Pruebas unitarias.

Las pruebas que realizaremos sobre la base de datos implementada será sobre la clase DatabaseHelper en la cual se realizaran las operaciones de guardado y obtención relacionadas con las líneas de buses, las paradas de buses y sobre el color relacionado a cada línea.

Se probara el método creaColor(Color) con los siguientes datos:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Alpha | Red | Green | Blue | Resultado |
| C1 | 23 | 245 | 12 | 24 | Su Id |
| C2 | 249 | 23 | 123 | 98 | Su Id |
| C3 | -1 | -1 | 256 | -1 | -1 |
| C4 | 2 | 123 | 123 | 123 | Su Id |

Luego se probara a obtener los colores, para el método getColor(long color\_id) se probara con los resultados de la parte anterior como parámetro de este método. Para el método gelAllColor() se comprobara que la lista que retorna contiene los colores del apartado anterior.

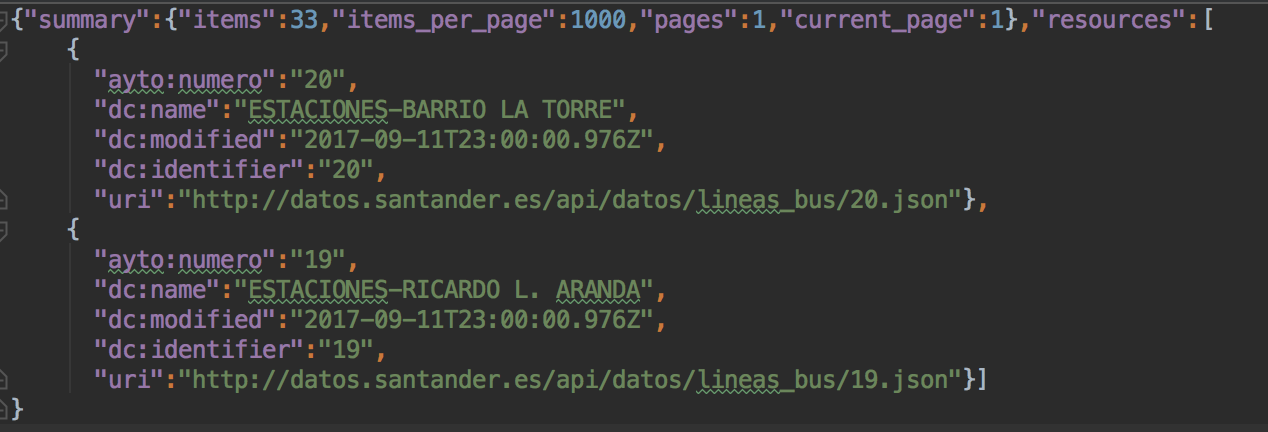
A continuación probaremos las líneas, se ingresaran los datos

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Numero | Nombre | Identificador | Color | Resultado |
| L1 | 1 | uno | 1 | C1 | Su Id |
| L2 | 2 | dos | 2 | C2 | Su Id |
| L3 | 3 | tres | 234 | C1 | Su Id |
| L4 | 4 | cuatro | 999 | C4 | Su Id |

Luego se probara a obtener las líneas, para el método getLinea(long linea\_id) se probara con los resultados de la parte anterior como parámetro de este método. Para el método gelAllLinea() se comprobara que la lista que retorna contiene las líneas del apartado anterior.

A continuación probaremos el método getColorbyLinea(long linea\_id) usando como parámetros las lineas P1, P2 y P3 y asegurándonos que retorna los colores indicados en la tabla.

Realizaremos pruebas unitarias para comprobar el correcto funcionamiento de la función de ParseJSON en el caso de obtener todas las lineas de un json y convertirlo a una lista de lineas. Para su prueba se usara un fichero que contiene lo siguiente:



Tambien probaremos del ParseJSON el resto de métodos para parseas paradas de bus, para ello se contara con pequeños json para comprobar su funcionamiento, esos ficheros contienen para la prueba de las paradas:



Y para la prueba de las paradas con nombre:



Las pruebas unitarias que realizaremos sobre ListLineasPresenter son para probar los métodos obtenLineas ya que es el que llevan toda la lógica de la clase, ya que se encarga de obtener de la base de datos local la lista de las líneas de bus este método se ejecuta de forma asíncrona no en el hilo principal por lo que probaremos también el funcionamiento de ese AsyncTask.

Se comprobara sus correcto funcionamiento ejecutando el método y comprobando que la lista de líneas de bus en la clase se rellena con los datos correctos.

En el caso de que todo sea correcto se llenara la lista con líneas.

En caso de que se produzca alguna error retornara algo indicándolo.

Para comprobar que esta funciona correctamente se supone que esta precargado con los datos de las tablas anteriores así que la lista debería contener L1, L2, L3 y L4.

Pruebas de integración.

Primero se realizaran las pruebas de integración de la capa de presentación (ListLineasPresenter) con la base de datos (DatabaseHelper) haciendo las pruebas unitarias de la capa de presentación pero con la base de datos local.

Para las pruebas de integración se realizara la prueba de la interfaz, LineasActivity, con la capa de presentación, ListLineasPresenter, para así comprobar que se muestran las lineas de la bases de datos repitiendo las pruebas unitarias que se definieron sobre la capa de presentación.

Pruebas de aceptación.

Prueba 1: Comprobar líneas

1. El usuario selecciona la función Mostrar líneas en el menú principal.

2. El sistema obtendrá las líneas de la base de datos local.

3. Se mostrará un listado de todas las lineas, con el formato correcto.

Prueba 2: Recargar base de datos.

1. El usuario selecciona la función líneas en el menú principal.

2. El usuario selecciona la función de recargar la base de datos.

3. Se mostrara indicadores para mostrar que se esta ejecutando, se notificara cuando termine y se mostraran los nuevos datos obtenido.

Prueba 3: Base de datos vacía.

1. El usuario selecciona la función líneas en el menú principal.

2. Se intentan cargar los datos pero al estar vacía, se notifica y se recomienda actualizarla.