



Plan de pruebas #506177-Ordenar por precio

PRUEBAS DE ACEPTACIÓN

En base de los criterios de aceptación se identifican los siguientes escenarios:

A1. a CA: Caso de éxito: La aplicación muestra la lista ordenada de datos correctamente bajo el criterio de orden seleccionado (ASCENDENTE).

1. El usuario pulsa el botón de ordenar en la toolbar.
2. El usuario selecciona el tipo de carburante en el menú desplegable.
3. El usuario selecciona el orden por el cual organizar los precios.
4. El usuario pulsa el botón “ordenar”.
5. La aplicación muestra una lista con las gasolineras encontradas con el carburante seleccionado y ordenadas de manera ascendente.
6. La aplicación muestra un toast con el número de gasolineras encontradas.
7. Se verifica que la lista ha sido ordenada correctamente de manera ascendente.

A1. b CA: Caso de éxito: La aplicación muestra la lista ordenada de datos correctamente bajo el criterio de orden seleccionado (DESCENDENTE).

1. El usuario pulsa el botón de ordenar en la toolbar.
2. El usuario selecciona el tipo de carburante en el menú desplegable.
3. El usuario selecciona el orden por el cual organizar los precios.
4. El usuario pulsa el botón “ordenar”.
5. La aplicación muestra una lista con las gasolineras encontradas con el carburante seleccionado y ordenadas de manera descendente.
6. La aplicación muestra un toast con el número de gasolineras encontradas.
7. Se verifica que la lista ha sido ordenada correctamente de manera descendente.



A1. c CA: Caso de éxito: La aplicación muestra la lista ordenada de datos correctamente y las gasolineras con caracteres erróneos o nulos son enviados al final de la lista de datos

1. El usuario pulsa el botón de ordenar en la toolbar.
2. El usuario selecciona el tipo de carburante en el menú desplegable.
3. El usuario selecciona el orden por el cual organizar los precios.
4. El usuario pulsa el botón “ordenar”.
5. La aplicación muestra una lista con las gasolineras encontradas con el carburante seleccionado y ordenadas de manera ascendente o descendente en función de su selección, al existir gasolineras que carecen del carburante por el que se ordena ó su valor es nulo, éstas se muestran sin orden alguno al final de la página.
6. La aplicación muestra un toast con el número de gasolineras encontradas.
7. Se verifica que la lista ha sido ordenada correctamente y que las gasolineras con caracteres erróneos se encuentran al final de la misma.

A2. CA: Caso de error: Resultado con lista vacía.

1. El usuario pulsa el botón de ordenar en la toolbar.
2. El usuario selecciona el tipo de carburante en el menú desplegable.
3. El usuario selecciona el orden por el cual organizar los precios.
4. El usuario pulsa el botón “ordenar”.
5. La aplicación no encuentra gasolineras que puedan ser ordenadas bajo el criterio seleccionado.
6. La aplicación muestra un toast con 0 gasolineras encontradas.

En el caso de éste plan de pruebas se obvia el correcto funcionamiento de la funcionalidad de filtros. Al tener valores fijos en los selectores de orden dentro de la interfaz, no es posible que éstos produzcan un fallo de entrada (las combinaciones de ambos selectores son siempre válidas). Por lo tanto el único caso de “fallo” posible es el cual parta de una lista vacía por motivos ajenos a ésta funcionalidad.



PRUEBAS UNITARIAS

Se realizarán las pruebas unitarias de los métodos *ordenarGasolinerasPorPrecio* e *init* de la clase *mainPresenter*.

Método ***ordenarGasolinerasPorPrecio(Combustible combustible, Orden orden)***

- I. Combustible y orden válidos ascendente.
- II. Combustible y orden válidos descendente.
- III. Combustible y orden válidos pero lista de gasolineras vacía.
- IV. Combustible y orden válidos pero hay gasolineras sin datos específicos.

Tabla 1. Casos de prueba de unitaria

Identificador	Entrada*	Valor esperado
UP.I	("GASOLEOA","ASCENDENTE") gasolinerasFiltradas	Se actualiza la lista de gasolineras. Se muestra una lista ordenada de manera ascendente.
UP.II	("BIODIESEL","DESCENDENTE") gasolinerasFiltradas	Se actualiza la lista de gasolineras. Se muestra una lista ordenada de manera descendente.
UP.III	("GASOLEOA","DESCENDENTE") gasolinerasFiltradas (vacía)	Se actualiza la lista de gasolineras. Se muestra una lista vacía de gasolineras.
UP.IV	("BIODIESEL","ASCENDENTE") gasolinerasFiltradas (con valores nulos o erróneos)	Se actualiza la lista de gasolineras. Se muestra la lista de gasolineras ordenadas correctamente con los valores nulos o erróneos al final.



- Método *init(View view)*

Internamente el método *init* realiza una llamada al método *load()* por lo tanto se añade el caso de que éste recibe correctamente sus parámetros (gasolineras) y el caso de error si falla la carga de las mismas.

Tabla 2. Casos de prueba unitaria

Identificador	Entrada	Valor esperado
UP2.I	<i>vista gasolineras</i>	Se cargan correctamente las gasolineras.
UP2.II	<i>vista gasolineras</i>	Error en carga de gasolineras.

Pruebas de interfaz:

- *OrdenarPorPrecioAscUITest.*
- *OrdenarPorPrecioDescUITest.*

Ambas pruebas trabajan sobre un archivo local que contiene la información de 6 gasolineras con valores de combustible diferentes, 5 de ellas tienen valores distintos y válidos que pueden ser ordenados de manera ascendente y descendente mientras que las otras 2 restantes contienen valores nulos. Las pruebas verifican el correcto orden tanto ascendente como descendente de la lista verificando su posición en la interfaz.

Resultados de las pruebas:

Pruebas unitarias:

Se comprueba con un determinado grupo de gasolineras que al realizar el orden éste es correcto y se muestran como se espera en función de su precio de combustible.

Pruebas de interfaz**:

Se comprueba el correcto funcionamiento de los componentes de orden en la aplicación.

*** Debido a una serie de errores con el entorno de desarrollo ya no se comprueba el mensaje de confirmación de carga (toast) en las pruebas de interfaz.*