

PLAN DE PRUEBAS “Ver Favoritos”

Pruebas de Aceptación

Las pruebas de interfaz son los mismos casos que los criterios de confirmación tratados con el Product Owner y definidos en Scrumdesk.

Prueba 00: Éxito

1. El usuario selecciona ver favoritos.
2. Se verifica que el sistema muestra al usuario todos los puntos de carga añadidos a favoritos.

Prueba 01: No existe ningún punto de carga en favoritos.

1. El usuario selecciona ver favoritos.
2. Se verifica que el sistema muestre un mensaje alertando de que no existe ningún punto de carga añadido a favoritos.

Prueba 02: Fallo al acceder al fichero de persistencia al ver favoritos.

1. El usuario selecciona ver favoritos.
2. Se verifica que el sistema muestre un mensaje alertando de que se ha producido un fallo al acceder a los datos.

Pruebas de Interfaz

Voy a trabajar sobre el conjunto de cargadores del archivo chargers_es_100.json, y las pruebas se realizarán utilizando Espresso.

En este caso, tenemos 2 historias de usuario que trabajan conjuntamente, para llevar a cabo las pruebas de interfaz de ver favoritos previamente hemos tenido que añadir favoritos (ya que sino la lista estaría siempre vacía). Por lo que estas pruebas de interfaz se harán en el mismo método que las pruebas de interfaz de añadir favoritos.

Tabla 1. Casos de prueba de interfaz

Identificador	Entrada	Resultado
Prueba 00	Elige “Ver Favoritos”	Lista de X puntos de carga, que son los que están categorizados como favoritos.

Métodos Susceptibles a Pruebas

Clase Charger.java:

Clase MainPresenter.java:

- onMenuFavoritosClicked -> método que muestra la lista de los puntos de carga añadidos a favoritos.

Clase MainView.java:

- `onOptionsItemSelected` -> Se añade la opción de ver favoritos
- `getFavoriteChargers` -> método que devuelve una lista con los puntos de carga que están en favoritos.

Pruebas Unitarias

Para las pruebas unitarias usaré Junit y Mockito.

Clase `MainPresenter.java`

Método `onMenuFavoritosClicked()`

Para probar este método necesito usar un mock de `MainView`.

Encuentro estos casos a probar:

1. Ver favoritos con tres cargadores en favoritos
2. Ver favoritos con un cargador en favoritos
3. Ver favoritos sin cargadores en favoritos

Tabla 2. Casos de prueba unitarias de `VerFavoritos`

Entrada	Salida
Puntos de carga: A, B, C	[A, B, C]
Puntos de carga: A	[A]
Puntos de carga:	[]

Clase `Charger.java`

Método `getFavoriteChargers ()`

En el desarrollo de esta historia de usuario estos son los dos únicos métodos que se han creado, por lo que no hay otra opción para realizar las pruebas unitarias.

Como este método está en la vista `MainView`, no se realizarán pruebas unitarias sobre el mismo.

INFORME FINAL

El plan de pruebas ha sido realizado por Jesús, las pruebas de interfaz han sido realizadas por Jorge, ya que ha probado conjuntamente añadir y ver favoritos. Finalmente, la ejecución de las pruebas unitarias ha sido realizada por Laura.