

Plan de pruebas US 500954 Filtrar por marcas.

PRUEBAS UNITARIAS.

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

Pruebas unitarias de dominio.

Deberían probarse los métodos de la clase Filter.

- Método setGasBrands(gasBrands : List<String>) : IFilter
- Método toCopy() : IFilter
- Método toFilter(g : List<Gasolinera>) : List<Gasolinera>
- Método clear() : Void

Pruebas unitarias de negocio:

Deberían probarse los métodos de la clase MainPresenter para ello serán necesarias las clases de GasolinerasRepository, IFilter, Filter y un Mock de IMainContract.View.

- Método onFiltersClicked() : Void
- Método getBrandsSelections(f : IFilter)
- Método onFiltersPopUpBrandsSelected()
- Método onFiltersPopUpBrandsOneSelected(int index, boolean value)
- Método onFiltersPopUpAccepted()
- Método setFiltersPopUpValues()

Método onFiltersPopUpBrandsOneSelected(int index, boolean value):

Identificador.	Entrada.	Valor esperado.
UGIC.1a	index = 0; value = true; tempListSelection= [("Todos", false), ("Marca1", true), ("Marca2", false)]	Llama a: view.updateFiltersPopUpBrandsSelection(index, value) para desactivar todas las marcas que no sean "Todos". Para activar todos, llamará a: view.updateFiltersPopUpBrandsSelection(index, value). Se mostrarán todas las gasolineras de todas las marcas.
UGIC.1b	index = 0; value = false; tempListSelection= [("Todos", true), ("Marca1", false), ("Marca2", false)]	Llama a: view.updateFiltersPopUpBrandsSelection(index, value) porque siempre debe haber algo seleccionado. Se mostrarán todas las gasolineras de todas las marcas.

UGIC.1c	index = 1; value = true; tempListSelection= [("Todos", true), ("Marca1", false), ("Marca2", false)]	Llama a: view.updateFiltersPopUpBrandsSelection(index, value) para desactivar la casilla de todos. Llama a: view.updateFiltersPopUpBrandsSelection(index, value) para mostrar los detalles de una gasolinera específica. Se mostrarán todas las gasolineras de la marca seleccionada.
UGIC.1d	index = 2; value = true; tempListSelection= [("Todos", false), ("Marca1", true), ("Marca2", false)]	Llama a: view.updateFiltersPopUpBrandsSelection(0, true) para activar la casilla de todos. Llama a: view.updateFiltersPopUpBrandsSelection(i, false) para desmarcar todas las gasolineras marcadas. Se mostrarán todas las gasolineras de todas las marcas.
UGIC.1e	index = 0; value = false; tempListSelection= [("Todos", true), ("Marca1", true), ("Marca2", false)]	Llama a: view.updateFiltersPopUpBrandsSelection(index, value) para desactivar la casilla de todos. Se mostrarán únicamente las gasolineras de las marcas previamente marcadas.

Método onFiltersPopUpBrandsSelected():

Identificador.	Entrada.	Valor esperado.
UGIC.2a	TempList= ["Marca1"]	Llama a getBrandsSelections(tempFilter) que debe generar correctamente la lista con "Marca 1". tempListSelection= [("Todos", false), ("Marca1", true), ("Marca2", false)] Llama a view.showFiltersPopUpBrandSelector(tempList Selection) con los valores correctos.
UGIC.2b	tempList = ["Marca1", "Marca 2"]	Llama a getBrandsSelections(tempFilter) que debe generar correctamente la lista con "Todos". tempListSelection= [("Todos", true), ("Marca1", false), ("Marca2", false)] Llama a view.showFiltersPopUpBrandSelector(tempListSelection).

PRUEBAS DE INTEGRACIÓN.

Deberán probarse los métodos de la clase MainPresenter, para ello usaremos la clase GasolinerasRepository, Filter y el mockup de I MainContract.View (Se usará la opción de cargar datos mediante json en vez del servicio).

- Método onFiltersClicked()
- Método onFiltersPopUpBrandsSelected()
- Método onFiltersPopUpBrandsOneSelected(int index, boolean value)
- Método onFiltersPopUpBrandsAccepted()
- Método onFiltersPopUpAcceptClicked()
- Método onFiltersPopUpClearFiltersClicked()
- Método load ()
- Método getBrandsSelections (IFilter f)

Método onFiltersPopUpBrandsAccepted():

Identificado r.	Entrada.	Valor esperado.
IT.1a	tempListSelection = [(“Todos”, true), (”Marca1”, false), (”Marca2”, false)] tempList=[”Marca 1”]	Se deberán mostrar todas las gasolineras. Además, se deberá verificar que tempList= [”Todos”] y que se realiza una llamada a los métodos tempFilter.setBrands(Arrays.asList(BrandsEnum.values())) y a view.updateFiltersPopupTextViews(null, getStringOfSelections(tempListSelection)).
IT.1b	tempListSelection = [(“Todos”, false), (”Marca1”, true), (”Marca2”, true)] tempList=[”Todos”]	Se deberá comprobar que se muestran las gasolineras de las marcas seleccionadas. Además, se verificará que tempList poseer las marcas seleccionadas.
IT.1c	tempListSelection = [(“Todos”, false), (”Marca1”, false), (”Marca2”, false)] tempList=[”Marca 1”]	Se deberá comprobar que se muestran todas las gasolineras.
IT.1e	tempListSelection = [(“Todos”, true), (”Marca1”, false), (”Marca2”, false)] tempList=[]	Se deberá verificar que cuando la lista está vacía y se añaden todas las marcas, se muestran todas las gasolineras.

PRUEBAS DE UI.

Identificador	Entrada	Resultado
UIT.1	["REPSOL"]	Lista de gasolineras y toast con 45 gasolineras.
UIT.2	["REPSOL", "BALLENOIL"]	Lista de gasolineras y toast con 164 gasolineras.
UIT.3	Click en reestablecer filtros.	Lista con todas las gasolineras y toast con 164 gasolineras.
UIT.4	Click en cerrar ("X").	Lista con todas las gasolineras y toast con 164 gasolineras.
UIT.5	["Todas"]	Lista con todas las gasolineras y toast con 164 gasolineras.
UIT.6	["S.V.M"]	Se cierra la ventana emergente, no se muestra ninguna gasolinera y toast 0 gasolineras.

REPORTE DE LAS PRUEBAS**Pruebas unitarias: Sergio Salas**

Se han encontrado dificultades durante el desarrollo una vez solventados los problemas, las pruebas implementadas se han ejecutado satisfactoriamente sin encontrar problemas en el código.

Pruebas de integración: Sergio Salas

Las pruebas implementadas se han ejecutado satisfactoriamente sin encontrar problemas en el código.

Pruebas de UI: Miguel Monje

Durante la ejecución de las pruebas en el caso UIT.1 se ha detectado que al filtrar por marcas no se controlaba que el rotulo pudiese tener espacios en blanco, de esta manera no se encontraba una gasolinera con rotulo "REPSOL " con espacio al final, se ha solucionado este error comprobando que los rótulos contienen el nombre de la marca. Una vez solucionado el error las pruebas han pasado correctamente