



## Plan de pruebas #500238-Buscar Por Tipología Combustible

### **PRUEBAS DE ACEPTACIÓN**

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

A1. CA: Caso de éxito: Búsqueda por tipología de combustible (solo selecciona uno).

1. El usuario pulsa el botón de filtros en la toolbar.
2. El usuario selecciona un tipo de combustible en el menú desplegable.
3. El usuario pulsa el botón "Buscar".
4. La aplicación muestra una lista con las gasolineras encontradas con el tipo de gasolina seleccionado incluyendo en cada elemento de la lista el precio de este.
5. La aplicación muestra un toast con el número de gasolineras encontradas.
6. Se verifica que se muestra la lista de gasolineras de forma correcta.
7. Se verifica que el toast muestra el número correcto de gasolineras encontradas.

A2. CA: Caso de éxito: Búsqueda por tipología de combustible (selecciona más de uno).

1. El usuario pulsa el botón de filtros en la toolbar.
2. El usuario selecciona más de un tipo de combustible en el menú desplegable.
3. El usuario pulsa el botón "Buscar".
4. La aplicación muestra una lista con las gasolineras encontradas en base a los dos primeros tipos de gasolina seleccionados incluyendo en cada elemento de la lista el precio de estos para no sobrecargar la interfaz.
5. La aplicación muestra un toast con el número de gasolineras encontradas.
6. Se verifica que se muestra la lista de gasolineras de forma correcta.
7. Se verifica que el toast muestra el número correcto de gasolineras encontradas.

A3. CA: Caso de éxito: Búsqueda exitosa sin seleccionar tipología de combustible.

1. El usuario pulsa el botón de filtros en la toolbar.
2. El usuario no selecciona ningún tipo de combustible deseado.
3. El usuario pulsa el botón "Buscar".
4. La aplicación muestra una lista con todas las gasolineras.
5. La aplicación muestra un toast con el número de gasolineras encontradas.
6. Se verifica que se muestra la lista de gasolineras de forma correcta.
7. Se verifica que el toast muestra el número correcto de gasolineras encontradas.

A4. CA: Caso de éxito: Búsqueda de un combustible que previamente se ha usado para ordenar la lista.

1. El usuario pulsa el botón de ordenar la lista en la toolbar.
2. El usuario selecciona un tipo de combustible por el que desea ordenar la lista.
3. El usuario pulsa el botón "Ordenar".
4. La aplicación muestra la lista ordenada por el combustible seleccionado.
5. El usuario pulsa el botón de filtros en la toolbar.
6. El usuario selecciona tipos de combustible en donde al menos uno coincide con el que se ha seleccionado para ordenar la lista.



7. El usuario pulsa el botón “Buscar”.
8. La aplicación muestra una lista con todas las gasolineras ordenadas.
9. La aplicación muestra una toast con el número de gasolineras encontradas.
10. Se verifica que se muestra la lista de gasolineras de forma correcta.
11. Se verifica que el toast muestra el número correcto de gasolineras encontradas.

A5. CA: Caso erróneo: Búsqueda de un combustible distinto al que se ha seleccionado para ordenar la lista.

1. El usuario pulsa el botón de ordenar la lista en la toolbar.
2. El usuario selecciona un tipo de combustible por el que desea ordenar la lista.
3. El usuario pulsa el botón “Ordenar”.
4. La aplicación muestra la lista ordenada por el combustible seleccionado.
5. El usuario pulsa el botón de filtros en la toolbar.
6. El usuario selecciona tipos de combustible distintos al usado para ordenar la lista.
7. El usuario pulsa el botón “Buscar”.
8. La aplicación muestra un mensaje de error informando que no se ha podido filtrar ya que no se puede filtrar la lista por un combustible no ordenado.
9. La aplicación mantiene el estado de la lista previa al intento de búsqueda erróneo.
10. Se verifica que se muestre el mensaje de error informando sobre el problema.
11. Se verifica que se muestra la lista previa al intento de búsqueda erróneo.

A6. CA: Lista vacía

1. El usuario pulsa el botón de filtros en la toolbar.
2. El usuario selecciona cualquier opción relacionada con la tipología del combustible con la que no se puedan encontrar gasolineras disponibles.
3. El usuario pulsa el botón “Buscar”.
4. La aplicación muestra una lista vacía ya que la aplicación no encuentra ninguna gasolinera.
5. La aplicación muestra un toast con 0 gasolineras encontradas.
6. Se verifica que se muestra la lista vacía.
7. Se verifica que se muestra el toast con 0 gasolineras encontradas.

### **PRUEBAS UNITARIAS**

En esta historia de usuario se probará el método `filtrarPorCombustibles` de la clase `Filtros` y el método `onSearchStationsWithFilters` de la clase `MainPresenter`.

#### **Método `filtrarPorCombustibles`**

`List<Gasolinera> filtrarPorCombustibles(List<Gasolinera> gasolineras, List<String> combustibles)`

Las pruebas para este método se realizarán haciendo uso las gasolineras indicadas en el Anexo 1.



- A. Solo un tipo de combustible válido.
- B. Más de un tipo de combustible válido.
- C. Tipo de combustible vacío.
- D. Lista de salida de gasolineras vacía.

**Tabla 1. Casos de prueba para el método `filtrarPorCombustible` de la clase `Filtros`**

Identificador	Entrada	Valor esperado
UD.1A	(lista gasolineras, ["Gasolina 95 E5", ""])	[G1, G2, G3]
UD.1B	(lista gasolineras, ["Gasolina 95 E5", "Gasolina 95 E10", "Gasolina 98 E5"])	[G2]
UD.1C	(lista gasolineras, [""])	[G1, G2, G3, G4, G5]
UD.1D	(lista gasolineras, ["Biodiesel"])	[ ]

#### **Método `onSearchStationsWithFilters`**

Se debe tener en cuenta que se han realizado cambios en la interfaz y no se pueden introducir municipios, provincias ni compañías no válidas.

**`onSearchStationsWithFilters`**(String provincia, String municipio, String compañía, List<String> combustibles, Boolean abierto): void

El estado inicial para estas pruebas se ve reflejado en el Anexo 2.

- A. Provincia, municipio, compañía y combustibles válidos, abierto True.
- B. Provincia, compañía y combustibles vacíos, municipio válido sin resultado y abierto False.
- C. Provincia válida sin resultados, municipio vacío, compañía y combustibles válidos, abierto False.
- D. Provincia, municipio y compañía vacíos, combustibles válidos, abierto True.

**Tabla 2. Casos de prueba para el método `onSearchStationsWithFilters` de la clase `MainPresenter`**

Identificador	Entrada	Valor esperado
UD.1A	("Cantabria", "Santander", "Repsol", ["Gasolina 95 E5", "Gasolina 95 E10", "Gasolina 95 E5 Premium"], True)	Se llama al método <i>filtrarPorProvinciaYMunicipio</i> con <code>finalProvincia = Cantabria</code> y <code>finalMunicipio = Santander</code> . Se llama a <i>filtrarPorCompanhia</i> con <code>companhia = Repsol</code> y a <i>filtrarPorCombustibles</i> con <code>combustibles = ["Gasolina 95 E5", "Gasolina</code>



		95 E10", "Gasolina 95 E5 Premium"]. Además, se llama a <i>filtrarPorEstado</i> , <i>showStations</i> y <i>showLoadCorrect</i> , que usan como parámetro la lista de gasolineras.
UD.1B	("", "Santander", "", [""]), False)	Se llama al método <i>filtrarPorProvinciaYMunicipio</i> con <i>finalProvincia</i> = null y <i>finalMunicipio</i> = Santander. Se llama a <i>filtrarPorCompanhia</i> con <i>companhia</i> = null y a <i>filtrarPorCombustibles</i> con <i>combustibles</i> = null. Además, se llama a <i>filtrarPorEstado</i> , <i>showStations</i> y <i>showLoadCorrect</i> , que usan como parámetro la lista de gasolineras.
UD.1C	("Valencia", "", "Repsol", ["Gasolina 95 E5", "Gasolina 95 E10"], False)	Se llama al método <i>filtrarPorProvinciaYMunicipio</i> con <i>finalProvincia</i> = Valencia y <i>finalMunicipio</i> = null. Se llama a <i>filtrarPorCompanhia</i> con <i>companhia</i> = Repsol y a <i>filtrarPorCombustibles</i> con <i>combustibles</i> = ["Gasolina 95 E5", "Gasolina 95 E10"]. Además, se llama a <i>filtrarPorEstado</i> , <i>showStations</i> y <i>showLoadCorrect</i> , que usan como parámetro la lista de gasolineras.
UD.1D	("", "", "", ["Gasolina 95 E5"], True)	Se llama al método <i>filtrarPorProvinciaYMunicipio</i> con <i>finalProvincia</i> = null y <i>finalMunicipio</i> = null. Se llama a <i>filtrarPorCompanhia</i> con <i>companhia</i> = null y a <i>filtrarPorCombustibles</i> con <i>combustibles</i> = ["Gasolina 95 E5"]. Además, se llama a <i>filtrarPorEstado</i> , <i>showStations</i> y <i>showLoadCorrect</i> , que usan como parámetro la lista de gasolineras.

### **PRUEBAS DE INTERFAZ**

Las pruebas de interfaz coinciden con las pruebas de aceptación, sin embargo, se usarán los datos de las gasolineras de la tabla en el Anexo 2.

Tabla 3. Casos de prueba UI.

ID	Entrada	Resultado
UI.1	("", "", "", ["Gasolina 95 E5"], True)	[G1, G2, G3, G4, G5] + toast 5 gasolineras.
UI.2	("Cantabria", "Santander", "Repsol", ["Gasolina 95 E5", "Gasolina 95 E10", "Gasolina 95 E5 Premium"], True)	[G1, G2] + toast 2 gasolineras.
UI.3	("Cantabria", "", "", [""], True)	[G1, G2, G3] + toast 3 gasolineras.



UI.4	("Ascendente", "Gasolina 95 E5"); ("Cantabria", "Santander", "Repsol", ["Gasolina 95 E5", "Gasolina 95 E10", "Gasolina 95 E5 Premium"], True)	[G1, G2] ordenadas de forma ascendente + toast 2 gasolineras.
UI.5	("Ascendente", "Biodiesel"); ("Cantabria", "Santander", "Repsol", ["Gasolina 95 E5", "Gasolina 95 E10", "Gasolina 95 E5 Premium"], True)	Lista previa al intento de búsqueda + mensaje de error a la hora de ordenar por un combustible no seleccionado en los filtros.
UI.6	("Valencia", "", "Repsol", ["Gasolina 95 E5", "Gasolina 95 E10"], False)	[ ] + toast 0 gasolineras.

### **RESULTADOS FINAL:**

En este apartado se describirán los resultados de las pruebas realizadas en la historia de usuario relacionadas con la funcionalidad de filtrado de gasolineras a partir de una serie de combustibles seleccionados en la aplicación. Se han realizado pruebas unitarias y de interfaz para poder verificar el correcto funcionamiento de los métodos desarrollados para poder implementar esta funcionalidad.

Durante la ejecución de las pruebas realizadas, tanto unitarias como de interfaz, no se detectaron fallos representativos. Todas las pruebas realizadas fueron superadas con éxito.

### **Conclusión:**

La funcionalidad de búsqueda por tipos de combustibles ha sido realizada y comprobada de forma exitosa gracias a las pruebas unitarias y de interfaz. Los resultados obtenidos nos dan a entender que la aplicación se comporta tal y como se deseaba, consiguiendo que el usuario pueda aplicar de forma correcta los filtros desarrollados.

### **ANEXO 1. ESTADO INICIAL: MÉTODO filtrarPorCombustible**

Para probar el correcto funcionamiento de la implementación desarrollada para esta historia de usuario se utilizarán los datos incluidos en esta tabla.

**Tabla 4. Estado inicial: método filtrarPorCombustible.**

GASOLINERA	GASOLINAS
G1	["Gasolina 95 E5", "Gasolina 95 E5 Premium", "Gasolina 95 E10", "Gasolina 98 E5"]



G2	["Gasolina 95 E5"," Gasolina 95 E10"," Gasolina 98 E5"]
G3	["Gasolina 95 E5"," Gasolina 98 E5"," Gasolina 98 E10"]
G4	["Gasolina 98 E5"," Gasolina 98 E10"]
G5	[""]

#### ANEXO 2. ESTADO INICIAL: MÉTODO onSearchStationsWithFilters

Para probar el correcto funcionamiento de la implementación desarrollada para esta historia de usuario se utilizarán los datos incluidos en esta tabla.

**Tabla 5. Estado inicial: método onSearchStationsWithFilters.**

Gasolinera	Compañía	Provincia	Municipio	Horario	Combustibles
G1	("Repsol")	Cantabria	Santander	"L-D: 24H"	["Gasolina 95 E5"," Gasolina 95 E5 Premium", " Gasolina 95 E10", "Gasolina 98 E5"]
G2	("Repsol")	Cantabria	Santander	"L-D: 24H"	["Gasolina 95 E5"," Gasolina 95 E10"," Gasolina 98 E5"]
G3	("Cepsa")	Madrid	Las Rozas	"L-D: 24H"	["Gasolina 95 E5"," Gasolina 98 E5"," Gasolina 98 E10"]
G4	("Otros")	Cantabria	Maliaño	"L-D: 24H"	["Gasolina 98 E5"," Gasolina 98 E10"]
G5	("Repsol")	Asturias	Oviedo	"L-D: 24H"	["","",""]

*Andrés González Eguía*