

Lector de sensores arduino-android.

Programación de Sistemas - Grao en Ingeniería Informática 1ºQ

20/10/2014

Persona de contacto: Santiago García Lesta
<santiago.lesa@udc.es>
Oscar Blanco Novoa
<o.blanco@udc.es>

1 Introducción.

1.1 Objetivo:

El proyecto se basa en una aplicación de android comunicada con una estación de sensores controlada por un arduino, comunicado mediante un cable con el dispositivo android. Con el objetivo de obtener y almacenar datos de los sensores y modificar la configuración de la estación.

1.2 Motivación:

Dicha estación de sensores es funcional hoy en día pero carece de una interfaz adecuada para motorizar los datos obtenidos.

1.2 Intereses:

Modificar el código actual de la estación para que envíe los datos a través de la entrada usb del dispositivo.

2 Análisis de requisitos.

2.1 Funcionalidades que implementaremos:

- Motorización de datos en tiempo real.
- Almacenaje de datos
- Visualización de un histórico
- Modificación de la configuración de la estación
- Comunicación vía wifi.
- Obtención de datos remotamente desde otro terminal.
- Guardar configuración de la aplicación.
- Adición de nuevos sensores desde el dispositivo android.

2.2 Prioridades y orden:

- 1.Comunicación desde el arduino hacia el dispositivo android. Dependencias()
- 2.Comunicación desde el dispositivo android hacia el arduino. Dependencias()
- 3.Refactorización del código de la estación de sensores. Dependencias()
4. Motorización de datos en tiempo real .Dependencias (1,2,3)
5. Modificación de la configuración de la estación. Dependencias (1,2,3,4)
6. Almacenaje de datos. Dependencias(4)
7. Visualización de un histórico(6)
8. Adición de nuevos sensores desde el dispositivo android. Dependencias(4)
9. Guardar configuración de la aplicación. Dependencias(8)
10. Obtención de datos remotamente desde otro terminal. Dependencias(4,6)
11. Comunicación vía wifi. Dependencias(1,2,3)

Básicos= 1,2,3,4.

En caso de acabar los puntos básicos antes de la fecha límite implementaremos los puntos 5, 6 y 7.

En caso de haber realizado los puntos anteriores realizaremos los puntos 8 y 9.
En el mejor de los casos implementaremos el punto 10 y el 11.

2.3 Responsabilidades:

Individuales:

Santiago García Lesta será responsable de los puntos: 1 2 6 7.

Oscar Blanco Novoa será responsable de los puntos: 3 5 8 9.

Colectivas:

Debido a la imposibilidad de trabajo en paralelo(una sola estación de sensores y posible complejidad) los siguientes puntos serán responsabilidad de los dos: 4 9 10.

3.1 Hitos:

10º semana: Funcionalidad básica terminada. (test de correcta visión de los datos).

11º semana: Funcionalidad básica más la opción de la configuración de la estación.

13º semana: Posibilidad de almacenaje y visualización de históricos.

14º semana: Añadir y eliminar de forma dinámica sensores y guardar configuración.

16º semana: Acceso remoto y comunicación wifi.

3.2 Posible planificación:

Conocimientos previos: experiencia con arduino, experiencia con las librerías de comunicación del arduino.

Se ha probado la comunicación con el arduino desde el dispositivo. La estación de sensores ha sido probada. Se debe testear la comunicación por wifi.

Posibles errores: problemas de capacidad por las librerías de comunicación en el arduino.

4.1 Diseño: Una única pantalla para visionar los datos en tiempo real y un botón para conectar con el dispositivo y opcionalmente gráficas.

4.2 Posible arquitectura:

Posiblemente con el avance del proyecto la arquitectura será adaptada a las necesidades encontradas.

Tendremos una actividad principal donde en una primera instancia visualizaremos los datos en tiempo real y tendremos la opción de conectarnos con el dispositivo.

Utilizaremos un broadcast receiver para gestionar la conexión usb.

En segundo plano tendremos una actividad que se ocupe de la comunicación con el arduino.

Desde la actividad principal lanzaremos una AsyncTask que se ocupe de pedir los datos al servicio y actualizar los datos en la pantalla de la actividad principal.

En la segunda entrega añadiremos una segunda actividad llamada desde la primera para gestionar la configuración de la estación base.

En la tercera entrega le añadiremos al servicio la posibilidad de guardar los datos que reciba en una base de datos y desde la actividad principal la opción de llamar a una tercera actividad para la visualización de dichos datos.

En la cuarta entrega añadiremos la opción en la actividad de añadir y eliminar sensores y hacer persistente dicha configuración.

En la quinta entrega añadiremos la opción de que el servicio se comunique vía wifi

Primera entrega

1 Funcionalidades.

Las funcionalidades implementadas hasta el momento son las siguientes.

- 1.Comunicación desde el arduino hacia el dispositivo android.
- 2.Comunicación desde el dispositivo android hacia el arduino.
- 3.Refactorización del código de la estación de sensores.
4. Motorización de datos en tiempo real.

2 Diseño.

Una única pantalla para visionar los datos en tiempo real, un botón para conectar con el dispositivo y un botón con estado para gestionar una luz en la estación. En caso de perder la conexión con la placa se pide automáticamente la autorización del usuario para reconectar.

3 Planificación.

Hemos aprovechado la existencia de la estación de sensores funcional y hemos modificado el código para hacerlo compatible con nuestra aplicación para android.

4 Problemas encontrados.

Se nos ha presentado el problema de que el hardware de la estación de sensores utiliza los pines 10-13 del arduino como salidas SPI, algo que no es compatible con nuestro arduino Leonardo ya que esta versión solo implementa la salida específica de SPI de la placa, lo que nos ha obligado a deshabilitar la pantalla LCD y uno de los sensores. Algo que intentaremos solventar en próximas iteraciones.

Otro de los problemas que nos ha retrasado ha sido el no poder depurar por usb mientras estamos obteniendo datos de la estación, al estar el puerto USB del móvil ocupado.

Problema vigente: error al coincidir el encendido del led con el envío de datos de los sensores

5 Pruebas.

Es necesario un arduino leonardo(o compatible con el protocolo OTG) , el hardware específico para el correcto funcionamiento del código para la estación base y el cable tipo OTG microUSB.

Con todo el hardware correctamente conectado, abriremos la aplicación y pulsaremos en conectarnos, a continuación se nos solicitara el permiso para el uso de dispositivos USB, al aceptar la aplicación mostrará los datos en tiempo real. En caso de desconexión del arduino la aplicación se volverá a conectar automáticamente al conectar el dispositivo nuevamente. Además disponemos de la opción, a modo de configuración de la estación, de encender un led en el propio arduino.