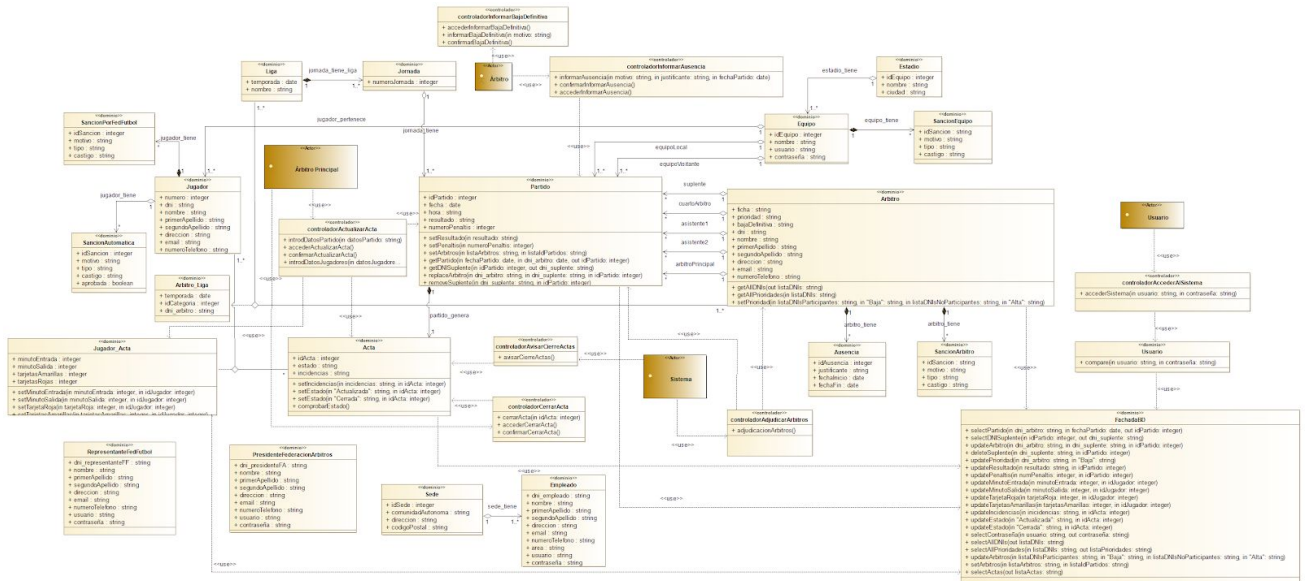


7.1. Diagrama de clases de diseño



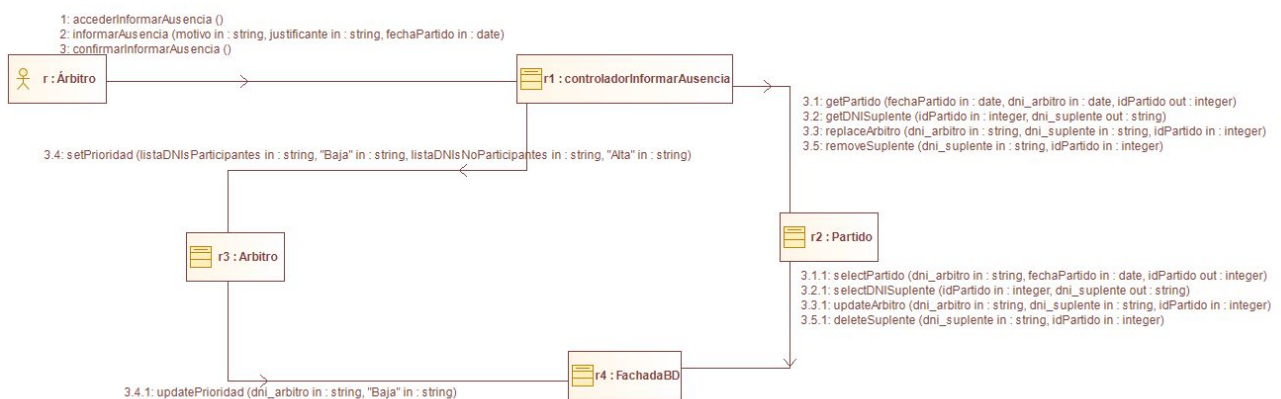
7.2. Modelo de objetos de software y sus colaboraciones

7.2.1. CU4-Informar de una ausencia

Diagrama de secuencia del sistema



Diagrama de colaboración de diseño



- **getPartido (fechaPartido:date, dni_arbitro)**
 - Función llamada desde *controladorInformarAusencia*.

- Esta función se comunica con la clase *Partido* para obtener información del partido correspondiente del cual un árbitro está informando su ausencia.
 - Recupera el identificador de la clase *Partido*.
- ***getDNISuplente (idPartido: int)***
 - Función llamada desde *controladorInformarAusencia*.
 - Esta función se comunica con la clase *Partido* para obtener información del árbitro suplente del partido.
 - Recupera el DNI del árbitro suplente y con partido asignado con identificador igual a *idPartido* pasado como parámetro.
- ***selectPartido (dni_arbitro:string, fechaPartido:date)***
 - Función llamada desde la clase *Partido*.
 - Esta función se comunica con la *FachadaBD* para obtener información del partido del que se va a ausentar el árbitro que está informando de su ausencia con una fecha pasada por parámetros. Además, como puede haber más de un partido en un mismo día, selecciona el partido donde el árbitro esté asignado.
 - Recupera el *idPartido* del partido del cual el árbitro se va a ausentar y lo devuelve al *controladorInformarAusencia*.
- ***selectDNISuplente (idPartido:int)***
 - Función llamada desde la clase *Partido*.
 - Se comunica con la clase *FachadaBD* para obtener el DNI del árbitro suplente del partido con identificador pasado como parámetro.
 - Esta función recupera el identificador del árbitro suplente del partido (*dni_suplente*) y lo devuelve al *controladorInformarAusencia*.
- ***updateArbitro (dni_arbitro:string, dni_suplente:string, idPartido:int)***
 - Función llamada desde la clase *Partido*.
 - Se comunica con la clase *FachadaBD* para reemplazar el árbitro ausente con *dni_arbitro* pasado por parámetros por el árbitro suplente con *dni_suplente* recibido como parámetro.
 - Esta función por tanto, modifica los datos del partido con *idPartido* pasado por parámetros cambiando al árbitro ausente por el árbitro suplente.
- ***deleteSuplente (dni_suplente:string, idPartido:int)***
 - Función llamada desde la clase *Partido*.
 - Se comunica con la clase *FachadaBD*. La función recibe como parámetros el DNI del árbitro suplente y el identificador del partido.

- Esta función elimina al árbitro suplente con dni igual a *dni_suplente*, del partido con identificador igual a *idPartido*.

7.2.2. CU5-Actualizar acta

Diagrama de secuencia del sistema

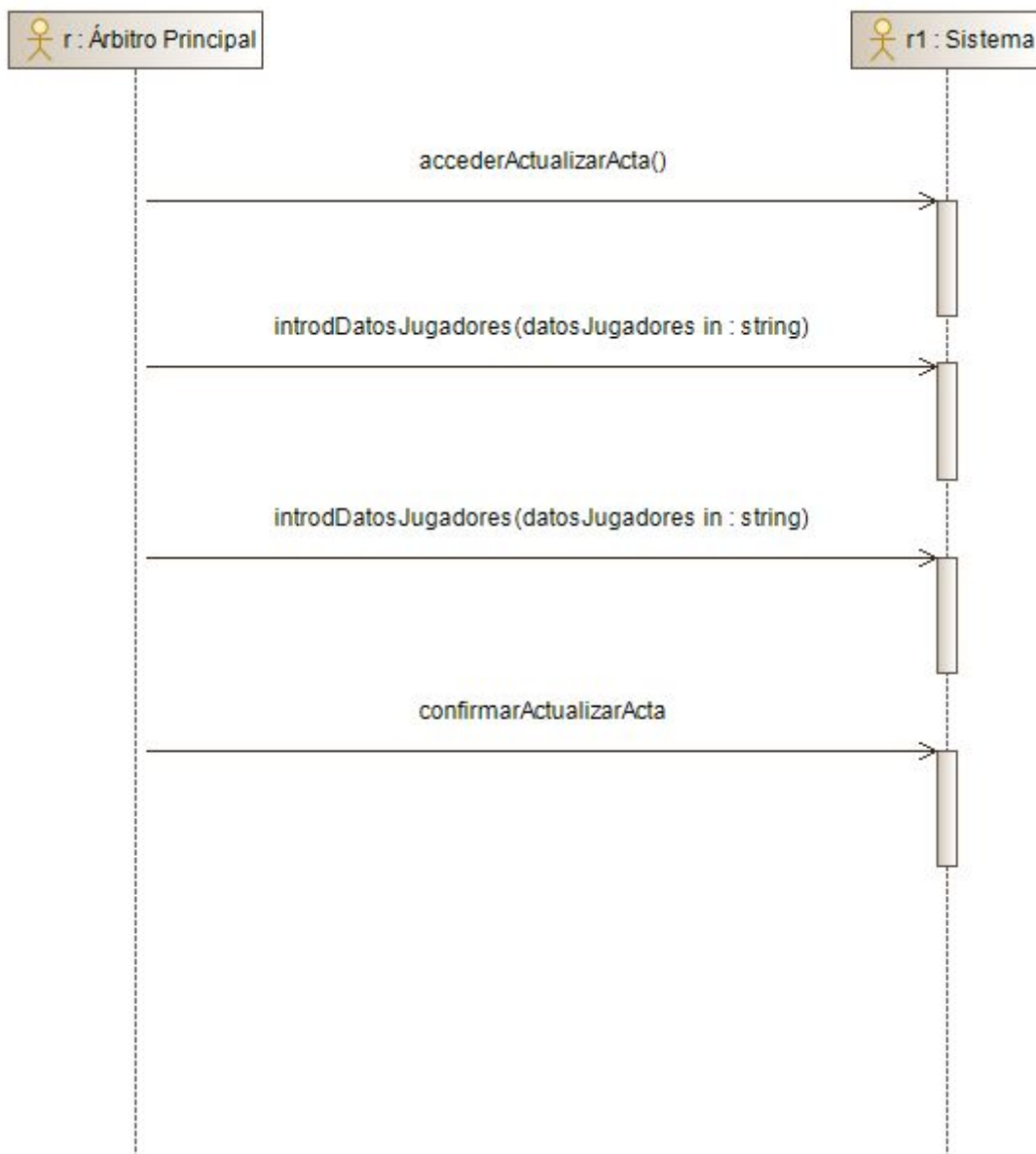
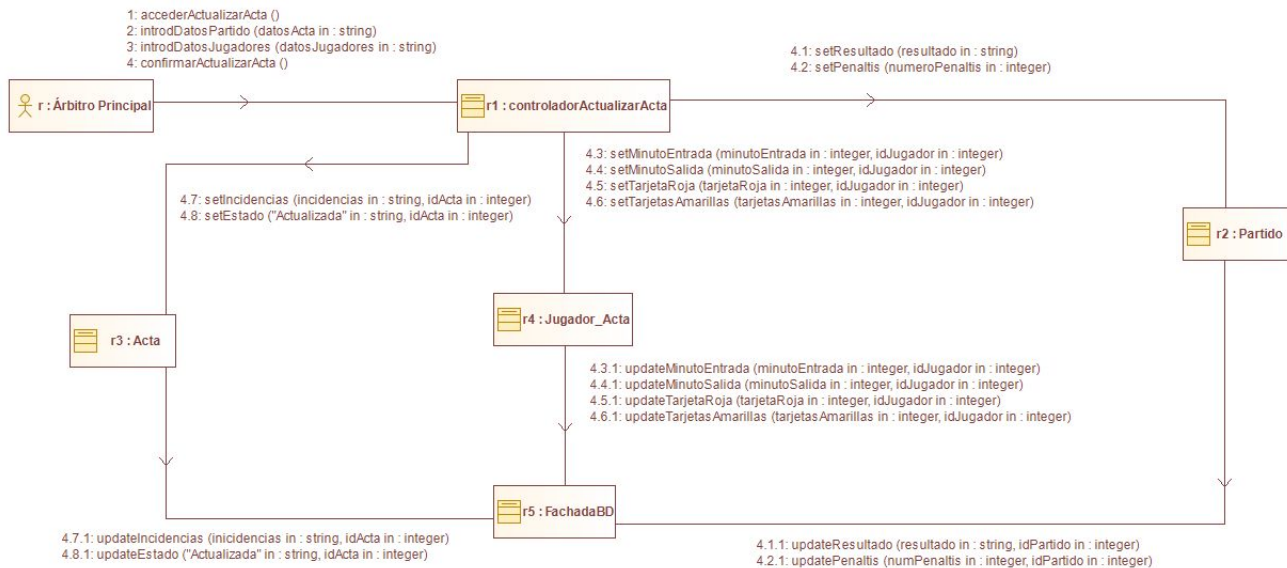


Diagrama de colaboración de diseño



- **setMinutoEntrada (minutoEntrada:int, idJugador:int)**

- Función llamada desde el *controladorActualizarActa* hacia *jugador_acta*.
- Se encarga de almacenar el minuto de entrada del jugador identificado por *idJugador* pasado por parámetro.
- Esta función se realiza tantas veces como jugadores hayan entrado a jugar a un partido con un determinado minuto de entrada.

- **setMinutoSalida (minutoSalida:int, idJugador:int)**

- Función llamada desde el *controladorActualizarActa* hacia *jugador_acta*.
- Se encarga de almacenar el minuto de salida del jugador identificado por *idJugador* pasado por parámetro.
- Esta función se realiza tantas veces como jugadores hayan salido de un partido con un determinado minuto de salida.

- **setTarjetaRoja (tarjetaRoja:int, idJugador:int)**

- Función llamada desde el *controladorActualizarActa* hacia *jugador_acta*.
- Se encarga de modificar el número de tarjetas rojas (0 ó 1) que dispone un jugador con identificador *idJugador* pasado por parámetro.
- Esta función se realiza tantas veces como tarjetas rojas se pongan a los jugadores en un partido.

- **setTarjetasAmarillas (tarjetaAmarilla:int, idJugador:int)**

- Función llamada desde el *controladorActualizarActa* hacia *jugador_acta*.

- Se encarga de modificar el número de tarjetas amarillas que dispone un jugador con identificador *idJugador* pasado por parámetro.
- Esta función se realiza tantas veces como tarjetas amarillas se pongan a los jugadores en un partido.

7.2.3. CU6-Cerrar acta

Diagrama de secuencia del sistema

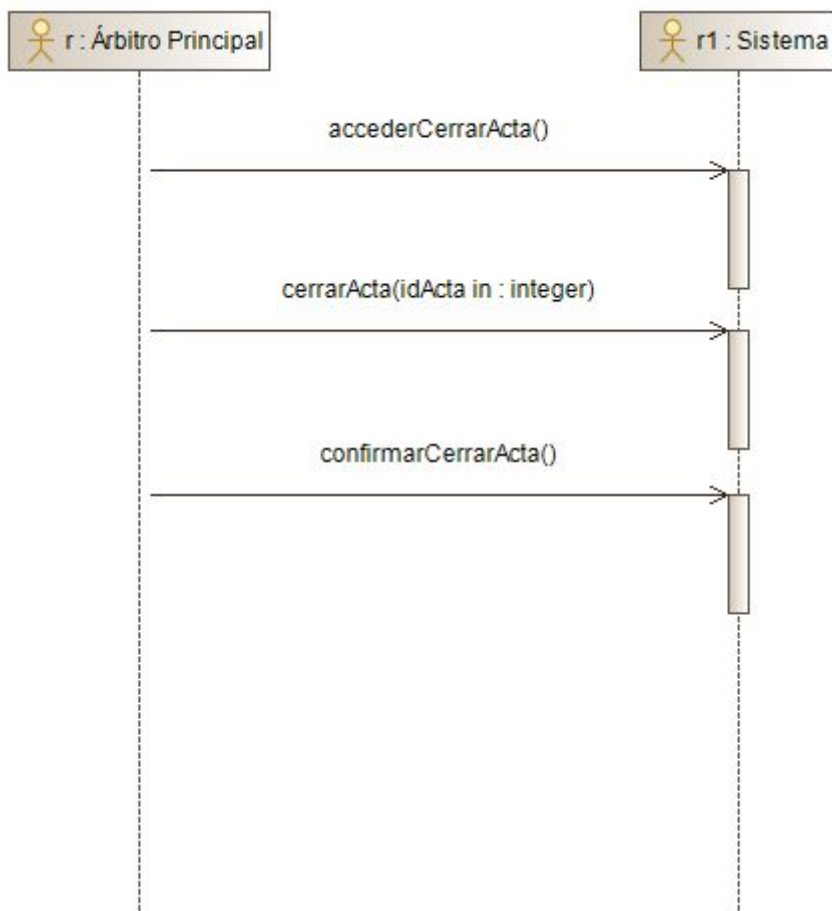
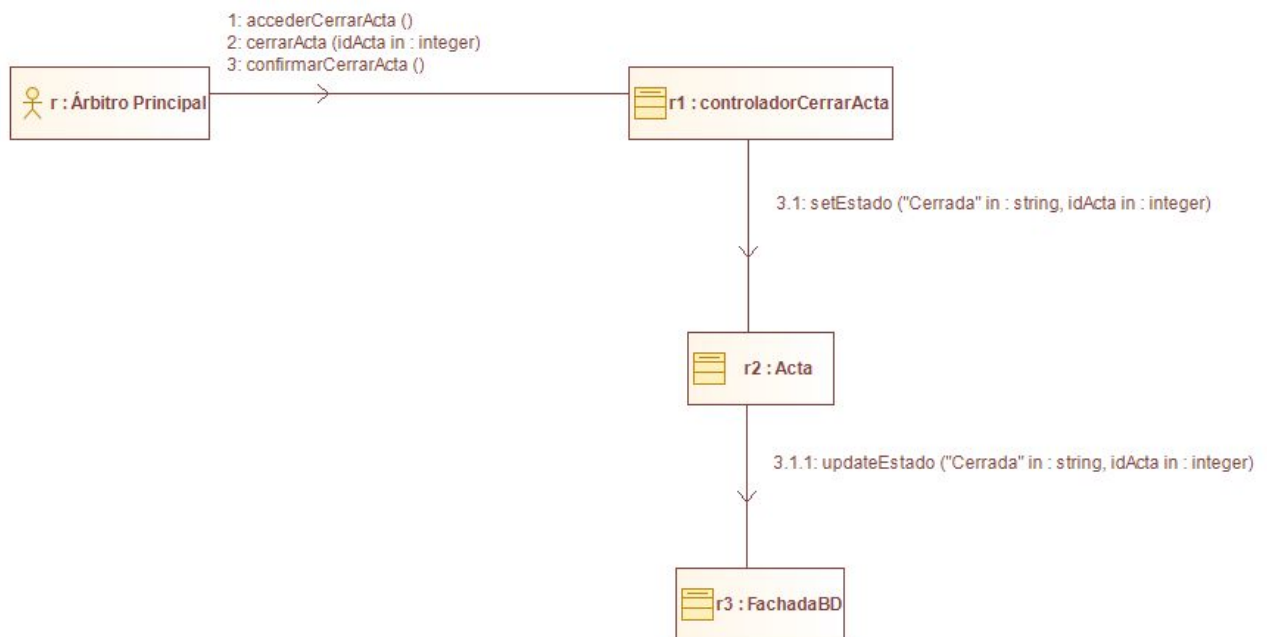


Diagrama de colaboración de diseño



7.2.4. CU1-Acceder al sistema

Diagrama de secuencia del sistema

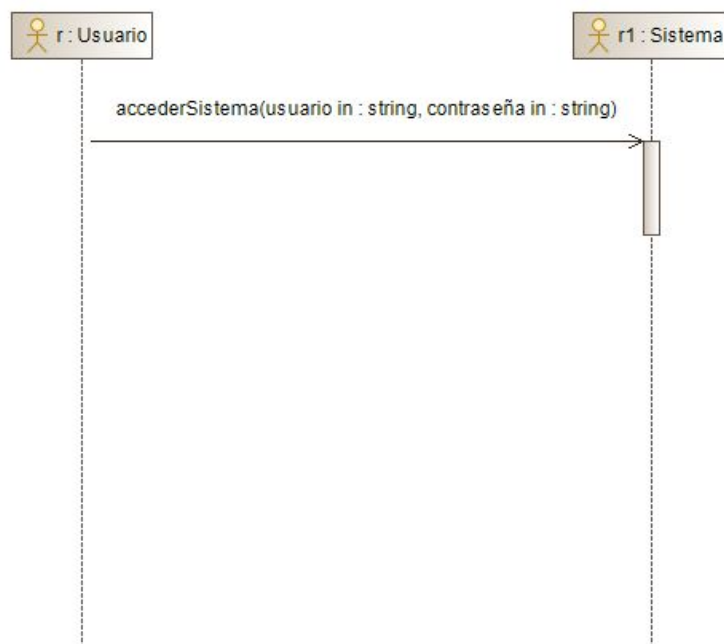
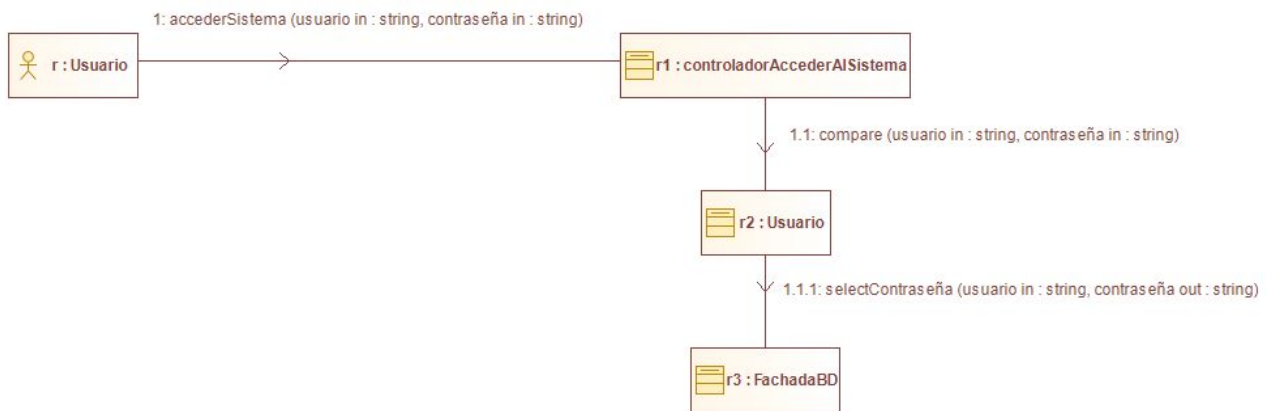


Diagrama de colaboración de diseño



- **compare (usuario:string, contraseña:string)**
 - Función llamada desde el *controladorAccederSistema*.
 - Esta función se comunica con la clase *Usuario*. La función recibe como parámetros el usuario y la contraseña.
 - Esta función compara la contraseña recibida por parámetro con la contraseña que devuelve el select correspondiente del usuario recibida por parámetro, la cual está almacenada en la base de datos.

7.2.5. CU11-Adjudicar árbitros a partidos

Diagrama de secuencia del sistema

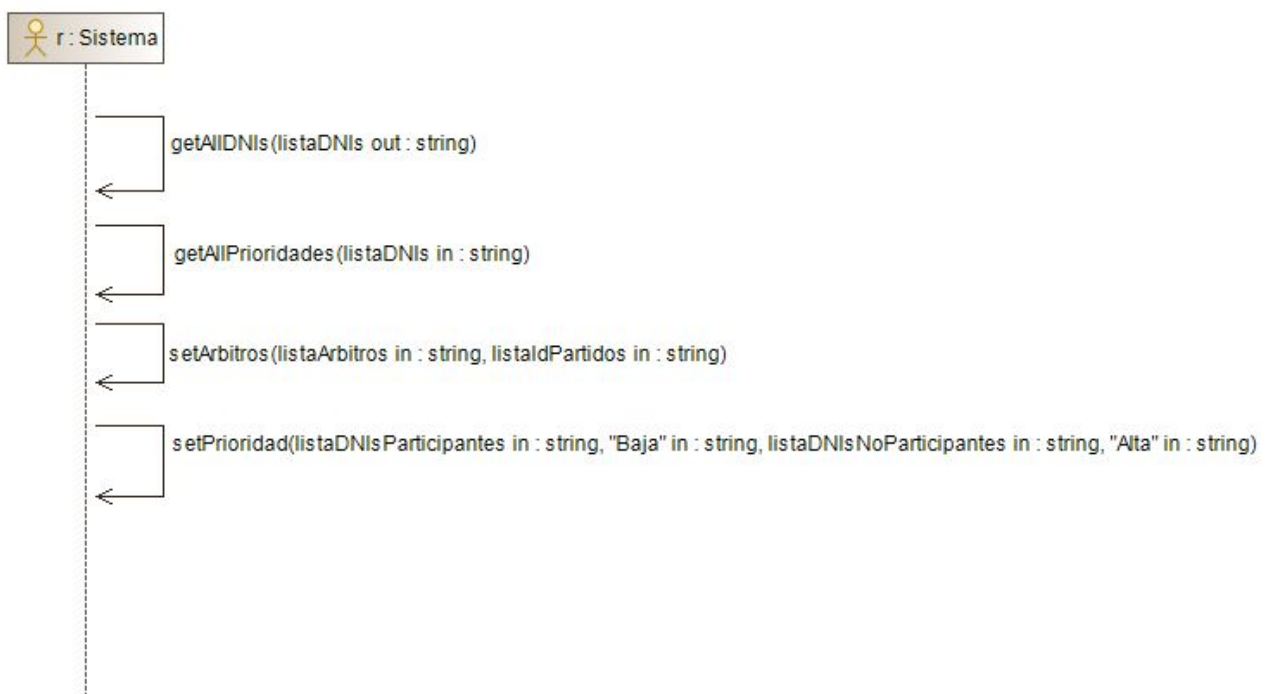
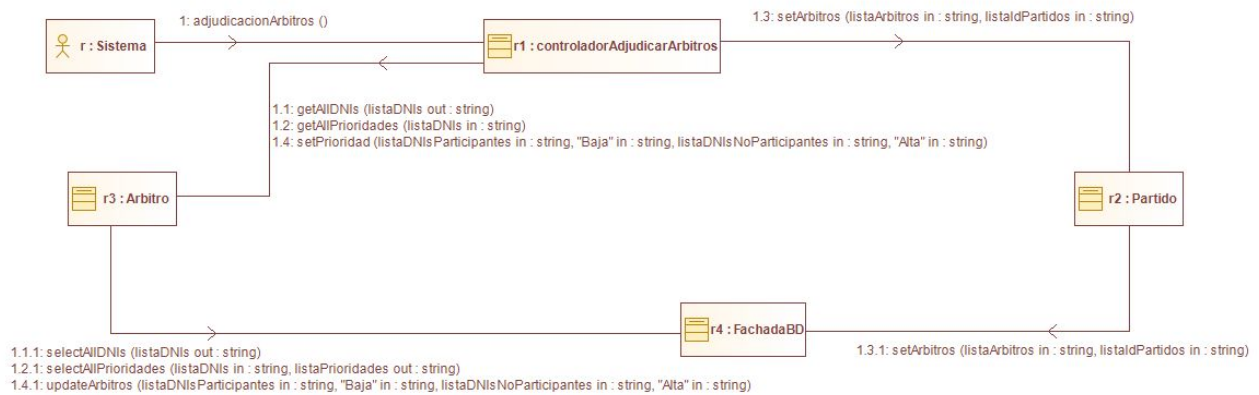


Diagrama de colaboración de diseño



- **getAllDNIs()**
 - Función llamada desde el *controladorAdjudicarArbitros*.
 - Se comunica con la clase *Árbitro*.
 - Esta función devuelve una lista con todos los DNIs de los árbitros.
- **getAllPrioridades (listaDNIs:list)**
 - Función llamada desde la clase *controladorAdjudicarArbitros*.
 - Esta función se comunica con la clase *Árbitro*. La función recibe como parámetros una lista de DNIs de los árbitros.
 - Devuelve una lista con la prioridad de todos los árbitros.
- **selectDNIs()**
 - Función llamada desde la clase *Árbitro*.
 - Se comunica con *FachadaBD*.
 - Esta función devuelve una lista con todos los DNIs de los árbitros existentes en la base de datos.
- **selectAllPrioridades (listaDNIs:list)**
 - Función llamada desde la clase *Árbitro*.
 - Esta función se comunica con la clase *FachadaBD*. La función recibe como parámetros una lista de DNIs de los árbitros.
- **setArbitros (listaDNIs:list,listaIdPartidos:list)**
 - Función llamada desde el *controladorAdjudicarArbitros*.
 - Esta función se comunica con la clase *Partido*. La función recibe como parámetros una lista con los DNIs de todos los árbitros que han sido seleccionados y otra lista con los ids de todos los partidos de la jornada. El resultado de la función es que se guardan en la base de datos los DNIs de cinco árbitros por cada partido de la jornada.

7.2.6. CU2-Informar de baja definitiva

Diagrama de secuencia del sistema

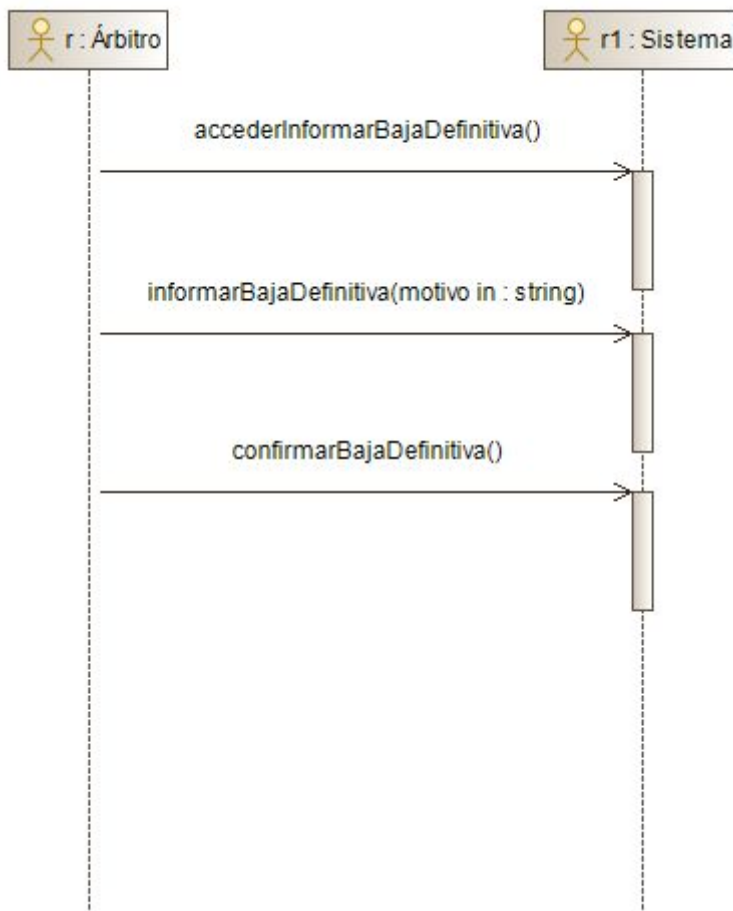
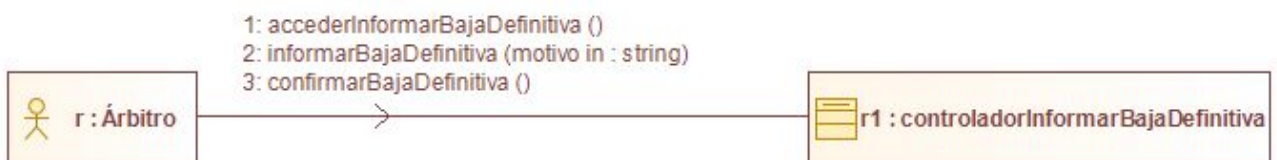


Diagrama de colaboración de diseño



7.2.7. CU10-Notificar al Presidente de la Federación para que acepte/deniegue la sanción

Diagrama de secuencia del sistema



Diagrama de colaboración de diseño

Este caso de uso envía un mensaje vía el sistema para avisar al Presidente de la Federación de las sanciones que tiene que aprobar/denegar por lo que en este caso de uso al no crear las sanciones, no realiza ningún cambio en la base de datos y no tiene ningún diagrama de colaboración.

7.2.8. CU15-Avisa del inminente cierre de las actas

Diagrama de secuencia del sistema

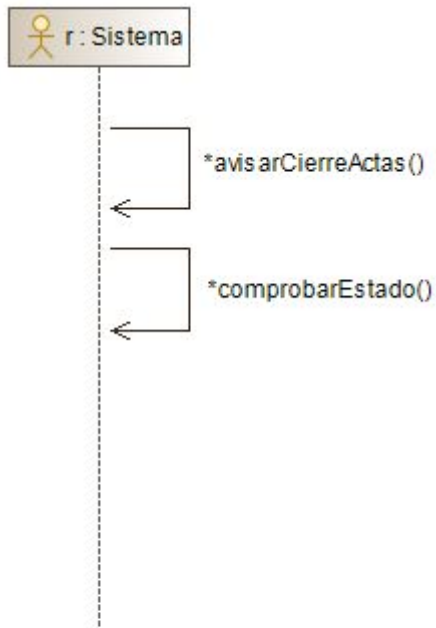
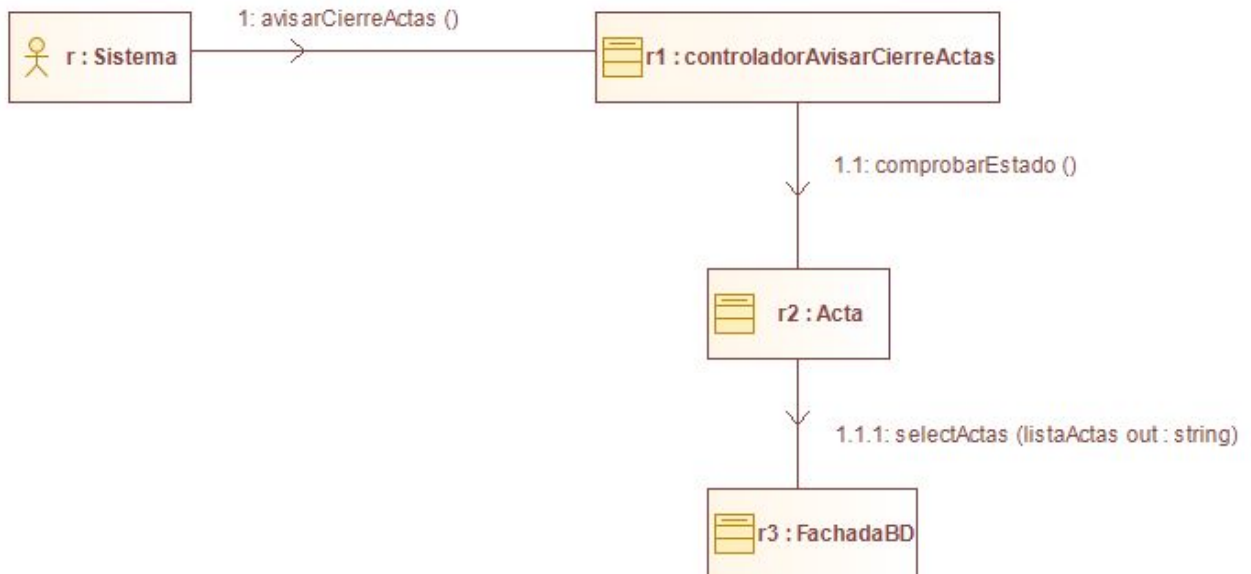


Diagrama de colaboración de diseño



- ***comprobarEstado()***

- Función llamada desde la clase *controladorAvisarCierreActas*.
- Se comunica con la clase *Acta*.
- Esta función comprueba el estado de un acta y devuelve una lista con las actas que están a una hora de cerrarse.

- ***selectActas ()***

- Función llamada desde la clase *Acta*.
- Esta función se comunica con la *FachadaBD* y selecciona todas las actas de una jornada.
- Devuelve a la clase *Acta* todas las actas de una jornada.