

# Proyecto final de curso

## Android: Fundamentos de Programación

### (julio-noviembre 2015)

**Nombre de la aplicación:** ElectriCost

**Autor:** Juan Carlos Perez Gonzalez

#### **Qué hace la aplicación:**

Permite consultar el coste de la energía eléctrica en Kw/h según la hora del día para la tarifa general, la tarifa nocturna y la tarifa de vehículos eléctricos:

- La consulta se puede realizar del día actual, del siguiente y de días anteriores.
- La información se mostrará en forma de gráfico y en forma de listado.
- Se puede configurar para que avise automáticamente de la hora más barata y de la más cara al usuario.

Además la aplicación también nos permite visualizar en un mapa los puntos públicos de recarga de vehículos eléctricos próximos a nuestra ubicación.

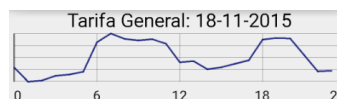
#### **Licencia:**

Autorizo la difusión del código fuente con fines educativos siempre que se haga referencia al autor bajo los términos generales de la licencia "Academic Free License v.3.0".

#### **A destacar:**

- La vista inicial ha sido rediseñada para verse correctamente en vertical y horizontal.
- Idiomas: español e inglés.
- Obtiene los precios de la luz de la web de REE (Red Eléctrica Española) <http://www.esios.ree.es> (formato XML)
- Obtiene la ubicación de los puntos públicos de recarga de la web del Ministerio de Industria, Energía y turismo <http://www.movele.es/> (formato html).
- Muestra los datos:

- en modo grafico:



- en modo de listado donde se nos marca el precio más barato (en verde), el más caro (en rojo) y la tendencia con respecto a la hora anterior (flecha roja o verde según tendencia):

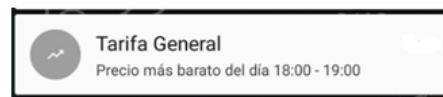
↓ 01:00 h	0.11916 €/kWh
↑ 02:00 h	0.11956 €/kWh
↑ 03:00 h	0.12123 €/kWh
↑ 04:00 h	0.12166 €/kWh
↑ 05:00 h	0.12261 €/kWh
↑ 06:00 h	0.1329 €/kWh
↑ 07:00 h	0.13593 €/kWh
↓ 08:00 h	0.13406 €/kWh

- Utiliza el GPS para establecer nuestra ubicación.

- Muestra en Google Maps para mostrar la ubicación actual y los puntos de recarga (📍):



- Permite avisar automáticamente de la hora más barata y de la más cara al usuario mediante una notificación:



### **Cómo lo hace:**

Disponemos de una actividad principal "MainActivity" que se encarga de lanzar las actividades principales:

- "ObtenerPreciosActivity" la cual es la responsable de:
  - obtener los precios de la luz para un determinado día y tarifa.
  - mostrar los precios en un gráfico (se hace uso del componente "GraphView")
  - mostrar los precios en un ListView (para mostrar los items se ha creado la clase "ListAdapter").

Además dispone de una clase privada "AccesoWeb" que hereda de "AsyncTask" y se encarga de acceder a la página web "<http://www.esios.ree.es/Solicitar/xxx>" , obtener el XML y recuperar los datos de las horas y los precios de la luz haciendo uso de la clase "ManejadorXMLREE" (extrae los datos con SAX).

- "MostrarEstacionesActivity" la cual se encarga de obtener la información de los puntos de recarga y mostrarlos en el mapa, a la vez que obtiene la ubicación del usuario (implementa el interfaz "LocationListener"). También dispone de una clase privada "AccesoWeb" que hereda de "AsyncTask" y se encarga de acceder a la página web "<http://www.movele.es/index.php/mod.puntos/mem.mapa>" , recorrerse dicha página html y extraer la información y ubicación de los puntos de recarga.

Por otro lado para la gestión de las notificaciones tenemos dos receptores de anuncios

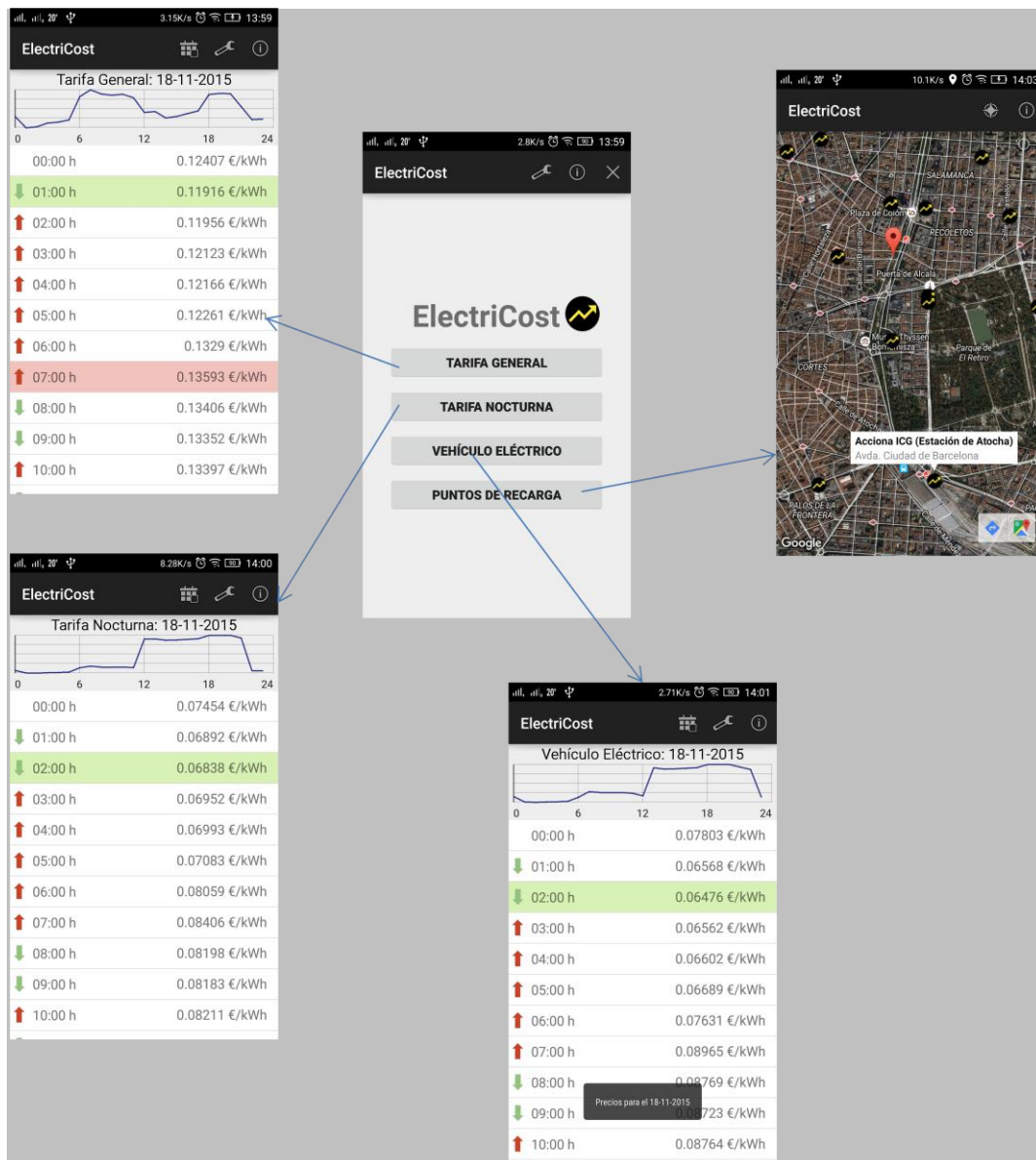
- "ReceptorArranque" que recibe el anuncio de sistema operativo arrancado ("BOOT\_COMPLETED") y el de aplicación instalada ("PACKAGE\_ADDED") ante lo cual programa, mediante la clase "AlarmManager", que se lance el servicio "ServicioAlarma" todos los días a las 23:00, para calcular las horas más caras y más baratas del día siguiente y lanzar las notificaciones a dichas horas. Se programa a

las 23:00 porque los datos del día siguiente se publican aproximadamente a las 22:00. Si a dicha hora el terminal no tiene acceso a internet se añade un registro con la fecha a calcular en la tabla de reintentos (mediante la clase “BDDReintentos” que hereda de “SQLiteOpenHelper”), para que cuando tengamos acceso se lance el “ServicioAlarma” de nuevo.


- “ReceptorCambioConexion” recibe el anuncio de que ha habido un cambio en la conexión (“CONNECTIVITY\_CHANGE”) y si tenemos conexión a internet accede a la tabla reintentos para ver si hay que lanzar de nuevo el servicio “ServicioAlarma”. Si existe una fecha en dicha tabla lanza el servicio para que recupere los datos para dicha fecha.

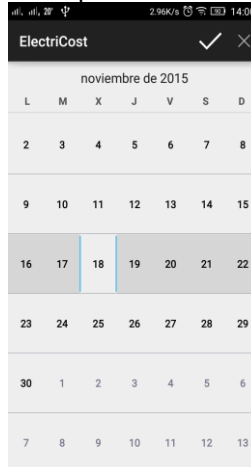
Como hemos visto los receptores de anuncios se encargan de lanzar el servicio “ServicioAlarma” el cual calcula la hora más cara y más barata de cada tipo de tarifa y en función de las preferencias del usuario programa, mediante la clase “AlarmManager”, que se lance el servicio “ServicioAvisoPrecio”. Este servicio mostrará una notificación al usuario a esas horas avisándole de que a esta hora el precio de la luz es la más barata o la más cara del día según corresponda.




### Captura de pantallas:

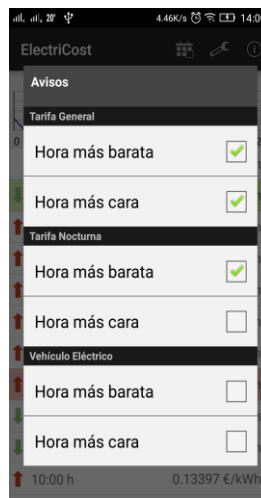



Cada pantalla tiene una barra de acciones que nos permite realizar una serie de funciones pulsando en los siguientes botones de acción:

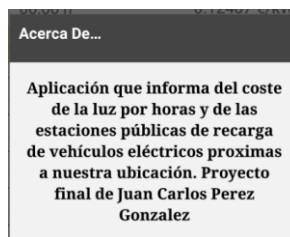
-  para mostrar la actividad del calendario en la que el usuario podrá seleccionar el día respecto al cual quiere obtener el precio de la luz:



-  para aceptar el cambio en el día seleccionado en el calendario
-  para salir de la actividad actual
-  para mostrar la ventana de configuración del usuario



-  para mostrar la ventana de información de la aplicación



-  para centrar el mapa respecto a nuestra ubicación