

# **Diseño de Pruebas de Unidad**

Daniel Ferreiro Presedo  
Alejandro Sánchez Sánchez

22 de diciembre de 2016

# Índice

<b>1. Observaciones</b>	<b>6</b>
<b>2. Diagramas</b>	<b>9</b>
2.1. GenericDao . . . . .	9
<b>3. ApuestaDao</b>	<b>9</b>
3.1. PR-UN-AD-01 . . . . .	9
3.2. PR-UN-AD-02 . . . . .	10
3.3. PR-UN-AD-03 . . . . .	10
3.4. PR-UN-AD-04 . . . . .	10
3.5. PR-UN-AD-05 . . . . .	11
3.6. PR-UN-AD-06 . . . . .	11
3.7. PR-UN-AD-07 . . . . .	12
3.8. PR-UN-AD-08 . . . . .	12
3.9. PR-UN-AD-09 . . . . .	12
3.10. PR-UN-AD-10 . . . . .	13
3.11. PR-UN-AD-11 . . . . .	13
3.12. PR-UN-AD-12 . . . . .	14
<b>4. OpcionDao</b>	<b>14</b>
4.1. PR-UN-OD-01 . . . . .	14
4.2. PR-UN-OD-02 . . . . .	14
4.3. PR-UN-OD-03 . . . . .	15
4.4. PR-UN-OD-04 . . . . .	15
4.5. PR-UN-OD-05 . . . . .	16
4.6. PR-UN-OD-06 . . . . .	16
4.7. PR-UN-OD-07 . . . . .	16
4.8. PR-UN-OD-08 . . . . .	17
4.9. PR-UN-OD-09 . . . . .	17
4.10. PR-UN-OD-10 . . . . .	17
<b>5. TipoApuestaDao</b>	<b>18</b>
5.1. PR-UN-TAD-01 . . . . .	18
5.2. PR-UN-TAD-02 . . . . .	18
5.3. PR-UN-TAD-03 . . . . .	18
5.4. PR-UN-TAD-04 . . . . .	19

5.5.	PR-UN-TAD-05 . . . . .	19
5.6.	PR-UN-TAD-06 . . . . .	19
5.7.	PR-UN-TAD-07 . . . . .	20
5.8.	PR-UN-TAD-08 . . . . .	20
5.9.	PR-UN-TAD-09 . . . . .	21
5.10.	PR-UN-TAD-10 . . . . .	21
<b>6.</b>	<b>CategoriaDao</b>	<b>21</b>
6.1.	PR-UN-CD-01 . . . . .	21
6.2.	PR-UN-CD-12 . . . . .	22
6.3.	PR-UN-CD-03 . . . . .	22
6.4.	PR-UN-CD-04 . . . . .	23
6.5.	PR-UN-CD-05 . . . . .	23
6.6.	PR-UN-CD-06 . . . . .	23
6.7.	PR-UN-CD-07 . . . . .	24
6.8.	PR-UN-CD-08 . . . . .	24
6.9.	PR-UN-CD-09 . . . . .	24
6.10.	PR-UN-CD-10 . . . . .	25
6.11.	PR-UN-CD-11 . . . . .	25
<b>7.</b>	<b>EventoDao</b>	<b>25</b>
7.1.	PR-UN-ED-01 . . . . .	25
7.2.	PR-UN-ED-02 . . . . .	26
7.3.	PR-UN-ED-03 . . . . .	26
7.4.	PR-UN-ED-04 . . . . .	27
7.5.	PR-UN-ED-05 . . . . .	27
7.6.	PR-UN-ED-06 . . . . .	28
7.7.	PR-UN-ED-07 . . . . .	28
7.8.	PR-UN-ED-08 . . . . .	28
7.9.	PR-UN-ED-09 . . . . .	29
7.10.	PR-UN-ED-10 . . . . .	29
7.11.	PR-UN-ED-11 . . . . .	29
7.12.	PR-UN-ED-12 . . . . .	30
7.13.	PR-UN-ED-13 . . . . .	30
7.14.	PR-UN-ED-14 . . . . .	30
7.15.	PR-UN-ED-15 . . . . .	31
7.16.	PR-UN-ED-16 . . . . .	31
7.17.	PR-UN-ED-17 . . . . .	31

7.18. PR-UN-ED-18 . . . . .	32
7.19. PR-UN-ED-19 . . . . .	32
7.20. PR-UN-ED-20 . . . . .	33
7.21. PR-UN-ED-21 . . . . .	33
7.22. PR-UN-ED-22 . . . . .	33
7.23. PR-UN-ED-23 . . . . .	34
7.24. PR-UN-ED-24 . . . . .	34
7.25. PR-UN-ED-25 . . . . .	35
7.26. PR-UN-ED-26 . . . . .	35
7.27. PR-UN-ED-27 . . . . .	35
7.28. PR-UN-ED-28 . . . . .	36
<b>8. AdminService</b>	<b>36</b>
8.1. PR-UN-AS-01 . . . . .	36
8.2. PR-UN-AS-02 . . . . .	37
8.3. PR-UN-AS-03 . . . . .	37
8.4. PR-UN-AS-04 . . . . .	37
8.5. PR-UN-AS-05 . . . . .	38
8.6. PR-UN-AS-06 . . . . .	38
8.7. PR-UN-AS-07 . . . . .	39
8.8. PR-UN-AS-08 . . . . .	39
8.9. PR-UN-AS-09 . . . . .	39
8.10. PR-UN-AS-10 . . . . .	40
8.11. PR-UN-AS-11 . . . . .	40
8.12. PR-UN-AS-12 . . . . .	41
8.13. PR-UN-AS-13 . . . . .	41
8.14. PR-UN-AS-14 . . . . .	41
8.15. PR-UN-AS-15 . . . . .	42
8.16. PR-UN-AS-16 . . . . .	42
8.17. PR-UN-AS-17 . . . . .	42
8.18. PR-UN-AS-18 . . . . .	43
8.19. PR-UN-AS-19 . . . . .	43
8.20. PR-UN-AS-20 . . . . .	44
8.21. PR-UN-AS-21 . . . . .	44
8.22. PR-UN-AS-22 . . . . .	44
8.23. PR-UN-AS-23 . . . . .	44

<b>9. BetService</b>	<b>45</b>
9.1. PR-UN-BS-01 . . . . .	45
9.2. PR-UN-BS-02 . . . . .	45
9.3. PR-UN-BS-03 . . . . .	46
9.4. PR-UN-BS-04 . . . . .	46
9.5. PR-UN-BS-05 . . . . .	46
9.6. PR-UN-BS-06 . . . . .	47
9.7. PR-UN-BS-07 . . . . .	47
9.8. PR-UN-BS-08 . . . . .	47
9.9. PR-UN-BS-09 . . . . .	47
9.10. PR-UN-BS-10 . . . . .	48
9.11. PR-UN-BS-10 . . . . .	48
9.12. PR-UN-BS-11 . . . . .	48
9.13. PR-UN-BS-12 . . . . .	48
9.14. PR-UN-BS-13 . . . . .	49
9.15. PR-UN-BS-14 . . . . .	49
9.16. PR-UN-BS-15 . . . . .	50

## 1. Observaciones

Debido a la falta de tiempo evitamos preparar la siguientes pruebas dando mayor prioridad a las otras, puesto que las consideramos más importantes al ser más críticas o al ser más usadas:

- Las relacionadas con la clase heredada *GenericDao*, suponiendo que el resultado no se verá modificado respecto al obtenido en la sección *ApuestaDao*.
- En la sección *OpcionDao*, el método *showWinners* cuando todas las Opciones son ganadoras porque suponemos que ya las filtra bien en las otras pruebas del método.
- En la sección *TipoApuestaDao*, el método *findDuplicateBetTypes* para cuando no hay ningún TipoApuesta duplicado, puesto que es una simple comprobación.
- En la sección *CategoriaDao*, el método *findCategories* cuando no hay ninguna, puesto que se supone que devolverá una lista vacía.
- En la sección *EventoDao* :

El método *findEventNoDate*, puesto que realiza lo mismo que el método *findEvent*, por lo que se podría refactorizar y solo probar un método.

El método *findDuplicateEvents* cuando no hay ningún Evento con el mismo nombre programado para la misma fecha de la misma Categoría, puesto que es una simple comprobación.

El método *getNumberOfEvents* y *getNumberOfEvents2*, puesto que se podrían refactorizar en el método *findEvent*.

El método *findAllEvents* y *findAllEventsNoDate*, puesto que se podrían refactorizar del método *findEvent*, ya que realizan lo mismo pero sin ningún filtro.

- En la sección *AdminService* :

El método *createEvent* cuando se inserta con una fecha anterior a la actual, puesto que es una simple comprobación.

El método *createEvent* cuando se intenta insertar un evento con una categoría inexistente, puesto que se supone que se soltará correctamente la excepción al ser una simple comprobación.

El método *findEvent* cuando hay menos eventos de los que se piden y cuando no hay palabras clave, puesto que ya se han realizado estas pruebas con el *findEventNoDate* del **EventoDao**, que es el método al que llama.

El método *findCategoriaById*, puesto que es una simple llamada al método *find* del **CategoriaDao**, el cual ya se ha probado anteriormente.

El método *markAsWinner* cuando se quiere establecer varias Opciones ganadoras cuando el TipoApuesta solo permite una, cuando el Evento ya ha expirado y cuando alguna de las Opciones no pertenece a ese TipoApuesta, puesto que son simples comprobaciones.

El método *addBettingType* cuando ya existe en base de datos el TipoApuesta puesto que es una llamada al método *findDuplicateBetType* del **TipoApuestaDao**, el cual ya está probado; así como cuando no existe el Evento al que asociar o cuando el Evento ya ha expirado, ya que son comprobaciones simples.

El método *getNumberOfEvents*, puesto que es una llamada al método *getNumberOfEvents2* del **EventoDao**, el cual ya está probado con anterioridad.

El método *findTipoApuesta*, puesto que es una simple llamada al método *find* del **TipoApuestaDao**, el cual ya está probado con anterioridad.

El método *findOptionById*, puesto que es una simple llamada al método *find* del **OpcionDao**, el cual ya está probado con anterioridad.

El método *findAllEvents*, puesto que es una simple llamada al método *findAllEventsNoDate* del **EventoDao**, el cual ya está probado anteriormente.

- En la sección **BetService** :

El método *findOptionById*, puesto que es una llamada al método *find* del **OpcionDao**, el cual ya está probado con anterioridad. Tampoco se probará el caso de que el evento ya haya expirado, puesto que es una simple comprobación.

El método *findEvent* cuando hay menos eventos de los que se piden y cuando no hay palabras clave, puesto que ya se han realizado estas pruebas con el *findEvent* del **EventoDao**, que es el método al que llama.

El método *bet* cuando se intenta apostar sobre un Evento ya pasado , puesto que es una simple comprobación. Tampoco se probará cuando se intenta apostar sobre un Evento inexistente, puesto que es una llamada al *find* del **EventoDao**, el cual ya está probado con anterioridad; así como cuando es un usuario inexistente, puesto que es una llamada al método *find* de la clase otorgada *UserService*.

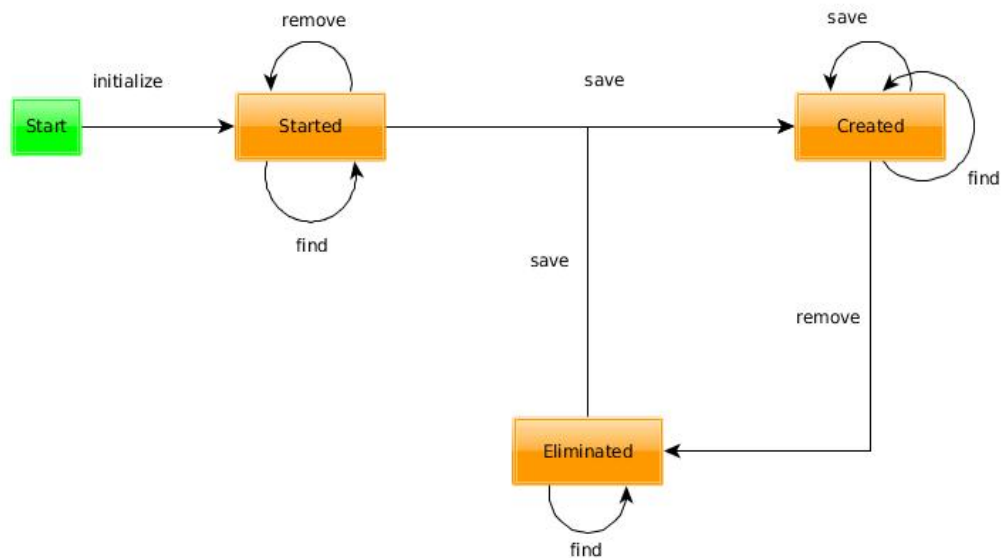
El método *checkBet* cuando hay menos Apuestas de las pedidas, puesto que es una llamada al método *findApuestasByIdUsuario* del **ApuestaDao**.

Los métodos *findEventById*, *getNumberOfEvents*, *getNumberOfBets*, *showWinners* y *findAllEvents*, puesto que son simples llamadas a los Daos correspondientes, los cuales han sido probados con anterioridad.



## 2. Diagramas

### 2.1. GenericDao



## 3. ApuestaDao

### 3.1. PR-UN-AD-01

**Unidad :** ApuestaDao

**Método :** save

**Motivación :** Creación de una apuesta en la base de datos no previamente creado

**Entrada :** Apuesta

**Salida :** Poder continuar con el programa

**Inicialización :** El usuario que realizará la apuesta, un Evento, un TipoApuesta asociado a ese Evento y una Opcion asociada a ese TipoApuesta sobre el que realizar la apuesta.

### **3.2. PR-UN-AD-02**

**Unidad :** ApuestaDao

**Método :** save

**Motivación :** Modificación de una apuesta en la base de datos previamente creado

**Entrada :** Apuesta

**Salida :** Poder continuar con el programa

**Inicialización :** El usuario que realizará la apuesta, un Evento, un TipoApuesta asociado a ese Evento y una Opcion asociada a ese TipoApuesta sobre el que realizar la apuesta, y la apuesta

### **3.3. PR-UN-AD-03**

**Unidad :** ApuestaDao

**Método :** find

**Motivación :** Recuperar de la base de datos una apuesta

**Entrada :** Long con el identificador de la apuesta

**Salida :** La apuesta previamente creada

**Inicialización :** El usuario que realizará la apuesta, un Evento, un TipoApuesta asociado a ese Evento y una Opcion asociada a ese TipoApuesta sobre el que realizar la apuesta, y la apuesta

### **3.4. PR-UN-AD-04**

**Unidad :** ApuestaDao

**Método :** find

**Motivación :** Recuperar de la base de datos una apuesta no existente

**Entrada :** Long con el identificador de la prueba

**Salida :** La excepción *InstanceNotFoundException*

**Inicialización :** -

### **3.5. PR-UN-AD-05**

**Unidad :** ApuestaDao

**Método :** save / find

**Motivación :** Actualizar los datos de una apuesta existente y después comprobar que al recuperarlos sean iguales

**Entrada :** Apuesta a actualizar / Long con el identificador de la apuesta actualizada

**Salida :** La apuesta con los datos actualizados

**Inicialización :** Una apuesta

### **3.6. PR-UN-AD-06**

**Unidad :** ApuestaDao

**Método :** remove

**Motivación :** Eliminar una apuesta de la base de datos

**Entrada :** Long con el identificador de la apuesta

**Salida :** - Poder continuar con el programa

**Inicialización :** El usuario que realizará la apuesta, un Evento, un TipoApuesta asociado a ese Evento y una Opcion asociada a ese TipoApuesta sobre el que realizar la apuesta, y la apuesta

### 3.7. PR-UN-AD-07

**Unidad :** ApuestaDao

**Método :** remove

**Motivación :** Eliminar una apuesta inexistente de la base de datos

**Entrada :** Long con el identificador de la apuesta

**Salida :** La excepción *InstanceNotFoundException*

**Inicialización :** -

### 3.8. PR-UN-AD-08

**Unidad :** ApuestaDao

**Método :** save / find / remove / find

**Motivación :** Comprobar que se crea y se elimina una apuesta

**Entrada :** Apuesta / Long con el identificador / Long con el identificador  
/ Long con el identificador

**Salida :** La excepción *InstanceNotFoundException*

**Inicialización :** Una Categoria, un Evento asociada a esa Categoria, TipoApuesta asociado a ese Evento y Opcion asociado a ese TipoApuesta.

### 3.9. PR-UN-AD-09

**Unidad :** ApuestaDao

**Método :** findApuestasByIdUsuario

**Motivación :** Comprobar que se recupera correctamente las apuestas de un usuario con la cantidad esperada y empezando en el índice esperado

**Entrada :** Long con el identificador del usuario / Índice inicial / Cantidad de apuestas

**Salida :** Las X primeras apuestas del usuario ordenadas por fecha

**Inicialización :** Varios usuarios que realizarán apuestas, varios Eventos, varios TipoApuestas asociado a esos Eventos ,varias Opciones asociada a esos TipoApuesta sobre el que realizar la apuesta, y varias apuestas.

### **3.10. PR-UN-AD-10**

**Unidad :** ApuestaDao

**Método :** findApuestasByIdUsuario

**Motivación :** Comprobar que se recupera correctamente las apuestas de un usuario con la cantidad esperada y empezando en el índice esperado, cuando hay menos apuestas de las que se piden

**Entrada :** Long con el identificador del usuario / Índice inicial / Cantidad de apuestas

**Salida :** Las primeras apuestas del usuario ordenadas por fecha

**Inicialización :** Varios usuarios que realizarán apuestas, varios Eventos, varios TipoApuestas asociado a esos Eventos ,varias Opciones asociada a esos TipoApuesta sobre el que realizar la apuesta, y varias apuestas.

### **3.11. PR-UN-AD-11**

**Unidad :** ApuestaDao

**Método :** findNumberOfBets

**Motivación :** Comprobar que se recuperan correctamente la cantidad de apuestas que un usuario tiene realizada

**Entrada :** Long con el identificador del Usuario

**Salida :** El número de apuestas que tiene el usuario

**Inicialización :** Varios usuarios que realizarán apuestas, varios Eventos, varios TipoApuestas asociado a esos Eventos ,varias Opciones asociada a esos TipoApuesta sobre el que realizar la apuesta, y varias apuestas.

### **3.12. PR-UN-AD-12**

**Unidad :** ApuestaDao

**Método :** findNumberOfBets

**Motivación :** Comprobar que se recuperan correctamente la cantidad de apuestas que un usuario tiene realizada cuando no tiene ninguna hecha

**Entrada :** Long con el identificador del Usuario

**Salida :** 0

**Inicialización :** Usuario

## **4. OpcionDao**

### **4.1. PR-UN-OD-01**

**Unidad :** OpcionDao

**Método :** showWinners

**Motivación :** Comprobación de que no devuelva ninguna opción ganadora cuando no exista ninguna

**Entrada :** Long con el identificador del TipoApuesta

**Salida :** Una lista vacía

**Inicialización :** Un usuario, un Evento, un TipoApuesta sobre ese evento y Opciones asociadas a ese TipoApuesta, pero ninguna marcada como ganadora.

### **4.2. PR-UN-OD-02**

**Unidad :** OpcionDao

**Método :** showWinners

**Motivación :** Comprobación de que devuelva correctamente las opciones ganadoras de un tipo de apuesta cuando existen algunas marcadas como ganadoras

**Entrada :** Long con el identificador del TipoApuesta

**Salida :** La lista con opciones ganadoras

**Inicialización :** Un usuario, un Evento, un TipoApuesta sobre ese evento y Opciones asociadas a ese TipoApuesta, pero ninguna marcada como ganadora.

### **4.3. PR-UN-OD-03**

**Unidad :** OpciónDao

**Método :** save

**Motivación :** Creación de una opcion de un tipo de apuesta en la base de datos previamente no creada.

**Entrada :** Objeto opción.

**Salida :** Poder continuar con el programa.

**Inicialización :** Una categoría, un Evento asociado a esa categoría y un TipoApuesta sobre ese evento.

### **4.4. PR-UN-OD-04**

**Unidad :** OpciónDao

**Método :** save

**Motivación :** Modificación de una opción en la base de datos previamente creada.

**Entrada :** Opción.

**Salida :** Poder continuar con el programa

**Inicialización :** Una categoría, un Evento asociado a esa categoría, un TipoApuesta sobre ese evento y la Opción asociada a ese TipoApuesta.

#### 4.5. PR-UN-OD-05

**Unidad :** OpciónDao

**Método :** find

**Motivación :** Recuperar de la base de datos una opción.

**Entrada :** Long con el identificador de la apuesta

**Salida :** La opción previamente creada.

**Inicialización :** Una categoría, un Evento asociado a esa categoría, un TipoApuesta sobre ese evento y la Opción asociada a ese TipoApuesta.

#### 4.6. PR-UN-OD-06

**Unidad :** OpciónDao

**Método :** find

**Motivación :** Recuperar de la base de datos una opción no existente.

**Entrada :** Long con el identificador de la prueba.

**Salida :** La excepción *InstanceNotFoundException*

**Inicialización :** -

#### 4.7. PR-UN-OD-07

**Unidad :** OpciónDao

**Método :** save / find

**Motivación :** Actualizar los datos de una opción existente y después comprobar que al recuperarlos sean iguales

**Entrada :** Opción a actualizar / Long con el identificador de la opción actualizada.

**Salida :** La opción con los datos actualizados.

**Inicialización :** Una categoría, un Evento asociado a esa categoria, un TipoApuesta sobre ese evento y la Opción asociada a ese TipoApuesta.



#### 4.8. PR-UN-OD-08

**Unidad :** OpciónDao

**Método :** remove

**Motivación :** Eliminar una opción de la base de datos.

**Entrada :** Long con el identificador de la opción

**Salida :** - Poder continuar con el programa

**Inicialización :** Una categoria, un Evento asociado a esa categoria, un TipoApuesta sobre ese evento y la Opción asociada a ese TipoApuesta.

#### 4.9. PR-UN-OD-09

**Unidad :** OpciónDao

**Método :** remove

**Motivación :** Eliminar una opción inexistente de la base de datos

**Entrada :** Long con el identificador de la opción.

**Salida :** La excepción *InstanceNotFoundException*

**Inicialización :** -

#### 4.10. PR-UN-OD-10

**Unidad :** OpciónDao

**Método :** save / find / remove / find

**Motivación :** Comprobar que se crea y se elimina una opción.

**Entrada :** Opción / Long con el identificador de la opción / Long con el identificador de la opción / Long con el identificador de la opción.

**Salida :** La excepción *InstanceNotFoundException*

**Inicialización :** Una Categoria, un Evento asociada a esa Categoria, TipoApuesta asociado a ese Evento.

## **5. TipoApuestaDao**

### **5.1. PR-UN-TAD-01**

**Unidad :** TipoApuestaDao

**Método :** findDuplicateBetTypes

**Motivación :** Comprobar que devuelve true cuando hay un TipoApuesta con la misma pregunta para el mismo Evento

**Entrada :** Pregunta del TipoApuesta / Long con el identificador del Evento

**Salida :** true

**Inicialización :** Una Categoría, un Evento asociado a esa Categoría y un TipoApuesta asociado a ese Evento

### **5.2. PR-UN-TAD-02**

**Unidad :** TipoApuestaDao

**Método :** save

**Motivación :** Creación de un tipo de apuesta en la base de datos. previamente no creada.

**Entrada :** Objeto TipoApuesta.

**Salida :** Poder continuar con el programa.

**Inicialización :** Una categoría, un Evento asociado a esa categoría.

### **5.3. PR-UN-TAD-03**

**Unidad :** TipoApuestaDao

**Método :** save

**Motivación :** Modificación de un tipo de apuesta en la base de datos previamente creada.

**Entrada :** TipoApuesta.

**Salida :** Poder continuar con el programa

**Inicialización :** Una categoría, un Evento asociado a esa categoría, un TipoApuesta sobre ese evento.

#### **5.4. PR-UN-TAD-04**

**Unidad :** TipoApuestaDao

**Método :** find

**Motivación :** Recuperar de la base de datos un TipoApuesta.

**Entrada :** Long con el identificador del TipoApuesta.

**Salida :** El tipo de apuesta previamente creada.

**Inicialización :** Una categoría, un Evento asociado a esa categoría, un TipoApuesta sobre ese evento.

#### **5.5. PR-UN-TAD-05**

**Unidad :** TipoApuestaDao

**Método :** find

**Motivación :** Recuperar de la base de datos un TipoApuesta no existente.

**Entrada :** Long con el identificador de la prueba

**Salida :** La excepción *InstanceNotFoundException*

**Inicialización :** -

#### **5.6. PR-UN-TAD-06**

**Unidad :** TipoApuestaDao

**Método :** save / find

**Motivación :** Actualizar los datos de un TipoApuesta existente y después comprobar que al recuperarlos sean iguales.

**Entrada :** TipoApuesta a actualizar / Long con el identificador del TipoApuesta a actualizar.

**Salida :** El TipoApuesta con los datos actualizados.

**Inicialización :** Una categoría, un Evento asociado a esa categoría, un TipoApuesta sobre ese evento.

### **5.7. PR-UN-TAD-07**

**Unidad :** TipoApuestaDao

**Método :** remove

**Motivación :** Eliminar un TipoApuesta de la base de datos.

**Entrada :** Long con el identificador del TipoApuesta.

**Salida :** - Poder continuar con el programa

**Inicialización :** Una categoría, un Evento asociado a esa categoría, un TipoApuesta sobre ese evento.

### **5.8. PR-UN-TAD-08**

**Unidad :** TipoApuestaDao

**Método :** remove

**Motivación :** Eliminar un TipoApuesta inexistente de la base de datos

**Entrada :** Long con el identificador del TipoApuesta.

**Salida :** La excepción *InstanceNotFoundException*

**Inicialización :** -

## 5.9. PR-UN-TAD-09

**Unidad :** TipoApuestaDao

**Método :** save / find / remove / find

**Motivación :** Comprobar que se crea y se elimina un TipoApuesta.

**Entrada :** Opción / Long con el identificador del TipoApuesta / Long con el identificador del TipoApuesta / Long con el identificador del TipoApuesta.

**Salida :** La excepción *InstanceNotFoundException*

**Inicialización :** Una Categoria, un Evento asociada a esa Categoria.

## 5.10. PR-UN-TAD-10

**Unidad :** TipoApuestaDao

**Método :** findDuplicateBetTypes

**Motivación :** Comprobar que devuelve false cuando no hay TipoApuesta con la misma pregunta para el mismo evento.

**Entrada :** Pregunta del TipoApuesta / Long con el identificador del Evento

**Salida :** true

**Inicialización :** -

# 6. CategoriaDao

## 6.1. PR-UN-CD-01

**Unidad :** CategoriaDao

**Método :** findCategories

**Motivación :** Obtener de la base de datos todas las categorías existentes para los eventos deportivos

**Entrada :** -

**Salida :** Todas las categorías encontradas

**Inicialización :** Varias Categorías

## **6.2. PR-UN-CD-12**

**Unidad :** CategoriaDao

**Método :** insertCategory

**Motivación :** Comprobar que no se pueden existir dos categorias con el mismo nombre.

**Entrada :** Un string igual en las dos instanciaciones del método.

**Salida :** Exception *ConstraintViolationException*

**Inicialización :** -

## **6.3. PR-UN-CD-03**

**Unidad :** CategoriaDao

**Método :** save

**Motivación :** Creación de una categoría en la base de datos. previamente no creada.

**Entrada :** Objeto Category.

**Salida :** Poder continuar con el programa.

**Inicialización :** -.

#### **6.4. PR-UN-CD-04**

**Unidad :** CategoriaDao

**Método :** save

**Motivación :** Modificación de una categoria en la base de datos previamente creada.

**Entrada :** Objeto tipo Category.

**Salida :** Poder continuar con el programa

**Inicialización :** Una categoría.

#### **6.5. PR-UN-CD-05**

**Unidad :** CategoriaDao

**Método :** find

**Motivación :** Recuperar de la base de datos una categoria.

**Entrada :** Long con el identificador de la category.

**Salida :** La categoria previamente creada.

**Inicialización :** Una categoría.

#### **6.6. PR-UN-CD-06**

**Unidad :** CategoriaDao

**Método :** find

**Motivación :** Recuperar de la base de datos una categoria no existente.

**Entrada :** Long con el identificador de la prueba.

**Salida :** La excepción *InstanceNotFoundException*

**Inicialización :** -

### 6.7. PR-UN-CD-07

**Unidad :** CategoriaDao

**Método :** save / find

**Motivación :** Actualizar los datos de una categoria existente y después comprobar que al recuperarlos sean iguales.

**Entrada :** Category a actualizar / Long con el identificador de la category a actualizar.

**Salida :** El TipoApuesta con los datos actualizados.

**Inicialización :** Una categoría.

### 6.8. PR-UN-CD-08

**Unidad :** CategoriaDao

**Método :** remove

**Motivación :** Eliminar una categoria de la base de datos.

**Entrada :** Long con el identificador del Category.

**Salida :** - Poder continuar con el programa

**Inicialización :** Una categoria.

### 6.9. PR-UN-CD-09

**Unidad :** CategoriaDao

**Método :** remove

**Motivación :** Eliminar una categoria inexistente de la base de datos.

**Entrada :** Long con el identificador de la category.

**Salida :** La excepción *InstanceNotFoundException*

**Inicialización :** -



## 6.10. PR-UN-CD-10

**Unidad :** CategoriaDao

**Método :** save / find / remove / find

**Motivación :** Comprobar que se crea y se elimina una categoria.

**Entrada :** Un objeto Category / Long con el identificador de la category  
/ Long con el identificador de la category / Long con el identificador  
de la category.

**Salida :** La excepción *InstanceNotFoundException*

**Inicialización :** -

## 6.11. PR-UN-CD-11

**Unidad :** CategoriaDao

**Método :** findCategories

**Motivación :** Obtener de la base de datos todas las categorías existentes  
para los eventos deportivos, que este caso comprobaremos que no se  
devuelve ninguna cuando no hay ninguna insertada en la base de datos.

**Entrada :** -

**Salida :** 0 categorias.

**Inicialización :** -

# 7. EventoDao

## 7.1. PR-UN-ED-01

**Unidad :** EventoDao

**Método :** findEvent

**Motivación :** Comprobar que encuentra la cantidad de eventos de una Categoría filtrados por palabras clave y posteriores a la fecha actual, empezando por un índice.

**Entrada :** Long con el identificador de la Categoría / Palabras clave / Índice inicial / Cantidad de eventos a recuperar

**Salida :** Los X primeros eventos ordenados por fecha

**Inicialización :** Eventos con fecha anterior y fecha posterior a la actual

## **7.2. PR-UN-ED-02**

**Unidad :** EventoDao

**Método :** findEvent

**Motivación :** Comprobar que encuentra la cantidad de eventos de una Categoría filtradas por palabras clave y posteriores a la fecha actual, empezando por un índice cuando hay menos soluciones que la cantidad pedida

**Entrada :** Long con el identificador de la Categoría / Palabras clave / Índice inicial / Cantidad de eventos a recuperar

**Salida :** Los eventos que coincidan

**Inicialización :** Eventos con fecha anterior y fecha posterior a la actual

## **7.3. PR-UN-ED-03**

**Unidad :** EventoDao

**Método :** findEvent

**Motivación :** Comprobar que encuentra la cantidad de eventos de una Categoría filtradas por palabras clave y posteriores a la fecha actual, empezando por un índice cuando no hay ningún Evento que coincida con las palabras clave.

**Entrada :** Long con el identificador de la Categoría / Palabras clave que no coincida que ningún evento / Índice inicial / Cantidad de eventos a recuperar

**Salida :** Una lista vacía

**Inicialización :** Eventos con fecha anterior y posterior a la actual

#### **7.4. PR-UN-ED-04**

**Unidad :** EventoDao

**Método :** findEvent

**Motivación :** Comprobar que busca la cantidad de eventos de una Categoría filtradas por palabras clave y posteriores a la fecha actual, empezando por un índice cuando no hay ninguna palabra clave

**Entrada :** Long con el identificador de la Categoría / Palabras clave vacías / Índice inicial / Cantidad de eventos a recuperar

**Salida :** Una lista con los X eventos de la misma categoría

**Inicialización :** Eventos con fecha anterior y posterior a la actual

#### **7.5. PR-UN-ED-05**

**Unidad :** EventoDao

**Método :** findDuplicateEvents

**Motivación :** Comprobar si devuelve true cuando hay un Evento de la misma Categoría programado para la misma fecha con el mismo nombre

**Entrada :** Nombre del evento / Fecha programada / Long con el identificador de la Categoría

**Salida :** true

**Inicialización :** Categoría / Evento con la misma fecha y mismo nombre

## **7.6. PR-UN-ED-06**

**Unidad :** EventoDao

**Método :** save

**Motivación :** Creación de un evento en la base de datos. previamente no creada.

**Entrada :** Objeto Evento.

**Salida :** Poder continuar con el programa.

**Inicialización :** Una categoria para asociar al evento a crear.

## **7.7. PR-UN-ED-07**

**Unidad :** EventoDao

**Método :** save

**Motivación :** Modificación de un evento en la base de datos previamente creada.

**Entrada :** Objeto tipo Evento.

**Salida :** Poder continuar con el programa

**Inicialización :** Una categoría y un evento asociado a esa categoría.

## **7.8. PR-UN-ED-08**

**Unidad :** EventoDao

**Método :** find

**Motivación :** Recuperar de la base de datos un evento.

**Entrada :** Long con el identificador del evento.

**Salida :** El evento previamente creado.

**Inicialización :** Una categoría y un evento asociado a esa categoría.

## **7.9. PR-UN-ED-09**

**Unidad :** EventoDao

**Método :** find

**Motivación :** Recuperar de la base de datos un evento no existente.

**Entrada :** Long con el identificador de la prueba.

**Salida :** La excepción *InstanceNotFoundException*

**Inicialización :** -

## **7.10. PR-UN-ED-10**

**Unidad :** EventoDao

**Método :** save / find

**Motivación :** Actualizar los datos de un evento existente y después comprobar que al recuperarlo sean iguales.

**Entrada :** Evento a actualizar / Long con el identificador del Evento a actualizar.

**Salida :** El Evento con los datos actualizados.

**Inicialización :** Una categoría y un evento asociado a esa categoría.

## **7.11. PR-UN-ED-11**

**Unidad :** EventoDao

**Método :** remove

**Motivación :** Eliminar un evento de la base de datos.

**Entrada :** Long con el identificador del Evento.

**Salida :** - Poder continuar con el programa.

**Inicialización :** Una categoría y un evento asociado a ella.

### **7.12. PR-UN-ED-12**

**Unidad :** EventoDao

**Método :** remove

**Motivación :** Eliminar un evento inexistente de la base de datos.

**Entrada :** Long con el identificador del Evento.

**Salida :** La excepción *InstanceNotFoundException*

**Inicialización :** -

### **7.13. PR-UN-ED-13**

**Unidad :** EventoDao

**Método :** save / find / remove / find

**Motivación :** Comprobar que se crea y se elimina un evento.

**Entrada :** Un objeto Evento / Long con el identificador del evento / Long con el identificador del evento / Long con el identificador del evento.

**Salida :** La excepción *InstanceNotFoundException*

**Inicialización :** Una categoría.

### **7.14. PR-UN-ED-14**

**Unidad :** EventoDao

**Método :** findEventNoDate

**Motivación :** Comprobar que encuentra la cantidad de eventos de una Categoría filtrados por palabras clave, empezando por un índice.

**Entrada :** Long con el identificador de la Categoría / Palabras clave / Índice inicial / Cantidad de eventos a recuperar

**Salida :** Los X primeros eventos ordenados por fecha

**Inicialización :** Eventos con fecha anterior y fecha posterior a la actual

### **7.15. PR-UN-ED-15**

**Unidad :** EventoDao

**Método :** findEventNoDate

**Motivación :** Comprobar que encuentra la cantidad de eventos de una Categoría filtradas por palabras clave, empezando por un índice cuando hay menos soluciones que la cantidad pedida

**Entrada :** Long con el identificador de la Categoría / Palabras clave / Índice inicial / Cantidad de eventos a recuperar

**Salida :** Los eventos que coincidan

**Inicialización :** Eventos con fecha anterior y fecha posterior a la actual

### **7.16. PR-UN-ED-16**

**Unidad :** EventoDao

**Método :** findEventNoDate

**Motivación :** Comprobar que encuentra la cantidad de eventos de una Categoría filtradas por palabras clave, empezando por un índice cuando no hay ningún Evento que coincida con las palabras clave.

**Entrada :** Long con el identificador de la Categoría / Palabras clave que no coincida que ningún evento / Índice inicial / Cantidad de eventos a recuperar

**Salida :** Una lista vacía

**Inicialización :** Eventos con fecha anterior y posterior a la actual

### **7.17. PR-UN-ED-17**

**Unidad :** EventoDao

**Método :** findEventNoDate

**Motivación :** Comprobar que busca la cantidad de eventos de una Categoría filtradas por palabras clave, empezando por un índice cuando no hay ninguna palabra clave

**Entrada :** Long con el identificador de la Categoría / Palabras clave vacías / Índice inicial / Cantidad de eventos a recuperar

**Salida :** Una lista con los X eventos de la misma categoría

**Inicialización :** Eventos con fecha anterior y posterior a la actual

### 7.18. PR-UN-ED-18

**Unidad :** EventoDao

**Método :** findDuplicateEvents

**Motivación :** Comprobar si devuelve false cuando no hay un Evento de la misma Categoría programado para la misma fecha con el mismo nombre

**Entrada :** Nombre del evento / Fecha programada / Long con el identificador de la Categoría

**Salida :** false

**Inicialización :** Categoría / Evento con la misma fecha y mismo nombre

### 7.19. PR-UN-ED-19

**Unidad :** EventoDao

**Método :** findNumberOfEvents

**Motivación :** Saber cuántos eventos hay en base datos conforme a las palabras clave, a la categoría introducida y a que esos eventos estén pendientes de producirse.

**Entrada :** Long con el identificador de una Categoría / Palabras clave

**Salida :** N° de eventos encontrados

**Inicialización :** Categoría(s) / Eventos asociados a esa categoria y a otras, que lleven a cabo en fechas futuras y pasadas.



## 7.20. PR-UN-ED-20

**Unidad :** EventoDao

**Método :** findNumberOfEvents

**Motivación :** Saber cuántos eventos hay en base datos conforme a las palabras clave no coincidentes con ningún evento, a la categoría introducida que tampoco coincida y a que esos eventos estén pendientes de producirse.

**Entrada :** Long con el identificador de una Categoría / Palabras clave

**Salida :** 0

**Inicialización :** Categoría / Eventos asociados a esa categoría y a otras, que lleven a cabo en fechas futuras y pasadas.

## 7.21. PR-UN-ED-21

**Unidad :** EventoDao

**Método :** findNumberOfEvents

**Motivación :** Saber cuántos eventos hay en base datos conforme a las palabras clave, a la categoría introducida.

**Entrada :** Long con el identificador de una Categoría / Palabras clave

**Salida :** N° de eventos encontrados.

**Inicialización :** Categoría(s) / Eventos asociados a esa categoría y a otras, que lleven a cabo en fechas futuras y pasadas.

## 7.22. PR-UN-ED-22

**Unidad :** EventoDao

**Método :** findNumberOfEvents

**Motivación :** Saber cuántos eventos hay en base datos conforme a las palabras clave no coincidentes con ningún evento, a la categoría introducida que tampoco coincida.

**Entrada :** Long con el identificador de una Categoría / Palabras clave

**Salida :** 0

**Inicialización :** Categoría / Eventos asociados a esa categoría y a otras, que lleven a cabo en fechas futuras y pasadas.

### **7.23. PR-UN-ED-23**

**Unidad :** EventoDao

**Método :** findAllEventsNoDate

**Motivación :** Comprobar que encuentra la cantidad de eventos asociados a una Categoría filtrados por palabras clave.

**Entrada :** Long con el identificador de la Categoría / Palabras clave

**Salida :** Todos los eventos ordenados por fecha.

**Inicialización :** Eventos con fecha anterior y fecha posterior a la actual

### **7.24. PR-UN-ED-24**

**Unidad :** EventoDao

**Método :** findAllEventsNoDate

**Motivación :** Comprobar que encuentra la cantidad de eventos asociados a una Categoría filtrados por palabras clave.

**Entrada :** Long con el identificador de la Categoría / Palabras clave que no coincida que ningún evento.

**Salida :** Una lista vacía.

**Inicialización :** Eventos con fecha anterior y posterior a la actual

### **7.25. PR-UN-ED-25**

**Unidad :** EventoDao

**Método :** findAllEventsNoDate

**Motivación :** Comprobar que busca la cantidad de eventos asociados a una Categoría filtrados por palabras clave.

**Entrada :** Long con el identificador de la Categoría / Palabras clave vacías.

**Salida :** Todos los eventos ordenados por fecha.

**Inicialización :** Eventos con fecha anterior y posterior a la actual

### **7.26. PR-UN-ED-26**

**Unidad :** EventoDao

**Método :** findAllEvents

**Motivación :** Comprobar que encuentra la cantidad de eventos asociados a una Categoría filtrados por palabras clave y posteriores a la fecha actual.

**Entrada :** Long con el identificador de la Categoría / Palabras clave

**Salida :** Todos los eventos ordenados por fecha.

**Inicialización :** Eventos con fecha anterior y fecha posterior a la actual

### **7.27. PR-UN-ED-27**

**Unidad :** EventoDao

**Método :** findAllEvents

**Motivación :** Comprobar que encuentra la cantidad de eventos asociados a una Categoría filtrados por palabras clave y posteriores a la fecha actual.

**Entrada :** Long con el identificador de la Categoría / Palabras clave que no coincida que ningún evento.

**Salida :** Una lista vacía.

**Inicialización :** Eventos con fecha anterior y posterior a la actual

## **7.28. PR-UN-ED-28**

**Unidad :** EventoDao

**Método :** findAllEvents

**Motivación :** Comprobar que busca la cantidad de eventos asociados a una Categoría filtrados por palabras clave y posteriores a la fecha actual.

**Entrada :** Long con el identificador de la Categoría / Palabras clave vacías.

**Salida :** Todos los eventos ordenados por fecha.

**Inicialización :** Eventos con fecha anterior y posterior a la actual

## **8. AdminService**

### **8.1. PR-UN-AS-01**

**Unidad :** AdminService

**Método :** createEvent

**Motivación :** Comprobar que crea correctamente el Evento en la base de datos

**Entrada :** Evento a crear / Long con el identificador de la Categoría

**Salida :** Evento creado

**Inicialización :** Categorías

## 8.2. PR-UN-AS-02

**Unidad :** AdminService

**Método :** createEvent

**Motivación :** Comprobación de que no inserta en la base de datos el Evento si ya existe previamente

**Entrada :** Evento creado previamente / Long con el identificador de la Categoría

**Salida :** La excepción *DuplicateInstanceException*

**Inicialización :** Evento

## 8.3. PR-UN-AS-03

**Unidad :** AdminService

**Método :** findEvent

**Motivación :** Comprobar que se encuentra los eventos en la base de datos que coincidan con los parámetros de búsqueda, indicando que hay más eventos a mostrar

**Entrada :** Long con el identificador de la Categoría / Palabras clave / Índice inicial / Cantidad de objetos a recuperar

**Salida :** EventoBlock con los Eventos coincidentes e indicando que hay más Eventos.

**Inicialización :** Varios eventos perteneciente a la misma Categoría

## 8.4. PR-UN-AS-04

**Unidad :** AdminService

**Método :** markAsWinner

**Motivación :** Comprobar que marca correctamente las Opciones indicadas como ganadoras.

**Entrada :** Una lista con los identificadores de las Opciones / El identificador del TipoApuesta,

**Salida :** Poder continuar con el programa

**Inicialización :** Evento / TipoApuestas del Evento / Opciones del TipoApuesta

### **8.5. PR-UN-AS-05**

**Unidad :** AdminService

**Método :** markAsWinner

**Motivación :** Comprobar que no marca las Opciones como ganadoras cuando ya estaban marcadas con un estado.

**Entrada :** Una lista con los identificadores de las Opciones / El identificador del TipoApuesta

**Salida :** La excepción AlreadySetOptionException

**Inicialización :** Evento / TipoApuestas del Evento / Opciones del TipoApuesta

### **8.6. PR-UN-AS-06**

**Unidad :** AdminService

**Método :** addBettingType

**Motivación :** Comprobar que inserta correctamente el TipoApuesta

**Entrada :** TipoApuesta a introducir en la base de datos / Identificador del evento al que asociar el TipoApuesta / Lista de Opciones a asociar al TipoApuesta

**Salida :** TipoApuesta creado

**Inicialización :** Evento a que asociar el TipoApuesta / Opciones a asociar al TipoApuesta

## 8.7. PR-UN-AS-07

**Unidad :** AdminService

**Método :** createEvent

**Motivación :** Comprobación de que no inserta en la base de datos el Evento si su fecha es anterior a la actual

**Entrada :** Evento con una fecha anterior a la actual / Long con el identificador de la Categoría

**Salida :** La excepción *InvalidEventDateException*

**Inicialización :** Categoría asociada al evento a crear.

## 8.8. PR-UN-AS-08

**Unidad :** AdminService

**Método :** createEvent

**Motivación :** Comprobación de que no inserta en la base de datos el Evento si la categoría que se le intenta asociar no existe.

**Entrada :** Evento / Long con el identificador de la Categoría no existente.

**Salida :** La excepción *IntanceNotFoundException*

**Inicialización :** -.

## 8.9. PR-UN-AS-09

**Unidad :** AdminService

**Método :** markAsWinner

**Motivación :** Comprobar que no marca varias Opciones como ganadoras cuando el tipo apuesta solo permite marcar una como ganadora.

**Entrada :** Una lista con los identificadores de las Opciones / El identificador del TipoApuesta

**Salida :** La excepción *NotMultipleOptionsException*

**Inicialización :** Evento / TipoApuesta del Evento con el campo múltiple seteado a false / Opciones del TipoApuesta

## 8.10. PR-UN-AS-10

**Unidad :** AdminService

**Método :** markAsWinner

**Motivación :** Comprobar que no marca varias Opciones como ganadoras cuando las opciones a marcar no están asociadas al TipoApuesta introducido.

**Entrada :** Una lista con los identificadores de las Opciones asociadas a diferentes tipos de apuesta / El identificador del TipoApuesta

**Salida :** La excepción *IllegalParameterException*

**Inicialización :** Evento / TipoApuesta del Evento / Opciones del TipoApuesta y otras asociadas a otro TipoApuesta.

## 8.11. PR-UN-AS-11

**Unidad :** AdminService

**Método :** markAsWinner

**Motivación :** Comprobar que no marca varias Opciones como ganadoras cuando el evento al que referencian aún no se celebró.

**Entrada :** Una lista con los identificadores de las Opciones / El identificador del TipoApuesta

**Salida :** La excepción *ExpiredEventException*

**Inicialización :** Evento con fecha de celebración posterior a la ejecución de la prueba/ TipoApuesta del Evento / Opciones del TipoApuesta.



### 8.12. PR-UN-AS-12

**Unidad :** AdminService

**Método :** markAsWinner

**Motivación :** Comprobar que no marca varias Opciones como ganadoras cuando el evento al que referencian aún no se celebró.

**Entrada :** Una lista con los identificadores de las Opciones no creados / El identificador de un TipoApuesta inexistente.

**Salida :** La excepción *InstanceNotFoundException*

**Inicialización :** -

### 8.13. PR-UN-AS-13

**Unidad :** AdminService

**Método :** findCategoriaByID

- Es una llamada a un método del CategoryDao, cuya prueba fue diseñada anteriormente.

### 8.14. PR-UN-AS-14

**Unidad :** AdminService

**Método :** addBettingType

**Motivación :** Comprobar que no se permite crear un tipoApuesta sobre un evento ya pasado.

**Entrada :** TipoApuesta a introducir en la base de datos / Identificador del evento al que asociar el TipoApuesta / Lista de Opciones a asociar al TipoApuesta

**Salida :** La excepción *InvalidEventDateException*

**Inicialización :** Categoria para asociar el evento, Evento a que asociar el TipoApuesta con una fecha anterior a la ejecución de la prueba.

### 8.15. PR-UN-AS-15

**Unidad :** AdminService

**Método :** addBettingType

**Motivación :** Comprobar que no se permite crear un TipoApuesta si ya existe un tipo de apuesta con la mismas características.

**Entrada :** TipoApuesta previamente creado / Identificador del evento al que asociar el TipoApuesta / Lista de Opciones a asociar al TipoApuesta

**Salida :** La excepción *DuplicateInstanceException*

**Inicialización :** Categoria, Evento, TipoApuesta asociado al evento creado/ Opciones asociados al TipoApuesta.

### 8.16. PR-UN-AS-16

**Unidad :** AdminService

**Método :** addBettingType

**Motivación :** Comprobar que no se permite crear un TipoApuesta si el evento al que estará asociado no existe.

**Entrada :** TipoApuesta a crear / Un long / Lista de Opciones a asociar al TipoApuesta

**Salida :** La excepción *InstanceNotFoundException*

**Inicialización :** -

### 8.17. PR-UN-AS-17

**Unidad :** AdminService

**Método :** addOption

**Motivación :** Comprobación de que no inserta en la base de datos una opción si el tipo de apuesta al que irá asociado no existe en base de datos.

**Entrada :** Long / Nombre de la opción/ Cuota de ganancia/ Estado de la opción.

**Salida :** La excepción *IntanceNotFoundExpection*

**Inicialización :** -.

## 8.18. PR-UN-AS-18

**Unidad :** AdminService

**Método :** addOption

**Motivación :** Comprobación de que no inserta en la base de datos una opción si el evento al que está asociado el tipo de apuesta al que se quiere vincular la opción ya sucedió.

**Entrada :** Id del TipoApuesta / Nombre de la opción/ Cuota de ganancia/ Estado de la opción.

**Salida :** La excepción *InvalidEventDateException*

**Inicialización :** Category / Evento con una fecha de celebración anterior al de la prueba/ TipoApuesta asociada al evento.

## 8.19. PR-UN-AS-19

**Unidad :** AdminService

**Método :** addOption

**Motivación :** Comprobación de que se inserta correctamente un opción.

**Entrada :** Id del TipoApuesta / Nombre de la opción/ Cuota de ganancia/ Estado de la opción.

**Salida :** La opción creada.

**Inicialización :** Category / Evento/ TipoApuesta asociada al evento.

## **8.20. PR-UN-AS-20**

**Unidad :** AdminService

**Método :** getNumberOfEvents

- Es una llamada a un método del EventoDao, cuya prueba fue diseñada anteriormente.

## **8.21. PR-UN-AS-21**

**Unidad :** AdminService

**Método :** findTipoApuesta

- Es una llamada a un método del TipoApuestaDao, cuya prueba fue diseñada anteriormente.

## **8.22. PR-UN-AS-22**

**Unidad :** AdminService

**Método :** findOptionById

- Es una llamada a un método del OptionDao, cuya prueba fue diseñada anteriormente.

## **8.23. PR-UN-AS-23**

**Unidad :** AdminService

**Método :** findAllEvents

- Es una llamada a un método del EventoDao, cuya prueba fue diseñada anteriormente.

=====

## 9. BetService

### 9.1. PR-UN-BS-01

**Unidad :** BetService

**Método :** findEvent

**Motivación :** Comprobar que se busca correctamente los Eventos según los parámetros indicados, indicando además que hay más Eventos coincidentes.

**Entrada :** Identificador de la Categoría / Palabras clave / Índice inicial / Cantidad de Eventos a recuperar

**Salida :** EventoBlock con los Eventos encontrados y un true en el indicador de que existen más Eventos coincidentes.

**Inicialización :** Categoría / Eventos asociados a esa Categoría

### 9.2. PR-UN-BS-02

**Unidad :** BetService

**Método :** bet

**Motivación :** Comprobar que se realice correctamente una Apuesta sobre una Opción para el Usuario correspondiente.

**Entrada :** Identificador de la Opción / La cantidad a apostar por esa Opción / Identificador del Usuario

**Salida :** La Apuesta creada

**Inicialización :** Usuario / Evento / TipoApuesta asociada a ese evento / Opción asociada a ese TipoApuesta

### 9.3. PR-UN-BS-03

**Unidad :** BetService

**Método :** checkBet

**Motivación :** Comprobación de que se recuperan la cantidad de Apuestas indicadas de un Usuario indicando por un índice inicial, devolviendo un true para indicar que hay más.

**Entrada :** Identificador del Usuario / Índice inicial / Cantidad de Apuestas a recuperar

**Salida :** Las Apuestas del Usuario ordenadas por fecha, y un true en el indicador de que hay más Apuestas.

**Inicialización :** Usuario / Apuestas realizadas por ese Usuario

### 9.4. PR-UN-BS-04

**Unidad :** BetService

**Método :** bet

**Motivación :** Comprobación de que cuando se quiere realizar una apuesta con una cantidad negativa de dinero, no se permite

**Entrada :** Identificador de la Opción / La cantidad a apostar por esa Opción / Identificador del Usuario

**Salida :** La excepción *NegativeAmountException*

**Inicialización :** Usuario / Evento / TipoApuesta asociada a ese evento / Opción asociada a ese TipoApuesta

### 9.5. PR-UN-BS-05

**Unidad :** BetService

**Método :** findAllCategories

- Es una llamada a un método del CategoryDao, cuya prueba fue diseñada anteriormente.

## **9.6. PR-UN-BS-06**

**Unidad :** BetService

**Método :** findOptionById

**Motivación :** Comprobar que se encuentra una opción buscando por su Id.

**Entrada :** Identificador de la Opción.

**Salida :** La opción

**Inicialización :** Usuario / Evento / TipoApuesta asociada a ese evento /  
Opción asociada a ese TipoApuesta

## **9.7. PR-UN-BS-07**

**Unidad :** BetService

**Método :** findEventById

- Es una llamada a un método del EventoDao, cuya prueba fue diseñada anteriormente.

## **9.8. PR-UN-BS-08**

**Unidad :** BetService

**Método :** getNumberOfEvents

- Es una llamada a un método del EventoDao, cuya prueba fue diseñada anteriormente.

## **9.9. PR-UN-BS-09**

**Unidad :** BetService

**Método :** getNumberOfBets

- Es una llamada a un método del ApuestaDao, cuya prueba fue diseñada anteriormente.

### 9.10. PR-UN-BS-10

**Unidad :** BetService

**Método :** showWinners

- Es una llamada a un método del OpcionDao, cuya prueba fue diseñada anteriormente.

### 9.11. PR-UN-BS-10

**Unidad :** BetService

**Método :** findAllEvents

- Es una llamada a un método del EventoDao, cuya prueba fue diseñada anteriormente.

### 9.12. PR-UN-BS-11

**Unidad :** BetService

**Método :** findOptionById

**Motivación :** Comprobar que no se encuentra una opción buscando por su Id si previamente no se creó.

**Entrada :** Long.

**Salida :** La excepción *IntanceNotFoundException*

**Inicialización :** -

### 9.13. PR-UN-BS-12

**Unidad :** BetService

**Método :** findOptionById

**Motivación :** Comprobar que no se devuelve la opción buscada si el evento al que está asociada ya sucedió.



**Entrada :** Id de la opción creada.

**Salida :** La excepción *ExpiredEventException*

**Inicialización :** Usuario / Evento con una fecha anterior al momento en que se ejecute la prueba / TipoApuesta asociada a ese evento / Opción asociada a ese TipoApuesta

#### 9.14. PR-UN-BS-13

**Unidad :** BetService

**Método :** bet

**Motivación :** Comprobar que no se permite apostar si el evento ya sucedió.

**Entrada :** Identificador de la Opción / La cantidad a apostar por esa Opción / Identificador del Usuario

**Salida :** La excepción *ExpiredEventException*

**Inicialización :** Usuario / Evento con el campo fecha anterior al momento de la prueba / TipoApuesta asociada a ese evento / Opción asociada a ese TipoApuesta

#### 9.15. PR-UN-BS-14

**Unidad :** BetService

**Método :** bet

**Motivación :** Comprobar que no se permite apostar si el usuario indicado no existe.

**Entrada :** Identificador de la Opción / La cantidad a apostar por esa Opción / Long

**Salida :** La excepción *InstanceNotFoundException*

**Inicialización :** Evento con el campo fecha anterior al momento de la prueba / TipoApuesta asociada a ese evento / Opción asociada a ese TipoApuesta .

## 9.16. PR-UN-BS-15

**Unidad :** BetService

**Método :** bet

**Motivación :** Comprobar que no se permite apostar si la opción indicada no existe.

**Entrada :** Long / La cantidad a apostar por esa Opción / Identificador del Usuario

**Salida :** La excepción *InstanceNotFoundException*

**Inicialización :** Usuario.