



Análisis y diseño de aplicaciones - TFU 3

Profesores: José Abadie, Sebastián Feirres

Equipo 4

PARTE 1

Requisitos:

Automatización: El sistema debe ser automatizado para gestionar los puntajes de todas las disciplinas olímpicas de manera eficiente y precisa. Esto para intentar eliminar la gestión manual lo más posible.

- Criterio de ajuste: la única parte manual es ingresar los puntajes a mano.
- Razón fundamental: se busca optimizar el proceso de ingreso de puntajes para que sea más ágil y tenga menos errores.

Gestión de múltiples disciplinas: El sistema debe ser capaz de manejar una amplia variedad de disciplinas olímpicas, cada una con sus propias reglas y métodos de puntuación.

- Criterio de ajuste: que cada disciplina respete el método de puntuación que le corresponde, además de que se deben poder agregar todas las disciplinas necesarias.
- Razón fundamental: al tratarse de un sistema para los Juegos Olímpicos, contará con muchas disciplinas y es importante poder llevar el registro de cada una.

Gestión de múltiples competidores: El sistema debe albergar varios competidores, cada uno debe tener sus atributos en caso de ser necesarios para el cálculo de puntos de su disciplina.

- Criterio de ajuste: que cada competidor tenga su información y que esa información sea relevante para el cálculo de su puntaje en la disciplina.
- Razón fundamental: dado que cada disciplina tiene sistemas diferentes para evaluar la competencia, es necesario poder mantener la información relevante de cada participante.

Ingreso de puntajes: El sistema debe permitir el ingreso de puntajes para cada competidor y debe adaptarse a cada disciplina de acuerdo a sus sistemas de puntuación.

- Criterio de ajuste: al tratar de ingresar un puntaje, el sistema solo debe aceptar dígitos.
- Razón fundamental: como son puntajes sobre los que se realizan cálculos, si se ingresa un caracter no numérico habrá un error.

Obtención de puntajes de un competidor: El sistema debe obtener el puntaje calculado para un competidor en particular.

- Criterio de ajuste: al tratar de obtener los puntos de un competidor, el sistema devuelve el cálculo correspondiente a su disciplina a su puntuación final.

- Razón fundamental: al tratarse de una competencia, es necesario conocer el puntaje final de cada participante

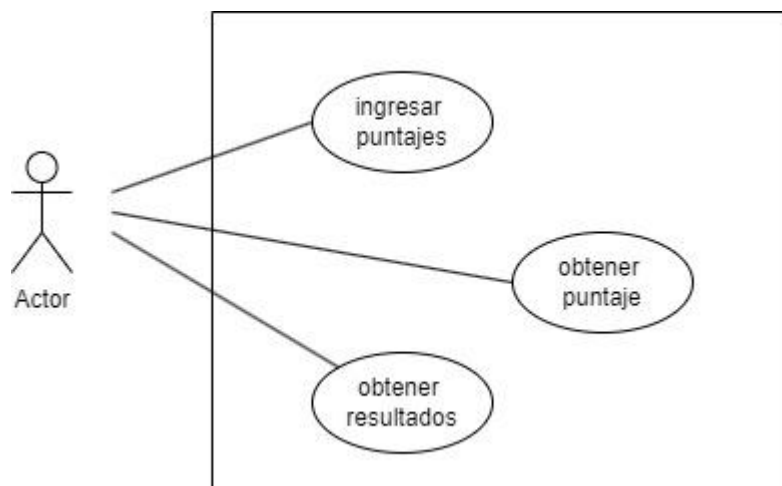
Obtención de resultados: El sistema debe permitir obtener los resultados de los puntajes de una disciplina, ordenados según el mayor puntaje de acuerdo a la forma de medición de esa disciplina.

- Criterio de ajuste: al tratar de obtener los resultados de un deporte, debe devolver una lista ordenada por puntuación de mayor a menor de los puntajes de los competidores.
- Razón fundamental: dentro de cada disciplina es necesario saber las posiciones de los competidores para obtener los resultados de la competencia.

PARTE 2

En cuanto al funcionamiento de la aplicación, la interacción con actores es ingresar puntajes y obtenerlos. Al momento de ingresar puntajes, el actor elige un deporte, selecciona su categoría (si tiene) y el sistema le brinda formularios para rellenar. Estos formularios se adaptan según los sistemas de puntuación de cada disciplina. Por otra parte, al obtener puntajes, el actor elige qué disciplina y categoría (si tiene), y el sistema devuelve los puntajes de los competidores de forma ordenada de mayor a menor acorde a los sistemas de puntuación de esa disciplina.

Para modelar estos casos de uso, creamos un diagrama de casos de uso, en donde se ilustra un juez que puede acceder a estas dos funcionalidades del sistema.



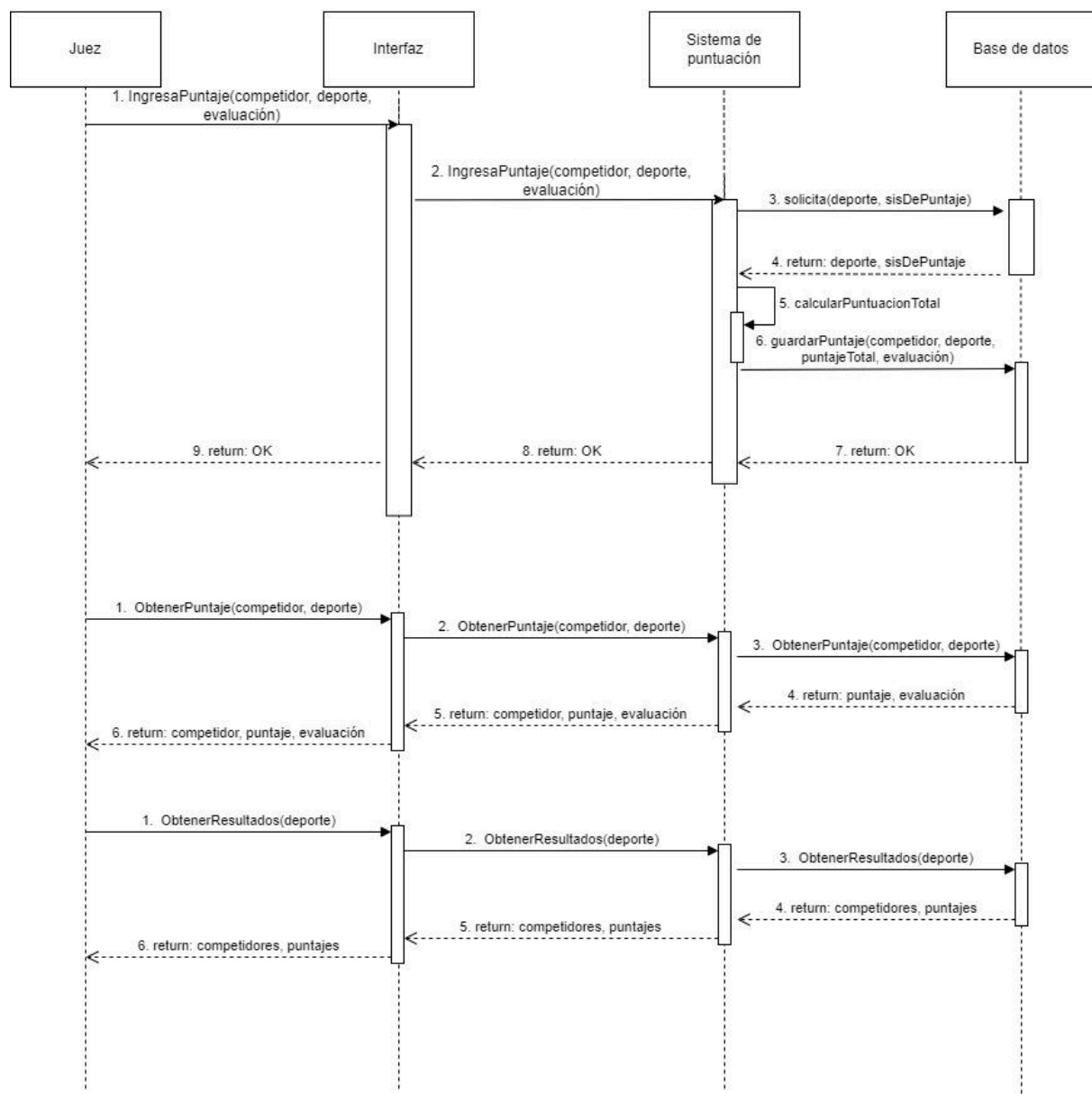
Para entrar en detalle con los procesos, creamos un diagrama de secuencia que permite visualizar el orden de las interacciones. En el caso de ingreso de puntaje el juez interactúa a través de una interfaz con el sistema, en donde especifica el deporte, competidor y puntaje. El sistema de puntuación calcula el

puntaje acorde a la disciplina y se guardan los datos en la base de datos. Luego, se le informa al usuario que los datos fueron ingresados con éxito.

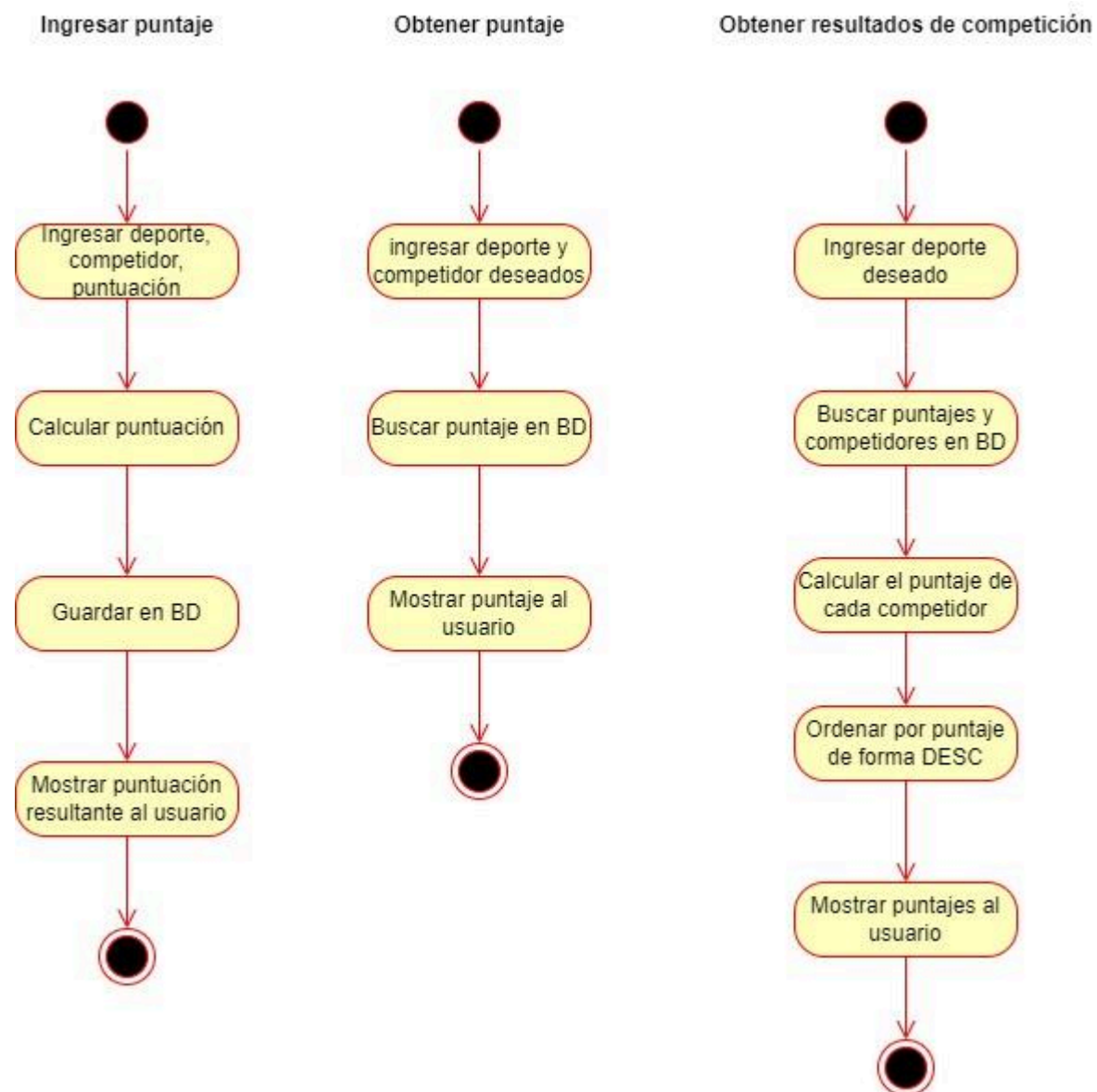
Para la funcionalidad de obtener el puntaje de un competidor, el juez ingresa el deporte y el competidor, porque tomamos como supuesto que un competidor puede estar en varios deportes, se busca en la base de datos y se le devuelven los datos al juez.

Para obtener los resultados de un deporte, se indica el deporte, la base de datos devuelve los puntajes de ese deporte, el sistema de puntuación ordena de mayor a menor los puntajes, y le devuelve al juez la lista ordenada.

Diagrama de secuencia - manejo de puntuaciones



También creamos un diagrama de actividad, para observar el flujo de trabajo del sistema. Las funcionalidades son lineales para el sistema, no existen condicionales ni bifurcaciones.



Finalmente, creamos un diagrama de clases que modela el sistema más concretamente. Primero definimos una clase que sería la encargada de interactuar con el usuario, donde el usuario usa las funcionalidades ingresando datos si es necesario, y donde puede observar los resultados de las mismas. Esta clase interactúa con otra llamada "Deporte", que modela una disciplina de los Juegos. Contiene como atributos una lista de competidores y jueces (la clase "Juez" contiene los datos del juez, no tiene métodos), la modalidad y categoría. Como métodos, puede calcular los resultados de la competencia, así como también obtener el puntaje de un competidor en particular. Estos métodos van a ser implementados de distintas formas acorde al deporte.

En cuanto a los competidores, la clase “Competidor” tiene los datos del mismo, y tiene una lista de actuaciones. Las actuaciones representan el sistema de puntaje en una disciplina, y en el caso de que la misma persona compita en varios deportes, se tiene la información de los puntajes dentro del mismo competidor. La clase “Actuación” es abstracta, que será concretada de acuerdo a los sistemas de puntaje de cada deporte, con sus propios atributos.

