

## Grupo 4. PineApple.

### PLAN DE PRUEBAS DE LA HISTORIA DE USUARIO “SELECCIONAR VEHÍCULO”

#### AUTORES

- Definición de las pruebas de aceptación y del Plan de Pruebas: Daniel Llovió y Víctor Pérez.
- Definición de las pruebas de interfaz: Daniel Llovió y Víctor Pérez.
- Implementación de las pruebas de interfaz: Daniel Llovió y Víctor Pérez.
- Definición de las pruebas unitarias: Eduardo Barañano y Roberto Matas
- Definición de las pruebas de integración: Eduardo Barañano y Roberto Matas
- Implementación de las pruebas unitarias: Eduardo Barañano y Roberto Matas
- Implementación de las pruebas de integración: Eduardo Barañano y Roberto Matas
- Redacción del informe de pruebas: Eduardo Barañano, Roberto Matas, Víctor Pérez y Daniel Llovio

#### ÍNDICE

- Introducción
- Pruebas de Aceptación
- Pruebas Unitarias
- Pruebas de Integración
- Pruebas de Interfaz

#### INTRODUCCIÓN

Mediante el presente documento se va a definir el plan de pruebas multinivel para la historia de usuario “Seleccionar Vehículo” de la aplicación AppGasolineras. Los niveles de prueba que se van a aplicar son los siguientes:

- **Pruebas de aceptación:** En base a los casos de uso especificados mediante la reunión con el Product Owner, se han identificado unos escenarios que explicaremos en su apartado correspondiente.
- **Pruebas unitarias:** Se utilizará la técnica de prueba de métodos, usando técnicas de caja negra para la definición de los casos de prueba de cada método. Será necesaria la utilización de Junit.

- **Pruebas de integración:** Se utilizará JUnit y Espresso para implementar las pruebas de integración.

- **Pruebas de interfaz:** Se ha utilizado JUnit y Espresso para implementar las pruebas de interfaz.

## **PRUEBAS DE ACEPTACIÓN**

Las pruebas de aceptación que se realizarán a la hora de que el Product Owner decida evaluar el funcionamiento real de la aplicación son las que a continuación se especifican:

1. En este caso esta lista de vehículos estará vacía. El Product Owner procederá a añadir un vehículo pulsando en el botón de “Añadir vehículo” y rellenará todos los datos solicitados en el formulario que le aparecerá a continuación. Una vez rellenados los datos correctamente, el Product Owner aceptará estos datos y el vehículo se añadirá. En ese momento, en el vehículo aparecerá un icono de forma circular que tendrá un color verde e indicará que el vehículo está seleccionado. Debemos comprobar que el vehículo que se ha añadido a la lista vacía de vehículos se selecciona automáticamente poniéndose el icono circular en verde.
2. En este caso la lista de vehículos tendrá vehículos. El Product Owner procederá a añadir un vehículo pulsando en el botón de “Añadir vehículo” y rellenará todos los datos solicitados en el formulario que le aparecerá a continuación. Una vez rellenados los datos correctamente, el Product Owner aceptará estos datos y el vehículo se añadirá. En ese momento, en el vehículo aparecerá un icono de forma circular que tendrá un color rojo e indicará que el vehículo no se ha seleccionado. Debemos comprobar que el vehículo que se ha añadido a la lista de vehículos no se selecciona automáticamente manteniéndose el icono circular en color rojo y que el vehículo seleccionado con anterioridad tiene el icono en verde.
3. El Product Owner seleccionará uno de los vehículos de la lista. Posteriormente volverá al menú principal donde aparecen la lista de gasolineras y comprobará que en el spinner del tipo de combustible se ha seleccionado el combustible del vehículo escogido. Debemos comprobar que al seleccionar un vehículo y volver al menú de gasolineras, el tipo de combustible que aparece en el spinner es el combustible del vehículo seleccionado.
4. El Product Owner seleccionará uno de los vehículos de la lista. Posteriormente cerrará la aplicación y volverá a entrar. Una vez dentro de la aplicación verá que en la lista de gasolineras se mantiene el combustible del vehículo seleccionado en la anterior sesión. El Product Owner entrará de nuevo en la lista de vehículos y comprobará que el vehículo seleccionado anteriormente sigue seleccionado. Debemos comprobar que al seleccionar un vehículo y salir de la aplicación de forma no esperada y volver a entrar, el vehículo que aparece como seleccionado sea el que se seleccionó antes de salir de la aplicación. Además, se deberá comprobar que el tipo de combustible que aparece en el spinner en la lista principal de gasolineras es el mismo que el vehículo seleccionado.

## **PRUEBAS UNITARIAS**

### **Método readGasolinera()**

Los casos de prueba para el método readGasolinera() de la clase ParserJSONGasolineras serán los mostrados a continuación: (gasolineraOk es la descripción en JSON de una gasolinera cualquiera de forma que permita al método identificar sus componentes y transformarla en un objeto de la clase Gasolinera válido y con la asignación de sus atributos correcta, mientras que gasolineraNotOk es la descripción en JSON de una gasolinera que no será parseable por el método y que en consecuencia no podrá retornar un objeto de Gasolinera con el formato y la asignación de valores esperada)

#### **Casos de prueba**

<b>Identificador</b>	<b>Entrada</b>	<b>Valor esperado</b>
<b>UT.1a</b>	(gasolineraOk)	new Gasolinera (Retorna una gasolinera con los siguientes valores en sus atributos: rotulo = CEPSA, localidad = LAREDO, provincia = CANTABRIA, id = 77777, gasoleoA = 0.996, sinplomo95 = 0.912 y direccion = Calle Inventada 1, y asignados a la misma correctamente)
<b>UT.1b</b>	(gasolineraNotOk)	(?) (se espera que no se pueda crear una gasolinera ya que no va a encontrar la forma de hacerlo)

*Tabla 1. Casos de prueba para el método readGasolinera() de la clase ParserJSONGasolineras*

### **Método readArrayGasolineras()**

Los casos de prueba para el método readArrayGasolineras() de la clase ParserJSONGasolineras serán los mostrados a continuación: (listaGasolinerasOk es la descripción en JSON de varias gasolineras cualquiera de forma que permita al método identificar sus componentes y transformarla en una lista con objetos de la clase Gasolinera que sea válida y con la asignación de sus atributos correcta, mientras que listaGasolinerasNotOk es la descripción en JSON de varias gasolinera que no serán parseables por el método y que en consecuencia no podrá retornar una lista con de Objetos de la clase Gasolinera con el formato y la asignación de valores esperada)

#### **Casos de prueba**

<b>UT.1c</b>	(listaGasolinerasOk)	new ArrayList<Gasolinera> (Retorna una lista de dos gasolineras con los siguientes valores en sus atributos, la primera: rotulo = REPSOL, localidad = COLINDRES,
--------------	----------------------	--

		provincia = CANTABRIA, id = 55555, gasoleoA = 1.096, sinplomo95 = 1.012 y direccion = Avenida Real 1, la segunda rotulo = AVIA, localidad = LIMPIAS, provincia = CANTABRIA, id = 66666, gasoleoA = 0.896, sinplomo95 = 0.812 y direccion = Avenida Real 2, asignados correctamente)
UT.1d	(listaGasolinerasNotOk)	(?) (se espera que no se pueda crear una lista de gasolineras ya que no va a encontrar la forma de hacerlo)

*Tabla 2. Casos de prueba para el método readArrayGasolineras() de la clase ParserJSONGasolineras*

## **PRUEBAS DE INTEGRACIÓN**

Para la realización de las pruebas de integración para la implementación de seleccionarVehiculo() hay que fijarse en que a la hora de seleccionar un vehículo para su uso, los datos de este se escriben en un fichero aparte y será necesario acceder a él para poder llevar a cabo las comprobaciones pertinentes. En este caso nuestro interés girará en torno a la aparición y desaparición de los datos de un determinado vehículo en el ya mencionado fichero. Por tanto, el objetivo final de nuestros test de integración será probar dos casos, el primero que al escoger un determinado vehículo este esté seleccionado realmente, es decir, que al acceder al fichero de vehículos seleccionados (donde, recordamos, solo puede haber un vehículo recogido a la vez) los datos que aparezcan sean los del vehículo en cuestión, y, por otra parte, que al deseleccionar dicho vehículo se acceda al fichero y este no contenga sus datos (esto es, que los datos que aparezcan sean los de otro vehículo porque se ha cambiado la selección o ninguno porque no hay ningún vehículo seleccionado en ese momento). Los casos de prueba para el método seleccionarVehiculo() de la clase VehiclesActivity son los siguientes (se parte sin tener ningún vehículo seleccionado y un vehículo añadido de matrícula 1111BBB):

### **Casos de prueba válidos**

Identificador	Entrada	Valor esperado
IT.1	Seleccionar vehículo con marca Seat, modelo Ibiza, matrícula 1111BBB y combustible Gasolina95 de la lista de vehículos.	Los datos del vehículo con marca Seat, modelo Ibiza, matrícula 1111BBB y combustible Gasolina95 se encuentran escritos en el fichero
IT.2	De-seleccionar vehículo con marca Seat, modelo Ibiza, matrícula 1111BBB y combustible Gasolina95 seleccionado de la lista de vehículos.	Los datos del vehículo con marca Seat, modelo Ibiza, matrícula 1111BBB y combustible Gasolina95 ya no aparecen en el fichero (está vacío o no existe)

## **PRUEBAS DE INTERFAZ**

Los casos a que se han implementado en la clase SeleccionarVehiculoUITest son los definidos en la tabla que se muestra a continuación. Para establecer dichos casos, se han tenido como referencia los indicados por el Product Owner en las pruebas de aceptación, de manera que se puedan comprobar las diferentes alternativas que se puedan dar a la hora de seleccionar un vehículo en la aplicación.

<b>Identificador</b>	<b>Entrada</b>	<b>Valor esperado</b>
<b>UIT.1</b>	Se pulsa el botón Añadir Vehículo y se introduce los siguientes valores, Nombre: Seat, Modelo: León, Combustible: gasóleo, Matrícula: 1235TPL. A continuación, pulsa el botón aceptar.	El vehículo se muestra a continuación en la lista de vehículos con el icono circular en verde.
<b>UIT.2</b>	Se pulsa el botón Añadir Vehículo y se introduce los siguientes valores, Nombre: Ford, Modelo: Focus, Combustible: gasolina, Matrícula: 1111PPP. A continuación, pulsa el botón aceptar.	El vehículo se muestra a continuación en la lista de vehículos con el icono circular en rojo.
<b>UIT.3</b>	Se selecciona el vehículo Ford Focus, se selecciona el botón para volver al menú principal.	En el spinner del tipo de combustible debe aparecer el combustible del vehículo seleccionado.
<b>UIT.4</b>	Se selecciona el vehículo Ford Focus y se cierra la aplicación. Se vuelve a entrar en la aplicación.	El vehículo seleccionado sigue siendo el Ford Focus.