



Plan de Pruebas Historia de usuario: “#505805 Buscar gasolineras abiertas”

PRUEBAS DE ACEPTACIÓN :

”Se tiene que tener en cuenta que el fallo en Base de datos a pesar de estar hablado con él no se utiliza y Ausencia de datos que es útil pero no se negoció con el Product Owner y hemos tenerla en cuenta”. En base a los casos de uso se identifican los siguientes escenarios:

A1. CA: Caso de éxito : Se muestran las gasolineras abiertas

- 1.El usuario clica en el botón de filtros.
- 2.El usuario elige que estén abiertas en el checkbox.
- 3.El usuario presiona buscar.
- 4.Se muestra la lista de gasolineras abiertas y cuántas han sido cargadas.

A2. CA: Caso de éxito : Se muestran todas las gasolineras (si no se elige el filtro , se muestran gasolineras tanto cerradas como abiertas)

- 1.El usuario clica en el botón de filtros.
- 2.El usuario no selecciona nada en el checkbox.
- 3.El usuario presiona buscar.
- 4.Se muestran la lista de gasolineras al completo y cuántas han sido cargadas.

A3. CA: Caso de error: Se muestra una lista vacía

- 1.El usuario clica en el botón de filtros.
- 2.El usuario elige que estén abiertas en el checkbox.
- 3.El usuario presiona buscar.
- 4.Ninguna gasolinera está abierta/cerrada



5. Se muestra un mensaje diciendo que ninguna gasolinera ha sido cargada

A4. CA: Caso de error: Ausencia de los datos

1. El usuario clicca en el botón de filtros.
2. El usuario elige que estén abiertas en el checkbox.
3. El usuario presiona buscar.
4. Algunas gasolineras no disponen de los datos necesarios a la hora de cargarlo.
5. Se muestra un mensaje diciendo las gasolinera que han sido cargadas (las que sí tienen los datos si se cargan).

Tabla 1. Casos de prueba de aceptación

Identificador	Entrada	Resultado
A1	boolean(true)	Mostrar Lista de Gasolineras abiertas
A2	boolean(false)	Mostrar Lista de Gasolineras (todas)
A3	boolean(true/false)	Mensaje Lista está vacía
A4	boolean(true/false)	Mensaje Lista Gasolineras



Samuel Carrera Garmendia Plan de Pruebas



Proyecto Integrado 4º Grado en Ing. Informática

Facultad de Ciencias Universidad de Cantabria

PRUEBAS UNITARIAS :

Lista de métodos posibles de pruebas unitarias :

Clase Horario :

- obtenerHorario (String horario) : String[]
- letraDiaActual () : String
- horaActual() : int
- minutoActual() : int
- estaAbierto(String horario) : String
- compruebaHorario (String horario) : boolean
- estaEnFranjaDia(String horario): boolean

Clase Filtros :

- filtrarPorEstado(List<Gasolinera> gasolineras): List<Gasolinera>

Clase Gasolinera :

- compruebaEstado(String horario) : String

Clase MainPresenter :

- onSearchStationsWithFilters(String provincia,String municipio,boolean abierto) : void

MÉTODOS ELEGIDOS : compruebaHorario y onSearchStationsWithFilters



Pruebas unitarias de las clases de dominio :

Método compruebaHorario (String horario) : boolean

Están contemplados los distintos formatos y casos de horas y minutos

Tabla 2. Casos de prueba para el método compruebaHorario de la clase Horario

Identificador	Entrada	Valor esperado
UD.1a	("L-S: 07:00-22:30 ; D: 08:00-14:00") //Resultado horaActual : 8 //ResultadoMinutoActual : 0 //Resultado diaActual : "L"	True
UD.1b	("L-S: 07:00-22:30 ; D: 08:00-14:00") //Resultado horaActual : 7 //ResultadoMinutoActual : 1 //Resultado diaActual : "L"	True
UD.1c	("L-S: 07:00-22:30 ; D: 08:00-14:00") //Resultado horaActual : 22 //ResultadoMinutoActual : 10 //Resultado diaActual : "L"	True
UD.1d	("L-S: 07:00-14:30 y 16:00-22:00") //Resultado horaActual : 8 //ResultadoMinutoActual : 0 //Resultado diaActual : "L"	True



UD.1e	("L-S: 07:00-14:30 y 16:00-22:00") //Resultado horaActual : 7 //ResultadoMinutoActual :1 //Resultado diaActual : "L"	True
UD.1f	("L-S: 07:00-14:30 y 16:00-22:00") //Resultado horaActual : 14 //ResultadoMinutoActual : 10 //Resultado diaActual : "L"	True
UD.1g	"L-D: 24H" //Resultado horaActual : 8 //ResultadoMinutoActual : 0 //Resultado diaActual : "L"	True
UD.1h	("L-S: 07:00-22:00 ; D: 08:00-14:00") //Resultado horaActual : 6 //ResultadoMinutoActual : 0 //Resultado diaActual : "L"	False
UD.1i	()	False
UD.1j	(null)	Se lanza la excepción <i>IllegalArgumentExcepcion</i> en la función <i>obtenerHorario</i> (String horario).



Método `onSearchStationsWithFilters(String provincia,String municipio,boolean abierto)`
: void

A.Las gasolineras abiertas.

B.Todas las gasolineras.

Tabla 3. Casos de prueba para el método `onSearchStationsWithFilters(String provincia,String municipio,boolean abierto)` de la clase `MainPresenter`

Identificador	Entrada	Valor esperado
UD.1a	“_” “” True	Se comprueba que se llama a <code>showStations()</code> y <code>showLoadCorrect(2)</code> . Implica por ello que llamen al método <code>filtrarPorEstado(List<Gasolinera> gasolineras)</code>
UD.1b	“_” “” False	Se comprueba que se se llama a <code>showStations()</code> y <code>showLoadCorrect(4)</code> .

Estas pruebas se realizarán entorno al anexo número 1.

PRUEBAS DE INTERFAZ :



En cuanto a las pruebas de interfaz , probaremos los siguientes casos definidos en las diferentes pruebas de aceptación (el estado inicial en el que se realizarán las pruebas corresponde al anexo 1 para las dos primeras pruebas que se creará nada más empezar la prueba,y para la tercera se utilizará el anexo 2 y cuarta el anexo 3).

Tabla 4. Casos de prueba para la interfaz

Identificador	Entrada	Resultado
A1	boolean(true) Horario todos los días(L-D)	Repsol , Carrefour + toast("Cargadas 2 gasolineras")
A2	boolean(false) Horario todos los días(L-D)	Repsol , Carrefour, Shell, Petronor + toast("Cargadas 4 gasolineras")
A3	boolean(true) Horario todos los días(L-D)	toast("Cargadas 0 gasolineras")
A4	boolean(true) Horario todos los días(L-D)	Repsol , Carrefour + toast("Cargadas 2 gasolineras")



Samuel Carrera Garmendia Plan de Pruebas

Proyecto Integrado 4º Grado en Ing. Informática

Facultad de Ciencias Universidad de Cantabria



Tanto el responsable de generar el plan de pruebas como de realizar las mismas es Samuel Carrera Garmendia.

En cuanto a la ejecución de las pruebas , me he podido dar cuenta de fallos en la clase Horario. A medida que comprobaba el primer método elegido (comprueba Horario()) me di cuenta que la mitad de los atributos era totalmente redundante para la correcta ejecución de la función. A su vez , el hecho de que aunque el horario esté vacío , seguir mostrando el resto de gasolineras que si lo tienen no estaba contemplado , ya que si el horario estaba vacío lanzaba directamente la excepción.

En general , estos han sido los errores que se han podido ver a medida que desarrollaba las diferentes pruebas del código. De esta forma , todos los test/pruebas generadas se completaron correctamente , dando el resultado esperado.



ANEXO 1 : Estado Inicial

Este archivo contiene el estado inicial para las pruebas de interfaz (las gasolineras se crearán antes de iniciar el test a partir de la horaActual, para que el test se pueda ejecutar siempre) :

Tabla 1. Estado inicial

Gasolinera	Horario
Repsol	"L-D: 24H"
Carrefour	"L-D: 24H"
Shell	horario creado a partir de la hora actual(cerrada)
Petronor	horario creado a partir de la hora actual(cerrada)



ANEXO 2 : Estado Inicial Prueba A3

Este archivo contiene el estado inicial para las pruebas de interfaz (las gasolineras se crearán antes de iniciar el test a partir de la horaActual, para que el test se pueda ejecutar siempre) :

Tabla 1. Estado inicial

Gasolinera	Horario
Repsol	horario creado a partir de la hora actual(cerrada)
Carrefour	horario creado a partir de la hora actual(cerrada)
Shell	horario creado a partir de la hora actual(cerrada)
Petronor	horario creado a partir de la hora actual(cerrada)



ANEXO 3 : Estado Inicial Prueba A4

Este archivo contiene el estado inicial para las pruebas de interfaz (las gasolineras se crearán antes de iniciar el test a partir de la horaActual, para que el test se pueda ejecutar siempre) :

Tabla 1. Estado inicial

Gasolinera	Horario
Repsol	"L-D: 24H"
Carrefour	"L-D: 24H"
Shell	horario creado a partir de la hora actual(cerrada)
Petronor	horario creado a partir de la hora actual(cerrada)
BP	sin horario