos alternos**.

1. Línea 1 y 2: No se muestra nada en pantalla, se debe recargar la pagina o bien revisar su conexión a internet.