**Plan de pruebas US 500955-Filtrar por tipo de combustible.**

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 según los criterios de aceptación definidos en la historia de usuario y se ejecutarán de forma manual.
* Pruebas de integración. La estrategia para la def­­inición del orden de las pruebas de integración será jerárquica. Se probará:
  + La integración entre la capa de negocio y la de persistencia. En este caso, para la definición de los casos de prueba se utilizará técnica de prueba de métodos y caja negra y se utilizará JUnit.
  + La integración entre las tres capas. En este caso, para la definición de los casos de prueba se utilizarán técnica de casos de uso y se utilizarán JUnit y Esspreso.
* Pruebas unitarias. Se utilizará técnica de prueba de métodos y 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 JUnit, Mockito y Esspreso.

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

**PRUEBAS DE ACEPTACIÓN**

En base a los criterios de aceptación se definen los siguientes escenarios (suponemos que la interfaz por construcción no permite dejar ningún campo requerido en blanco, para evitar casos de prueba referidos a información no disponible):

A1.CA: Caso de éxito con un tipo de combustible seleccionado.

1. El cliente realiza una pulsación de selección (un click) sobre la opción de filtrar en el toolbar de la aplicación.
2. La aplicación muestra una ventana emergente con las opciones de filtrado, entre ellas la del filtrado por tipo de combustible.
3. El cliente selecciona en un menú desplegable un tipo de combustible a filtrar.
4. El cliente selecciona la opción de aplicar filtros.
5. La aplicación cierra la ventana emergente de selección de los filtros.
6. Se verifica que la aplicación muestre correctamente las gasolineras filtradas.
7. Se verifica que la aplicación muestre correctamente en un toast el número de gasolineras cargadas al aplicar el filtro.

 A2.CA: Caso de éxito con varios tipos de combustible seleccionado. 

1. El cliente realiza una pulsación de selección (un click) sobre la opción de filtrar en el toolbar de la aplicación.
2. La aplicación muestra una ventana emergente con las opciones de filtrado, entre ellas la del filtrado por tipo de combustible.
3. El cliente selecciona en un menú desplegable varios tipos de combustible a filtrar.
4. El cliente selecciona la opción de aplicar filtros.
5. La aplicación cierra la ventana emergente de selección de los filtros.
6. Se verifica que la aplicación muestre correctamente las gasolineras filtradas.
7. Se verifica que la aplicación muestre correctamente en un toast el número de gasolineras cargadas al aplicar el filtro.

A3.CA: Se selecciona la opción de restablecer filtros.

Contexto: Se asume que ya se ha aplicado un filtrado por tipo de combustible previamente.

1. El cliente realiza una pulsación de selección (un click) sobre la opción de filtrar en el toolbar de la aplicación.
2. La aplicación muestra la ventana emergente con las opciones de filtrado, entre ellas la de restablecer filtros.
3. El cliente selecciona la opción de restablecer filtros.
4. Se verifica que la aplicación muestre todas las gasolineras almacenadas en el sistema.
5. Se verifica que la aplicación muestre correctamente en un toast el número de gasolineras cargadas al restablecer los filtros.

A4.CA: Se cierran las opciones de filtrado sin aplicar los filtros.

1. El cliente realiza una pulsación de selección (un click) sobre la opción de filtrar en el toolbar de la aplicación.
2. La aplicación muestra una ventana emergente con las opciones de filtrado, entre ellas la del filtrado por tipo de combustible.
3. El cliente selecciona en un menú desplegable un tipo de combustible a filtrar.
4. El cliente selecciona la opción de cerrar la ventana emergente.
5. La aplicación cierra la ventana emergente de selección de los filtros.
6. Se verifica que la aplicación muestre correctamente las gasolineras sin aplicar el filtro.

A5.CA: El usuario no selecciona ninguna opción de filtrado.

1. El cliente realiza una pulsación de selección (un click) sobre la opción de filtrar en el toolbar de la aplicación.
2. La aplicación muestra una ventana emergente con las opciones de filtrado, entre ellas la del filtrado por tipo de combustible.
3. El cliente no selecciona en un menú desplegable ningún tipo de combustible a filtrar.
4. El cliente selecciona la opción de aplicar filtros.
5. La aplicación cierra la ventana emergente de selección de los filtros.
6. Se verifica que la aplicación muestre correctamente las gasolineras aplicando el filtro predeterminado de "todos los tipos".
7. Se verifica que la aplicación muestre correctamente en un toast el número de gasolineras cargadas al aplicar el filtro.

A6.CA: No se encuentran coincidencias para el filtro aplicado.

1. El cliente realiza una pulsación de selección (un click) sobre la opción de filtrar en el toolbar de la aplicación.
2. La aplicación muestra una ventana emergente con las opciones de filtrado, entre ellas la del filtrado por tipo de combustible.
3. El cliente selecciona en un menú desplegable un tipo de combustible a filtrar.
4. El cliente selecciona la opción de aplicar filtros.
5. La aplicación cierra la ventana emergente de selección de los filtros.
6. Se verifica que la aplicación no muestre ninguna gasolinera.
7. Se verifica que la aplicación muestre en un toast que no ha encontrado ninguna coincidencia.

**Tabla 1. Casos de prueba de aceptación**

|  |  |  |
| --- | --- | --- |
| **Identificador** | **Entrada** | **Resultado** |
| A1 | Gasolina. | Lista de **X** gasolineras y toast con **X** gasolineras. |
| A2 | Gasolina, Diésel. | Lista de **X** gasolineras y Diésel y toast con X gasolineras. |
| A3 |  | Lista con todas las gasolineras. |
| A4 |  | Lista con las gasolineras que se mostraban previamente. |
| A5 |  | Lista de **X** gasolineras con todos los tipos de combustible y toast con **X** gasolineras. |
| A6 | -- | Lista vacía y toast con 0 gasolineras. |

**PRUEBAS DE UI:**

Son las mismas pruebas que las pruebas de aceptación con excepción de que los resultados obtenidos se filtran en base a un fichero JSON llamado **“”** y se automatizan a través de Espresso.

|  |  |  |
| --- | --- | --- |
| **Identificador** | **Entrada** | **Resultado** |
| A1 | Gasolina. | Lista de **X** gasolineras y toast con **X** gasolineras. |
| A2 | Gasolina, Diésel. | Lista de **X** gasolineras y Diésel y toast con X gasolineras. |
| A3 |  | Lista con todas las gasolineras. |
| A4 |  | Lista con las gasolineras que se mostraban previamente. |
| A5 |  | Lista de **X** gasolineras con todos los tipos de combustible y toast con **X** gasolineras. |
| A6 | -- | Lista vacía y toast con 0 gasolineras. |

**TODO:La enumeración de aquellos métodos que sean susceptibles de prueba unitaria o de integración a consecuencia de la implementación de la historia de usuario, bien por ser nuevos o por modificar su comportamiento.**

**TODO:De los métodos enumerados en el apartado anterior, la especificación de los casos de prueba unitarios o de integración de DOS métodos (no puede ser la misma prueba a nivel unitaria y de integración, pues los casos de prueba serían los mismos).**

**PRUEBAS UNITARIAS**

En esta historia de usuario no se modifica nada respecto al acceso y persistencia de datos, por lo tanto, las pruebas unitarias consistirán en pruebas en las clases de dominio, negocio y de presentación.

**Pruebas unitarias de las clases de dominio**

Deberían probarse los métodos de la clase Filter.

* Método setfuelTypes(List<FuelType> ) : IFilter
* Método typeFilter( g : Gasolinera ) : Boolean
* Método toFilter( g :List <Gasolinera> ) : List<Gasolinera>
* Método clear() : void
* Método getFuelTypes(): List<FuelType>
* Método toCopy(): IFilter
* Método toFilter: TODO

|  |  |  |
| --- | --- | --- |
| Identificador | Entrada | Valor esperado |
| UD2.a | Filter(1.80) Gasolinera(Nombre, 1.8) | true |
| UD2.b | Filter(1.80) Gasolinera(Nombre, 1.2) | true |
| UD2.c | Filter(1.80) Gasolinera(Nombre, 1.81) | false |
| UD2.d | Filter(1.80) Gasolinera(Nombre, NULL) | false |

* Método clear: TODO

|  |  |  |
| --- | --- | --- |
| Identificador | Entrada | Valor esperado |
| UD3.a | Filter(1.80) | Filter(null) |

**Pruebas unitarias de la capa de negocio**

Deben probarse los métodos añadidos a la clase MainPresenter:

Método public void onFiltersClicked();

* Método onFiltersClicked(): void
* Método onFiltersPopUpFuelTypesSelected(): void
* Método onFiltersPopUpFuelTypesOneSelected(int index, boolean value): void
* Método onFiltersPopUpFuelTypesAccepted(): void
* Método onFiltersPopUpCancelClicked(): void
* Método onFiltersPopUpAcceptClicked(): void
* Método onFiltersPopUpClearFiltersClicked(): void

**TODO: ¿Pruebas para dos?**

**PRUEBAS DE INTEGRACIÓN**

En lo referente a las pruebas de integración, solo se ha añadido la interfaz IFilter que contiene los métodos necesarios para llevar a cabo el filtrado por tipo. Deberemos probar, por ende, la integración entre el Presenter y la clase de dominio Filter. Para ello, será necesario el uso de objetos Mock para la interfaz IMainContract#View, probando así la interacción entre el Presenter y Filter.

Se probará el funcionamiento de:

* Método onFiltersClicked(): void
* Método onFiltersPopUpFuelTypesSelected(): void
* Método onFiltersPopUpFuelTypesOneSelected(int index, boolean value): void
* Método onFiltersPopUpFuelTypesAccepted(): void
* Método onFiltersPopUpCancelClicked(): void
* Método onFiltersPopUpAcceptClicked(): void
* Método onFiltersPopUpClearFiltersClicked(): void