

Plan de Pruebas

Historia de Usuario: *Marcar evento*

Se aplicarán los siguientes niveles de prueba:

- **Pruebas de aceptación.** Las realizará el *Product Owner* a mano.
- **Pruebas de interfaz gráfica.** Se realizarán haciendo uso de la librería **Espresso**.
- **Pruebas de integración.** Se realizarán haciendo uso del *framework* **JUnit** y la librería **Mockito**.
- **Pruebas unitarias.** Se realizarán haciendo uso del *framework* **JUnit** y la librería **Mockito**.

A continuación, se muestra una especificación detallada de los casos de prueba a aplicar en cada nivel mencionado anteriormente.

PRUEBAS DE ACEPTACIÓN

En base a los criterios de aceptación especificados por el *Product Owner* se identifican los siguientes escenarios:

A1. Marcar evento

- a. Marcado exitoso
- b. Desmarcado exitoso
- c. Error al marcar evento
- d. Error al desmarcar evento

A partir de esto, el cliente realizaría las siguientes pruebas, correspondientes a los casos mencionados anteriormente:

Prueba 00: Marcado exitoso.

1. El sistema muestra la lista de los eventos cargados.
2. El usuario selecciona un evento como favorito pulsando el botón de forma de estrella.
3. El sistema guarda el elemento en la lista de favoritos de la aplicación.
4. El sistema muestra el botón en forma de estrella coloreado.

Prueba 01: Desmarcado exitoso.

1. El sistema muestra la lista de los eventos cargados.
2. El usuario selecciona un evento que tiene el botón de favoritos coloreado.
3. El sistema elimina el elemento en la lista de favoritos de la aplicación.
4. El sistema muestra el botón en forma de estrella vaciado.

Prueba 02: Error al marcar evento.

1. El sistema muestra la lista de los eventos cargados.
2. El usuario selecciona un evento como favorito pulsando el botón de forma de estrella.
3. El sistema intenta guardar el elemento en la lista de favoritos de la aplicación, con resultado erróneo.
4. La imagen asociada al evento no varía.

Prueba 03: Error al desmarcar evento.

1. El sistema muestra la lista de los eventos cargados.
2. El usuario selecciona un evento para desmarcar como favorito pulsando el botón de forma de estrella.

3. El sistema intenta eliminar el elemento en la lista de favoritos de la aplicación, con resultado erróneo.
4. La imagen asociada al evento no varía.

PRUEBAS DE INTERFAZ GRÁFICA

En este caso se programarán, utilizando **Espresso**, los escenarios definidos en el apartado anterior. La nomenclatura A1.x de las pruebas de aceptación se corresponde con UIT1.x en las pruebas de interfaz gráfica.

Los casos de prueba definidos para este escenario, suponiendo que el estado inicial del sistema es el reflejado en el archivo [agenda_cultural.json](#), son los que se muestran en la tabla 1.

Tabla 1. Casos de prueba de interfaz gráfica

Identificador	Selección	Resultado
UIT1.a	“Abierto el plazo de inscripción para el Concurso Internacional de Piano de Santander Paloma O'Shea”	Relleno de imagen de favorito.
UIT1.b	“Abierto el plazo de inscripción para el Concurso Internacional de Piano de Santander Paloma O'Shea”	Vaciado de imagen de favorito.
UIT1.c	“En busca de vida en Marte”	No variación de imagen
UIT1.d	“En busca de vida en Marte”	No variación de imagen

Para simplificar la columna de “Selección” se ha indicado el título del evento que se desea seleccionar.

PRUEBAS DE INTEGRACIÓN

En este caso no hay ninguna llamada a la capa de DAO desde la capa de negocio por lo que no serán necesarias realizar pruebas de integración.

PRUEBAS UNITARIAS

Pruebas unitarias de la capa de negocio: en este caso probamos el funcionamiento de métodos de la clase `EventsPresenter` y `GestionarFavoritosUsuario`.

- Método `onFavouriteClicked(int eventIndex, Boolean isClicked, IGestionarFavoritos sharedPref)`

Identificador	Entrada	Valor esperado
---------------	---------	----------------

UT.1a	(1, false,sharedPref)	SharedPreferences: “favoritos”, “1”
UT.1b	(2, false,sharedPref)	SharedPreferences: “favoritos”, “1, 2”
UT.1c	(1,true,sharedPref)	SharedPreferences: “favoritos”, “2”
UT.1d	(0,false,sharedPref)	SharedPreferences: “favoritos”, “2”
UT.1e	(-1,false,sharedPref)	SharedPreferences: “favoritos”, “2”
UT.1f	(99999,false,sharedPref)	SharedPreferences: “favoritos”, “2”

Para simplificar la columna de “Valor esperado” hemos indicado la clave y el valor del objeto “SharedPreferences”. Estas pruebas están basadas en que la lista de favoritos inicialmente se encuentra vacía. En el “Valor Esperado” de los últimos tres casos de prueba no está especificado que se lance ninguna excepción, únicamente no deberá incluirlos en “SharedPreferences”.

- Método `setFavourite(int eventIndex, Event[*] cachedEvents)`

Identificador	Entrada	Valor esperado
UT.2a	(3,cachedEvents)	SharedPreferences: “favoritos”, “3”
UT.2b	(0,cachedEvents)	SharedPreferences: “favoritos”, “0,3”
UT.2c	(-1,cachedEvents)	java.lang.IndexOutOfBoundsException
UT.2d	(9999999,cachedEvents)	java.lang.IndexOutOfBoundsException

Al ser los métodos de tipo “void” para realizar la comprobación de estas pruebas se tendrá que tener en cuenta las “SharedPreferences” que son comunes a toda la aplicación.

- Método `onRemoveFavourite(int eventIndex, Event[*] cachedEvents)`

Identificador	Entrada	Valor esperado
UT.3a	(3,cachedEvents)	SharedPreferences: “favoritos”, “”
UT.3b	(null,cachedEvents)	SharedPreferences: “favoritos”, “”
UT.3c	(0,cachedEvents)	SharedPreferences: “favoritos”, “”
UT.3d	(-1,cachedEvents)	SharedPreferences: “favoritos”, “”
UT.3e	(9999999,cachedEvents)	SharedPreferences: “favoritos”, “”

Estas pruebas se ejecutarán después de las descritas anteriormente por eso el método UT.3a eliminará correctamente . En el “Valor Esperado” de los últimos cuatro casos de prueba no está especificado que se lance ninguna excepción, únicamente no deberá incluirlos en “SharedPreferences”.

REPORTE FINAL

Plan de pruebas realizado por: Diego San Román
Pruebas codificadas por: Iván González del Pozo
Número de fallos encontrados: 0