

# PLAN DE PRUEBAS PARA LA HISTORIO DE USUARIO: AÑADIR EVENTO A FAVORITOS

Arronti Cultura

## 1. Pruebas de Aceptación

### Prueba 00: Éxito al añadir a favoritos

1. El usuario marca un evento como favorito.
2. El sistema añade el evento a la lista de favoritos.
3. Se verifica que el evento se ha añadido correctamente a la lista de favoritos.
4. Se verifica que el evento se muestra en la pantalla de favoritos.
5. Se verifica que el icono que indica que el evento está en favoritos se muestra encendido.

### Prueba 01: Éxito al añadir a favoritos que se muestre al inicio de la lista

1. El usuario marca un evento como favorito.
2. El sistema añade el evento a la lista de favoritos.
3. Se verifica que el evento se ha añadido correctamente a la lista de favoritos.
4. Se verifica que el evento se muestra en la pantalla de favoritos.
5. Se verifica que el icono que indica que el evento está en favoritos se muestra encendido.
6. Se verifica que el evento recién añadido se muestra el primero en la lista de favoritos.

## 2. Pruebas Unitarias

Los casos de prueba para el método *onFavouriteEventsClicked()* de la clase *EventsDetailPresenter* se encuentran codificados en la clase *EventsDetailPresenterTest* y son los siguientes:

```
public void onFavouriteEventsClicked(Event event, boolean eliminar);
```

Identificador	Entrada	Valor Esperado
UGIC.1a	(evento, false)	Se realiza la petición de añadir el evento a la lista de favoritos
UGIC.1b	(evento, true)	Se realiza la petición de eliminar el evento de la lista de favoritos
UGIC.1c	(null, false)	Error: No se puede añadir un evento nulo a favoritos
UGIC.1d	(null, true)	Error: No se puede eliminar un evento nulo de favoritos

## 3. Pruebas de Integración

A continuación se muestra una especificación detallada de los casos de prueba a aplicar utilizando como fuente de datos para todas las pruebas la disponible en: [https://personales.unican.es/rivasjm/resources/agenda\\_cultural.json](https://personales.unican.es/rivasjm/resources/agenda_cultural.json).

Los casos de prueba para el método *onFavouriteEventsClicked()* de la clase *EventsDetailPresenter* se encuentran codificados en la clase *EventsDetailPresenterITest* y son los siguientes:

```
public void onFavouriteEventsClicked(Event event, boolean eliminar);
```

Identificador	Entrada	Valor Esperado
IGIC.1a	(evento, false)	Se añade el evento a la lista de favoritos
IGIC.1b	(evento, true)	Se quita el evento de la lista de favoritos

Los casos de prueba para el método *newFavouriteEvent()* de la clase *AccessSharedPrefs* se encuentran codificados en la clase *AccessSharedPrefsITest* y son los siguientes:

```
public static void newFavouriteEvent(int id);
```

Identificador	Entrada	Valor Esperado
IGIC.2a	(idEvento) idEvento de un evento que existe y no está en favoritos	Se añade el evento a la lista de favoritos
IGIC.2b	(idEvento) idEvento de un evento que no existe	No se añade el evento a la lista de favoritos
IGIC.2c	(idEvento) idEvento de un evento que existe y ya está en favoritos	El evento se mantiene en la lista de favoritos

#### 4. Pruebas de Interfaz

A continuación se muestra una especificación detallada de los casos de prueba a aplicar utilizando como fuente de datos para todas las pruebas la disponible en: [https://personales.unican.es/rivasjm/resources/agenda\\_cultural.json](https://personales.unican.es/rivasjm/resources/agenda_cultural.json).

Las pruebas de interfaz serán automatizadas a través de *Espresso* y se encuentran codificadas en la clase *AnhadirEventoAFavoritosUITest*:

Identificador	Entrada	Resultado
IVF.1a	Se pulsa el botón de favoritos en el evento "Gabinete de estampas virtual de la UC" (no aparece relleno)	Se rellena la estrella de dicho evento de color amarillo y se muestra en la lista de favoritos
IVF.1b	Se pulsa el botón de favoritos en el evento "Arte Postal" (aparece relleno)	Se rellena la estrella de dicho evento de color amarillo y se muestra en la

		lista de favoritos por encima del evento "Gabinete de estampas virtual de la UC"
--	--	--

## 5. Ejecución de las pruebas y fallos encontrados

### 5.1 Pruebas Unitarias

Al ejecutar las pruebas unitarias del metodo `onFavouriteEventsClicked()` nos dimos cuenta de que no se había contemplado la posibilidad de que el evento fuese nulo.

### 5.2 Pruebas de Integración

Al ejecutar las pruebas de integración del método `newFavouriteEvent()` nos hemos dado cuenta de que la prueba IGIC2.b y IGIC2.c fallaban porque no se contemplaban todos los casos de error.

En el caso IGIC2.b no se comprobaba que el evento no existe en el repositorio y por lo tanto no se tendría que poder añadir a favoritos.

Lo mismo ocurría a la hora de eliminar un evento, ya que no se comprobaba que el evento no existía en el repositorio y daba error en la prueba IGIC 2.b del documento US438172-EliminarEventoDeFavoritos-TestPlan.

En el caso IGIC2.c no se comprobaba que el evento ya se encontraba en la lista de favoritos y por lo tanto no se debería poder añadir a favoritos.

Lo mismo ocurría a la hora de eliminar un evento, ya que no se comprobaba que el evento no se encontraba en la lista de favoritos y daba error en la prueba IGIC 2.c del documento US438172-EliminarEventoDeFavoritos-TestPlan.

### 5.3 Pruebas de Interfaz

No se encuentran problemas en la interfaz. Si se han encontrado problemas en la codificación de las pruebas a la hora de comprobar si la estrella del botón asociado a añadir y eliminar evento de favoritos estaba vacía o rellena. Para solucionar este error, tuvimos que añadir una etiqueta al botón cuando se pulsaba para poder compararlo en el método de prueba.

Además, encontramos problemas con el menú desplegable en la esquina superior derecha de la vista principal de la aplicación. Al intentar acceder a la vista que mostraba los eventos favoritos, no podíamos acceder porque no sabíamos cómo abrir el desplegable. Tras consultarlo con el profesor, nos envió la línea que faltaba para desplegar el menú.

Nota: Se ha introducido una espera de 1 segundo tras abrir el menú desplegable y antes de dar click a “Eventos Favoritos” porque había veces que intentaba pulsarlo antes de que se hubiese cargado, ocasionando que el test no se ejecutara correctamente.

*Plan realizado y ejecutado por Eduardo Argüelles Bueno y Javier Barquín Escalada*