



Plan de pruebas – US437801-MantenerEstadoDeFiltros (Sergio Pérez Landaburu)

Los niveles de prueba que se van a aplicar son los siguientes:

- Pruebas de aceptación: las pruebas de aceptación se definirán siguiendo una estrategia basada en historias de usuario y serán ejecutadas de forma manual.
- Pruebas de integración: estas pruebas verifican la interacción entre clases. Se llevarán a cabo usando Junit y Robolectric.
- Pruebas de interfaz: se prueban todos los elementos desde la interfaz, comprobando que la funcionalidad es correcta y que las salidas son las esperadas respecto a las entradas que se han introducido. Para ello, se ha usado el framework Espresso, que nos permite automatizar las pruebas.

PRUEBAS DE ACEPTACIÓN

Se muestran las pruebas en base a los criterios de aceptación acordados con el *Product Owner*.

Prueba 00: Éxito

1. El usuario selecciona filtrar.
2. El sistema crea una ventana emergente para filtrar por tipo.
3. El usuario ajusta los filtros.
4. El usuario selecciona aplicar.
5. Se cierra la ventana emergente.
6. El sistema muestra la lista de eventos con el filtro aplicado.
7. El usuario vuelve a seleccionar filtrar.
8. El sistema crea una ventana emergente para filtrar por tipo.
9. El sistema muestra una ventana emergente con los filtros aplicados anteriormente.

Prueba 01: Cancelar

1. El usuario selecciona filtrar.
2. El sistema crea una ventana emergente con los tipos de eventos.
3. El usuario ajusta los filtros.
4. El usuario selecciona cancelar.
5. Se cierra la ventana emergente
6. El sistema muestra la lista de eventos previa a la selección del botón filtrar.

Prueba 02: Situación anómala (evento sin tipo)

1. El usuario selecciona filtrar por un tipo de evento.
2. El sistema crea una ventana emergente con los tipos de eventos.
3. El usuario ajusta los filtros
4. El usuario selecciona aplicar.
5. Se cierra la ventana emergente
6. El sistema detecta que hay eventos sin tipo
7. El sistema obvia estos eventos a la hora de filtrar.
8. El sistema muestra la lista de eventos con el filtro aplicado.
9. El usuario selecciona filtrar para comprobar que los filtros se mantienen.



PRUEBAS DE INTEGRACIÓN

Método **combinaFiltros()** de la clase **EventsPresenter.java**

Casos de prueba válidos:

Identificador	Entrada	Resultado
IT. 1A	Lista "eventosEnDeterminadosFiltros" vacía y lista "eventosEnDeterminadasFechas" vacía.	No se actualiza nada.
IT. 2A	Lista "eventosEnDeterminadosFiltros" con eventos y lista "eventosEnDeterminadasFechas" vacía.	La lista "eventosEnFiltrosCombinados" contiene los eventos de la lista "eventosEnDeterminadosFiltros".
IT. 3A	Lista "eventosEnDeterminadosFiltros" vacía y lista "eventosEnDeterminadasFechas" con eventos.	La lista "eventosEnFiltrosCombinados" contiene los eventos de la lista "eventosEnDeterminadasFechas".
IT. 4A	Lista "eventosEnDeterminadosFiltros" con eventos y lista "eventosEnDeterminadasFechas" con eventos.	La lista "eventosEnFiltrosCombinados" contiene los eventos que aparecen a la vez en ambas listas.

PRUEBAS DE INTERFAZ DE USUARIO

Se realizarán cuatro pruebas:

A. No se selecciona ningún evento.

1. Se abrirá la interfaz de filtrar pulsando el botón "Filtrar".
2. Se comprobará que no hay ningún tipo de evento seleccionado.
3. El usuario pulsará el botón "Aplicar" de la ventana flotante.
4. Se comprobará que la lista de eventos mostrada contiene todos los eventos.



B. Se selecciona un tipo evento

1. Se abrirá la interfaz de filtrar pulsando el botón “Filtrar”.
2. Se comprobará que no hay ningún tipo de evento seleccionado.
3. Se seleccionará un tipo de evento.
4. El usuario pulsará el botón “Aplicar” de la ventana flotante.
5. Se comprobará que la lista de eventos mostrada por la aplicación corresponde únicamente a los eventos de dicho tipo.

C. Se seleccionan otros dos tipos de eventos aparte del ya seleccionado.

1. Se abrirá la interfaz de filtrar pulsando el botón “Filtrar”.
2. Se seleccionará un tipo de evento.
3. El usuario pulsará el botón “Aplicar” de la ventana flotante.
4. Se abrirá la interfaz de filtrar pulsando el botón “Filtrar”.
5. Se comprobará que se mantiene el elemento seleccionado previamente.
6. Se seleccionarán dos tipos de eventos distintos al previo.
7. El usuario pulsará el botón “Aplicar” de la ventana flotante.
8. Se comprobará que la lista de eventos muestra los eventos de los nuevos tipos seleccionados además de los del primer tipo de evento.

D. Se elimina la selección de uno de los tipos de eventos.

1. Se abrirá la interfaz de filtrar pulsando el botón “Filtrar”.
2. Se seleccionarán cuatro tipos de eventos.
3. El usuario pulsará el botón “Aplicar” de la ventana flotante.
4. Se abrirá la interfaz de filtrar pulsando el botón “Filtrar”.
5. Se comprobará que los elementos seleccionados se mantienen seleccionados.
6. Se seleccionará uno de los tipos de eventos para deseleccionarlo.
7. El usuario pulsará el botón “Aplicar” de la ventana flotante.
8. Se comprobará que la lista de eventos mostrada contiene sólo los elementos seleccionados eliminando los del tipo de evento des-seleccionado.



Informe de pruebas – US437801-MantenerEstadoDeFiltros (Juan Velez Velasco)

-PRUEBAS DE INTERFAZ:

- He implementado todos los casos A,B,C,D.
- No he tenido ningún problema.

-PRUEBAS DE INTEGRACIÓN

- He implementado todos los casos en onCombinaFiltros().