

Plan de pruebas: historia de usuario

Listar gasolineras por precio - 399307

Estructura del plan de pruebas:

Pruebas de aceptación. Las pruebas de aceptación se definirán siguiendo una estrategia basada en historias de usuario, específicamente en sus criterios de aceptación, y se ejecutarán de forma manual.

Pruebas de interfaz de usuario (UI Test). En estas se prueban todos los componentes de la aplicación de manera conjunta. Se utilizará el framework Espresso, podría ser necesario el uso de Mocks para poder trabajar sobre un conjunto de datos conocidos ya que la fuente de datos de la que se depende es dinámica.

Pruebas de integración. Las pruebas de integración se definirán usando técnicas de métodos y caja negra. Se usará la librería JUnit y la librería Mockito si es necesaria.

Pruebas unitarias. Verifican el comportamiento de clases de manera aislada, por lo que si una de estas clases depende de otra, se necesitará el uso de la librería Mockito, por supuesto se utilizará el framework JUnit.

Pruebas de aceptación (Listar gasolineras por precio):

Para constituir un caso de éxito se deben de contemplar la prueba A1.a y A1.b como una sola.

La nomenclatura para las pruebas de aceptación es "A" seguido del número de historia de usuario, que al ser una sería "1", y seguido de "<letra>" en orden alfabético que corresponda al caso de prueba.

Casos de prueba de aceptación:

A1.a : Éxito - ordena gasolineras correctamente por precio

1. El usuario selecciona la aplicación de gasolineras de entre las diversas aplicaciones disponibles en su dispositivo móvil.

2. El sistema muestra por orden descendente en la pantalla: una barra de herramientas en la que situada a su izquierda se encuentra el logo de la aplicación, en la parte central el nombre de la aplicación y en la parte derecha un botón de opciones; bajo la barra de herramientas se muestra en una franja del ancho de la pantalla "Cantabria, <nombre de combustible>, "Mi casa" donde <nombre de combustible> se sustituirá por el tipo de combustible por el que se ordenarán (según su precio) las diferentes gasolineras, será el subtipo más común de diésel, en su defecto será gasolina 95; por último se mostrará un listado con las gasolineras ordenadas por precio, de menor a mayor. En cada línea del listado aparece de izquierda a derecha, la dirección de la gasolinera, el precio del combustible y la distancia que se encuentra la gasolinera de un punto, (este último valor será ficticio de momento).

3. Se verifica que el sistema muestra el listado de gasolineras correctamente ordenado por el precio del combustible indicado, en un orden de menor a mayor (ascendente).

A1.b : Éxito - la distancia desde un punto a la gasolinera se visualiza correctamente

1. El usuario selecciona la aplicación de gasolineras de entre las diversas aplicaciones disponibles en su dispositivo móvil.

2. El sistema muestra por orden descendente en la pantalla: una barra de herramientas en la que situada a su izquierda se encuentra el logo de la aplicación, en la parte central el nombre de la aplicación y en la parte derecha un botón de opciones; bajo la barra de herramientas se muestra en una franja del ancho de la pantalla "Cantabria, <nombre de combustible>, "Mi casa" donde <nombre de combustible> se sustituirá por el tipo de combustible por el que se ordenarán (según su precio) las diferentes gasolineras, será el subtipo más común de diésel, en su defecto será gasolina 95; por último se mostrará un listado con las gasolineras ordenadas por precio, de menor a mayor. En cada línea del listado aparece de izquierda a derecha, la dirección de la gasolinera, el precio del combustible y la distancia que se encuentra la gasolinera de un punto, (este último valor será ficticio de momento).

3. Se verifica que el sistema muestra correctamente para cada gasolinera de la lista la información correspondiente a la distancia desde un punto a una gasolinera, se entiende por correcto que no se pierda información con valores de gran longitud y que la forma en la que se muestra sea coherente. Para esto se usan distancias de longitudes variables y de unidades variables, siendo las predilectas el metro (m), y el kilómetro (km).

A1.c : imposibilidad de traer información de la API, no hay conexión a internet.

1. El usuario selecciona la aplicación de gasolineras de entre las diversas aplicaciones disponibles en su dispositivo móvil.
2. El sistema muestra un mensaje informando de que no se puede acceder a la API porque no hay conexión a internet.
3. El usuario pulsa aceptar.
4. El sistema se cierra.
5. Se verifica que se ha mostrado el mensaje informativo en relación a la imposibilidad de acceder a la información de la API por no disponer de conexión a internet, y que al seleccionar el botón aceptar se cierra la aplicación.

Pruebas de interfaz de usuario:

Usando la lista de gasolineras de la clase PresenterGasolineras como entrada de datos puedo hacer las comprobaciones pertinentes.

La nomenclatura es “I” por ser pruebas de interfaz seguidas de las iniciales del escenario a probar (iniciales de la historia de usuario), y “.<letra>” en el orden alfabético que corresponda al caso de prueba.

Las pruebas se hacen sobre la Activity MainActivity y se verifica:

Casos de prueba de interfaz de usuario:

Identificador: ILGPP.a

Entrada: Listado de gasolineras de la API.

Salida esperada: Listado de gasolineras ordenado por el atributo gasoleoB de la clase gasolinera.

Identificador: ILGPP.b

Entrada: Listado de gasolineras de la API.

Salida esperada: Listado de gasolineras con TextView de distancia en cada gasolinera con diferentes longitudes y unidades.

Identificador: ILGPP.c

Entrada: IOException

Salida esperada: AlertDialog con

título: "Error, no se ha podido cargar la información de la API"

mensaje de error: "Debido a que no tiene conexión a internet, ya sea a través de wifi o datos móviles, debemos cerrar la aplicación ya que no tenemos acceso a la información necesaria de la API."

botón con texto: "Aceptar"

Al seleccionar el botón aceptar se debe cerrar la aplicación.

Pruebas de integración:

La nomenclatura sería "I" de integración + Inicial de una de la clases + Inicial de otra de las clases + "<número>" donde este se sustituye por el número de método del total a indicar en esa clase a probar + "<letra>" en orden alfabético del caso de prueba que se va a especificar.

Pruebas de unitarias:

La nomenclatura sería "U" de unitaria + Inicial o iniciales de la clase de la que se hace la prueba + "<número>" donde este se sustituye por el número de método del total a indicar en esa clase a probar + "<letra>" en orden alfabético del caso de prueba que se va a especificar.

Los métodos involucrados en la historia y con una lógica que probar son:

Clase Gasolinera → int compareTo(Gasolinera o) .

Clase ParserJSONGasolineras → Gasolinera readGasolinera (JsonReader reader) throws IOException .

Casos de pruebas unitarias:

Clase Gasolinera → int compareTo(Gasolinera o)

Identificador	Valores de entrada	Valor esperado
UG.1a	Gasolinera gasolinera → gasoleoB igual al de la gasolinera con que se compara.	0
UG.1b	Gasolinera gasolinera → gasoleoB mayor al de la gasolinera con que se compara.	-1
UG.1c	Gasolinera gasolinera → gasoleoB menor al de la gasolinera con que se compara.	1
UG.1d	null	NullPointerException

Clase ParserJSONGasolinas → Gasolinera readGasolinera (JsonReader reader) **throws** IOException

Identificador	Valores de entrada	Valor esperado
UPJG.1a	JsonReader reader (de un JSON esperado para una gasolinera)	Gasolinera con atributos del JSON
UPJG.1b	JsonReader reader (de un JSON que no corresponde a una gasolinera)	Gasolinera con atributos inesperados
UPJG.1c	null	NullPointerException