Proyecto final de curso

Android: **Programación de aplicaciones**

Nombre de la aplicación: **LigthSource**

Autor: **Óscar Motto Varea**

Qué hace la aplicación:

Esta APK permite obtener un espectro de luz visible desde 380nm hasta 780nm a partir de una coordenada RGB, una temperatura de color CCT o un espectro obtenido a través de un espectrómetro portátil. La APK calcula los coeficientes de cada canal de la luminaria para posteriormente enviarlos mediante una conexión TCP a las luminarias seleccionadas en tiempo real. Para ello requiere de un fichero de calibración de la luminaria en formato JSON.

Licencia:

A destacar:

- Esta APK está ligada a un tipo de luminaria concreta cuyo fichero de calibración, en formato **JSON** se adjunta, para poder realizar el algoritmo de "fitting" en dicha luminaria.
- Inicialmente la APK comprueba si hay cargado el fichero de calibración, si no lo está, abre una ventana para seleccionar dicho archivo.
- Se ha implementado un **servicio web** en PHP para poder almacenar, recuperar y borrar los espectros generados en una base de datos MySQL.
- Se ha empleado la clase **Application** para almacenar variables globales a toda la aplicación, para que todas las vistas puedan interactuar entre sí.
- Los dispositivos Luminarias y Espectrómetros pueden ser creados y borrados en una **base de datos SQLite** en el dispositivo móvil.
- Las comunicaciones entre la APK y los diferentes dispositivos, envío de tramas, lectura de espectros, se realizan mediante **conexiones TCP**.
- Se han generado **vistas nuevas** para el selector de color y el selector de temperatura de color.
- Se han **agrupado todos los textos** en dos ficheros de idiomas español e inglés tal que por defecto se muestra en inglés.

- Se han usado **vistas de terceros** para mostrar los espectros obtenidos.
- Se ha generado un **fichero de estilo** propio para esta APK
- Se han fijado todas las **vistas** en modo "**portrait**"
- Se ha empleado **Fragments TabHost** para los menús.
- Se han utilizado las **OptionsMenu** con iconos**.**
- Se ha eliminado el **ActionBar**.
- Se han implementado **adaptadores de arrays** y listas propios

Mejoras a realizar:

- **Adaptar la APK para distintas resoluciones de pantalla** y otros dispositivos móviles como tablets.

Cómo lo hace:

Cuando arranca la Actividad se genera un FragmentTabHost con tres pestañas:

- Fragment RGBControl:
    - Esta Fragment está compuesto de una vista propia para el selector de color RGB con máxima saturación, un texto, una barra para graduar la intensidad y un botón para guardar el espectro obtenido en una base de datos almacenada en un servidor web.
    - Cuando modificamos el valor de progreso del selector de color llamamos al Listener para que obtenga el RGB y lo transforme en los coeficientes que generan el espectro de luz visible de la luminaria. Actualiza las variables globales de la aplicación con dichos coeficientes, y los envía a todas las luminarias definidas mediante TCP/IP.
    - Cuando modificamos la barra de intensidad toma los coeficientes que están en almacenados previamente en la clase tipo Application y los multiplica por un factor progresión / 100. Luego se envían dichos coeficientes a las luminarias.
    - Cuando pulsamos el botón de guardar, aparece un dialogo que nos pregunta el nombre del espectro y comprueba que no esté vacío para posteriormente guardar nombre y coeficientes en una base de datos MySQL mediante un servicio web.
- Fragment CCTControl
    - Esta Fragment está compuesto de una vista propia para el selector circular de CCT, un texto, una barra para graduar la intensidad y un botón para guardar el espectro obtenido en una base de datos almacenada en un servidor web.
    - Cuando modificamos el valor de progreso del selector de CCT llamamos al Listener para que obtenga el CCT y lo transforme en los coeficientes que generan el espectro de luz visible de la luminaria. Actualiza las variables

globales de la aplicación con dichos coeficientes, y los envía a todas las luminarias definidas mediante TCP/IP.

- Cuando modificamos la barra de intensidad toma los coeficientes que están en almacenados previamente en la clase tipo Application y los multiplica por un factor progresión / 100. Luego se envían dichos coeficientes a las luminarias.

- Cuando pulsamos el botón de guardar, aparece un dialogo que nos pregunta el nombre del espectro y comprueba que no esté vacío para posteriormente guardar nombre y coeficientes en una base de datos MySQL mediante un servicio web.

- Fragment SpectroControl

  - Esta Fragment está compuesto de una vista de terceros donde mostramos el espectro obtenido a partir de RGB, CCT o espectrómetro, u botón para cargar un espectro almacenado en la nube o un espectro obtenido a parte del espectrómetro, un texto, una barra para graduar la intensidad y un botón para guardar el espectro obtenido en una base de datos almacenada en un servidor web.

  - Cuando pulsamos el botón de Cargar, arrancamos una vista diálogo donde nos exige seleccionar el origen del espectro, de un fichero de la nube o de una lectura del espectrómetro. En ambos casos se abre otra vista nueva.

    - GetSpectro de tipo FragmentActivity donde hay un thread que se encarga todo el tiempo de ir leyendo del espectrómetro previamente seleccionado mediante conexiones TCP/IP y de mostrarlo en una vista.

    - ListLights de tipo ListActivity donde se abre una lista de los espectros previamente guardados en la web. Si relaizamos un LongClick podremos borrarlos. Si hacemos un click normal seleccionamos el espectro.

  - Cuando modificamos la barra de intensidad toma los coeficientes que están en almacenados previamente en la clase tipo Application y los multiplica por un factor progresión / 100. Luego se envían dichos coeficientes a las luminarias.

  - Cuando pulsamos el botón de guardar, aparece un dialogo que nos pregunta el nombre del espectro y comprueba que no esté vacío para posteriormente guardar nombre y coeficientes en una base de datos MySQL mediante un servicio web.

Luego comprueba que haya cargado un archivo de calibración para poder hacer el “fitting” del espectro a los coeficientes de los canales de las luminarias. Si no está cargado, se abre una nueva vista tipo ListActivity que representa un explorador de archivos, donde cada elemento de la lista es un archivo o una carpeta. Esta actividad nos permite ir navegando por el sistema de archivos para encontrar el fichero JSON que queramos cargar.

Dentro de la activad se ha definido un OptionsMenu que se accede mediante el botón de Menú del dispositivo móvil. Dentro de las posibles opciones tenemos llamadas a las clases:

- ListFiles tipo ListActivity

    - Representa un explorador de archivos, donde cada elemento de la lista es un archivo o una carpeta. Esta actividad nos permite ir navegando por el sistema de archivos para encontrar el fichero JSON que queramos cargar

- AddDevices tipo Activty

    - En esta actividad entramos el Nombre, el Puerto, la IP y el tipo de dispositivo que queremos dar de elata en la base de datos SQLite.

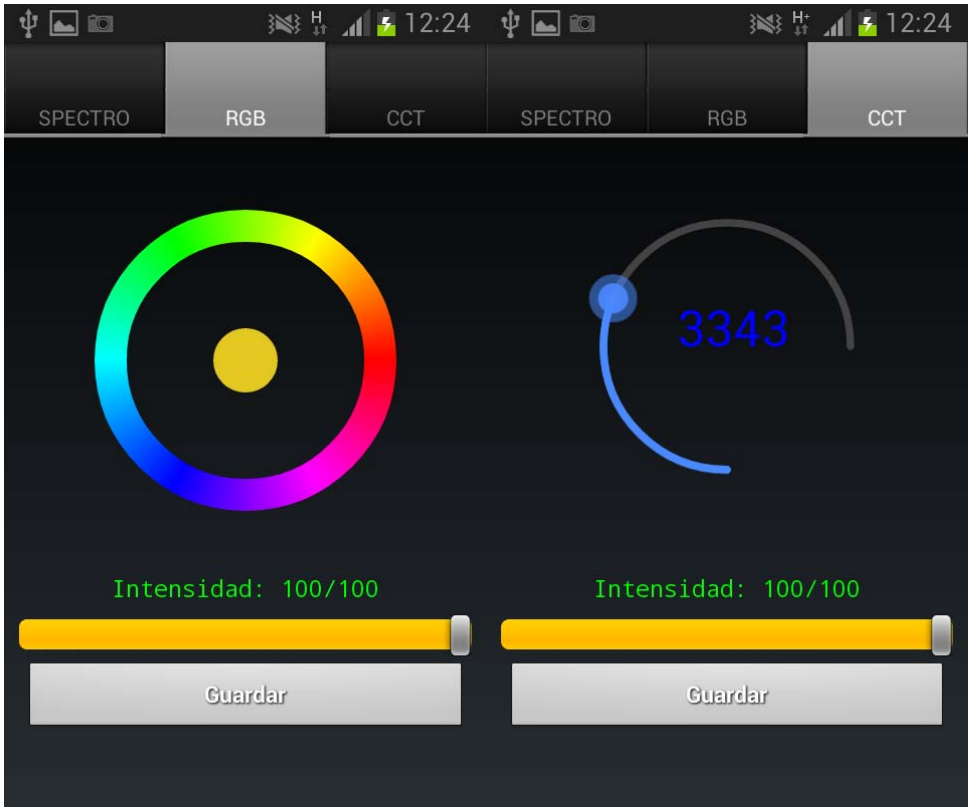
- ListDevices tipo ListActivity

    - En esta lista se cargan los dispositivos tipo Luminaria almacenados en la base de datos SQLite. Cada elemento de la lista contiene un icono, el Nombre, Puerto, IP y el CheckBox de selección. Los coeficientes se enviarán a todas aquellas luminarias que estén seleccionadas. Con un LongClick podremos borrar el dispositivo de la lista y de la base de datos.

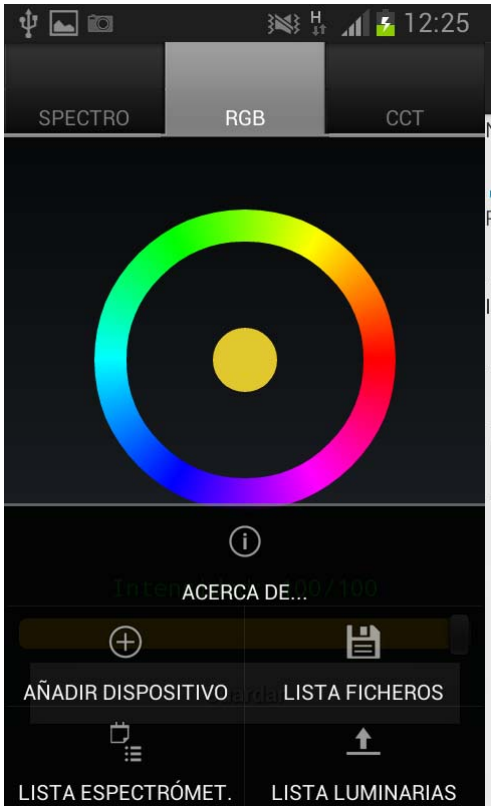
- ListSpectros tipo ListActivity

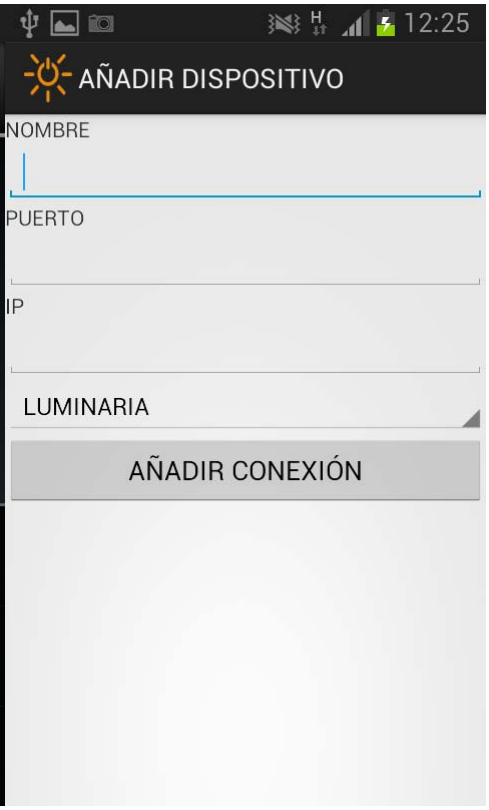
    - En esta lista se cargan los dispositivos tipo Espectrómetro almacenados en la base de datos SQLite. Cada elemento de la lista contiene un icono, el Nombre, Puerto, IP y un RadioButton para seleccionar un único espectrómetro de la lista. El espectrómetro del cual se obtendrán los espectros será aquel que haya sido seleccionado en esta actividad Con un LongClick podremos borrar el dispositivo de la lista y de la base de datos.

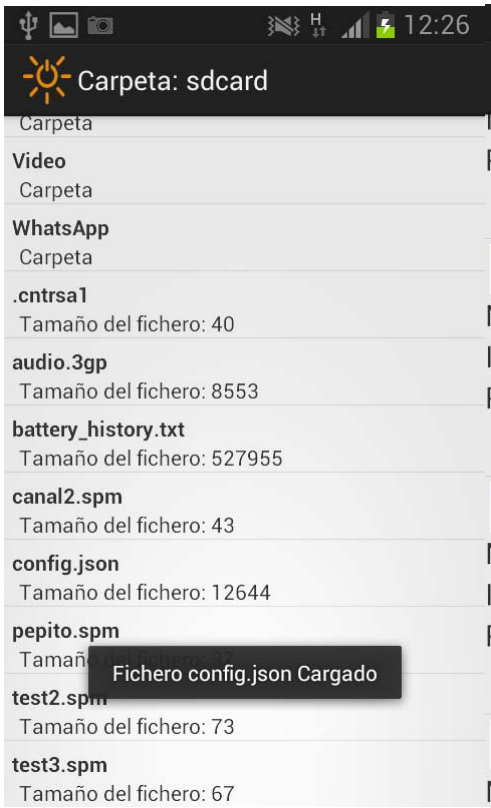
Capturas de pantalla:


12:24
12:24
SPECTRO
RGB
CCT
SPECTRO
RGB
CCT
3343
Intensidad: 100/100
Intensidad: 100/100
Guardar
Guardar

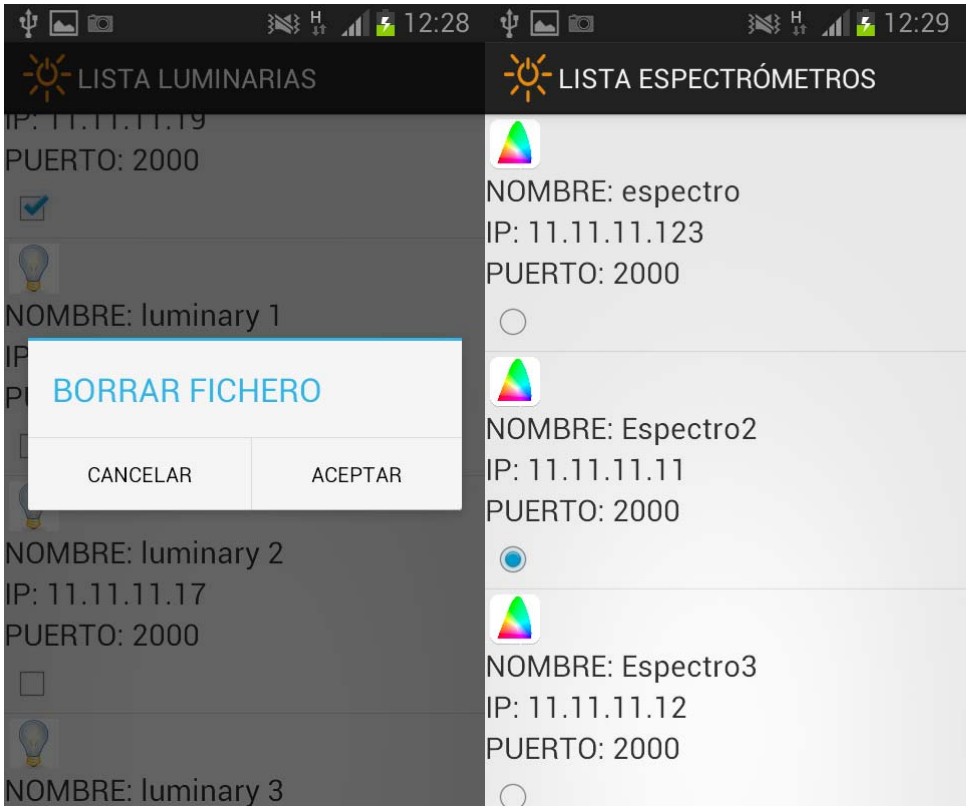

13:31
12:24
SPECTRO
RGB
CCT
Nombre del Espectro
Nombre del Espectro
ACEPTAR
CANCELAR
Nombre del Espectro
Nombre del Espectro
ACEPTAR
CANCELAR
Intensidad: 100/100
Guardar
que
de
la
123 Sym
ES(ES)
.com
Fin.


12:25
SPECTRO
RGB
CCT
ACERCA DE...
AÑADIR DISPOSITIVO
LISTA FICHEROS
LISTA ESPECTRÓMET.
LISTA LUMINARIAS


12:25
AÑADIR DISPOSITIVO
NOMBRE
PUERTO
IP
LUMINARIA
AÑADIR CONEXIÓN


12:26
Carpeta: sdcard
Carpeta
Video
Carpeta
WhatsApp
Carpeta
.cntrsa1
Tamaño del fichero: 40
audio.3gp
Tamaño del fichero: 8553
battery_history.txt
Tamaño del fichero: 527955
canal2.spm
Tamaño del fichero: 43
config.json
Tamaño del fichero: 12644
pepito.spm
Fichero config.json Cargado
test2.spm
Tamaño del fichero: 73
test3.spm
Tamaño del fichero: 67


12:27
LISTA LUMINARIAS
PUERTO: 2000
NOMBRE: luminary 1
IP: 11.11.11.18
PUERTO: 2000
NOMBRE: luminary 2
IP: 11.11.11.17
PUERTO: 2000
NOMBRE: luminary 3


12:28
LISTA LUMINARIAS
PUERTO: 2000
NOMBRE: luminary 1
BORRAR FICHERO
CANCELAR
ACEPTAR
NOMBRE: luminary 2
IP: 11.11.11.17
PUERTO: 2000
NOMBRE: luminary 3
12:29
LISTA ESPECTRÓMETROS
NOMBRE: espectro
IP: 11.11.11.123
PUERTO: 2000
NOMBRE: Espectro2
IP: 11.11.11.11
PUERTO: 2000
NOMBRE: Espectro3
IP: 11.11.11.12
PUERTO: 2000


12:29
LISTA ESPECTRÓMETROS
NOMBRE: espectro
IP: 11.11.11.123
PUERTO: 2000
BORRAR FICHERO
CANCELAR
ACEPTAR
NOMBRE: Espectro3
IP: 11.11.11.12
PUERTO: 2000
12:41
SPECTRO
RGB
CCT
LISTA ESPECTROS
NOMBRE: Amarillo
BORRAR FICHERO
ACEPTAR
CANCELAR
NOMBRE: cct6500
NOMBRE: ctt1465
Guardar