

Plan de pruebas Filtrar por Género

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

- Pruebas de aceptación. Las pruebas de aceptación se definirán basadas en la historia de usuario definida.
- Pruebas unitarias. Se utilizará la técnica de prueba de métodos, usando técnicas de caja negra (partición equivalente y AVL) para la definición de los casos de prueba de cada método de cada clase o componente. Será necesaria la utilización de Mocks.

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

PRUEBAS CAPA PRESENTACIÓN

Identificador	Entrada / Acción del usuario	Resultado esperado
A1	"Acción"	El sistema muestra las 6 películas que contienen el género acción.
A2	"Acción", "Terror"	El sistema muestra las 8 películas de ambos géneros.
A3	No se selecciona ningún género por el que filtrar	El sistema no permite pulsar el botón "Aplicar".
A4	Servicio de datos no está disponible	Mensaje "Error al cargar las películas"
A5	Pérdida de la conexión a Internet	Mensaje "Error al cargar las películas"
A6*	Selecciona "cancelar"	Regresa a la pantalla principal en la que se muestran todas las películas sin aplicarse ningún filtro.
A7*	Selecciona "NA"	Se muestra 1 película cuyo genero no existe.
A8*	El usuario pulsa fuera del desplegable del filtro de géneros	No ocurre nada

*Casos añadidos una vez recibida la corrección de los criterios de aceptación

PRUEBAS UNITARIAS

Para poder llevar a cabo estas pruebas, será necesario el uso de objetos Mock . Se aplica prueba de métodos, siendo los casos de prueba definidos para cada método los siguientes.

Los métodos para probar son:

- applyGenreFilter(List<Movie> movies)
- onFilterGenreMenuClicked()
- onGenresFiltered(List<String> selectedGenresWithCount)
- applyFilters()

Pruebas unitarias de la capa de presentación (PRESENTER)

Esta es la función implementada.

applyGenreFilter(List<Movie> movies)

Identificador	Entrada	Verificación
UTC1.a	Películas: Películas: 1(Acción), 2(Acción, Aventura), 3(Comedia) 4(Aventura) Géneros para filtrar: [Acción]	La lista devuelta contiene las películas 1 y 2.
UTC1.b	Películas: 1(Acción), 2(Acción, Aventura), 3(Comedia) 4(Aventura) Géneros para filtrar: [Acción, Aventura]	La lista devuelta contiene las películas 1 y 2 y 4
UTC1.c	Películas: 1(Acción), 2(Acción, Aventura), 3(Comedia) 4(Aventura) Géneros para filtrar: [Comedia]	Devuelve una lista vacía.
UTC1.d	Películas: 1(Acción), 2(Acción, Aventura), 3(Comedia) 4(Aventura) 5 (Null) Géneros para filtrar: [NA]	Devuelve la película 5 que no tiene género.
UTC1.e	Películas: Lista vacía. Géneros para filtrar: [Acción]	Retorna Null.

UTC.1f	Películas: 1(Acción), 2(Comedia). Géneros para filtrar: Lista vacía.	Retorna la lista de películas.
--------	--	--------------------------------

ERRORES ENCONTRADOS

Al realizar las pruebas de capa de presentación me di cuenta que faltaba de implementar la lógica que hacía que si no se selecciona ningún género no te dejase seleccionar el botón aplicar.

Con la realización de los test unitarios encontré un error que no se comprobaba de forma correcta que el género de una película fuese nulo.

REALIZACIÓN DE LAS PRUEBAS

He realizado tanto la especificación del plan de prueba como la implementación de las pruebas de capa de presentación con estos dos casos:

- `public void FiltrarGeneroExitoso()`
- `public void testFiltrarSinSeleccionarGenero()`

Y estos casos de pruebas unitarias de la capa de presentación para la función `applyGenreFilter(List<Movie> movies)`:

- `public void testFilterByOneGenre()`
- `public void testFilterByMultipleGenres()`
- `public void testFilterWithNoResults()`
- `public void testFilterWithEmptyMovieList()`
- `public void testFilterWithEmptyFilterList()`