

Descripción de los casos de uso:

Caso de uso: Mostrar paradas

Actor: Usuario

Secuencia de pasos:

1. El usuario pulsa sobre una línea
2. El sistema carga los datos
3. El sistema muestra los datos

Plan de pruebas a seguir:

Aceptación y de sistema

En base al caso de uso identificamos el siguiente caso:

AS1.CU: Mostrar paradas

- a. Selección válida (Mostrar paradas).

Casos de prueba:

AS1.a Usuario selecciona la línea 7C2 → se muestran todas las paradas correctamente.

Pruebas unitarias

Para probar los módulos de forma independiente, se realizará con pruebas de métodos (sólo los métodos públicos).

Por inspección se ha comprobado que los métodos de la clase Parada (getters y setters), funcionan correctamente.

Para probar de forma aislada la clase ParserJSON, se utilizará un fichero JSON local y conocido que contenga paradas, de forma que no se tenga que acceder a datos externos. Los casos de prueba de los métodos abarcarán los valores límite e intermedios para cada clase de equivalencia.

Para el método 'public static List<Parada> readArrayParadasBus (InputStream in)':

Casilla a comprobar	Valor esperado
lista[0]	Parada 42063
lista[11]	Parada 364
lista[442]	Parada 100

Por inspección se ha comprobado que el método `getBufferedData()` de la clase `RemoteFetch` funciona correctamente.

Para probar el método `'public void getJSON (String urlJSON)'`:

- Se ejecuta dicho método utilizando una url de un JSON conocido (p.ej: `http://datos.santander.es/api/rest/datasets/paradas_bus.json`).
- Se ejecuta el método `getBufferedData()`, que ya está probado que funciona, y se comprueba que los datos del `BufferedInputStream` que devuelve coinciden con los datos conocidos de la url que contiene el JSON.

Por inspección se ha comprobado que el método `getListaParadasBus()` de la clase `ListParadasPresenter` funciona correctamente.

Para probar el método `'public boolean obtenParadas()'`:

- Se ejecuta dicho método utilizando una url de un JSON conocido (p.ej: `http://datos.santander.es/api/rest/datasets/paradas_bus.json`).
- Se comprueba que la lista `listaParadasBus` es del mismo tamaño (443) que el número de paradas de bus del JSON y que los valores límite y uno intermedio son los esperados:

Casilla a comprobar	Valor esperado
<code>lista[0]</code>	Parada 42063
<code>lista[11]</code>	Parada 364
<code>lista[442]</code>	Parada 100

Pruebas de integración:

I1.a Usuario selecciona la línea 7C2 → Se accede a los JSON de TUS correspondientes y se muestran las paradas de la línea indicada.