



ESTUDIOS DE INGENIERÍA
DE TELECOMUNICACIÓN
PROYECTO FIN DE CARRERA

Mota sensora para monitorización de obras de arte

CURSO: 2014/2015

Fernando Pérez Bueno



ESTUDIOS DE INGENIERÍA DE TELECOMUNICACIÓN

Mota sensora para monitorización de obras de arte

REALIZADO POR:

Fernando Pérez Bueno

DIRIGIDO POR:

Alberto J. Palma López
Fernando Martínez Martí

DEPARTAMENTO:

Departamento de Electrónica y Tecnología de
Computadores

Granada, Junio de 2015

Mota sensora para monitorización de obras de arte

Fernando Pérez Bueno

PALABRAS CLAVE: Android, Arduino, monitorización, conservación preventiva, Bluetooth, sensor, luz, temperatura, humedad relativa, I²C, inalámbrico, arte.

RESUMEN:

La conservación preventiva de obras de arte es la principal corriente de conservación actual. Vigilar y mantener el entorno de la obra en unas condiciones optimas que eviten su deterioro es la opción más económica, duradera y menos invasiva de los métodos para su perfecta conservación. Actualmente existen sensores y microcontroladores de bajo coste y tamaño muy reducido, que permiten llevar a cabo la monitorización y control del entorno de forma muy discreta y eficiente.

Este proyecto versa del desarrollo de un sistema electrónico de monitorización ambiental, capaz de tomar medidas de temperatura, humedad relativa e iluminación que son los factores más influyentes sobre la conservación de una obra. Se implementara un sistema formado por una mota sensora basada en Arduino y una aplicación disponible para dispositivos móviles Android. Entre ambos componentes formaran un sistema que procese, monitorice y almacene los datos, permitiendo su análisis y control inmediato o su posterior consulta.

Sensing dot for art monitoring

Fernando Pérez Bueno

KEYWORDS: Android, Arduino, monitoring, preventive conservation, Bluetooth, sensor, light, temperature, relative humidity, I²C, wireless, art

ABSTRACT:

Preventive conservation for art pieces is the main move of actual conservation. To watch and maintain the environment of the art piece in optimal conditions to protect it from deterioration is the most inexpensive, durable and less invasive of the options available to achieve the perfect conservation of the art piece. Nowadays there are low-cost low-size sensors and microcontrollers, that allow the monitoring and control of the environment with discretion and efficiency.

This project focuses on the development of an electronic system of environmental monitoring, capable of the measurement of temperature, relative humidity and lighting, which are the most influencing factors on the art piece conservation. The system will be implemented on two different parts. A hardware side, a sensing dot based in Arduino and an application for Android mobile devices. Both components form part of a system to process, monitors results and store the data, allowing it to be analyzed and controlled immediately or to be consulted later.

D. _____, profesor del departamento de Electrónica y Tecnología de Computadores de la Universidad de Granada, como director del Proyecto Fin de Carrera de D. Fernando Pérez Bueno

Informa:

Que el presente trabajo, titulado:

Mota sensora para monitorización de obras de arte

ha sido realizado y redactado por el mencionado alumno bajo su dirección y con esta fecha autoriza su presentación:

Granada, a ____ de _____ de 2012

Fdo.

D. _____, profesor del departamento de Electrónica y Tecnología de Computadores de la Universidad de Granada, como director del Proyecto Fin de Carrera de D. Fernando Pérez Bueno

Informa:

Que el presente trabajo, titulado:

Mota sensora para monitorización de obras de arte

ha sido realizado y redactado por el mencionado alumno bajo su dirección y con esta fecha autoriza su presentación:

Granada, a ____ de _____ de 2012

Fdo.

Los abajo firmantes autorizan a que la presente copia de Proyecto Fin de Carrera se ubique en la Biblioteca del Centro y/o departamento para ser libremente consultada por las personas que lo deseen.

Granada, a __ de ____ de 2015

Agradecimientos

Este trabajo, y con ello la culminación de mi carrera, no hubiera sido posible sin la confianza, el compromiso y el apoyo de Alberto Palma y Fernando Martínez, directores de este trabajo a los que quiero hacer explícito mi reconocimiento.

A Rafael Marañón, que ha compartido conmigo su experiencia y paciencia respondiendo mis dudas sobre el desarrollo de la aplicación.

A Alejandra, que además de haber servido de inspiración y motivación para el desarrollo de este proyecto, me ha permitido el tiempo necesario.

Índice general

Índice de Figuras	19
I Descripción del proyecto.....	21
Capítulo 1 - Introducción.....	23
1.1. Descripción del proyecto	23
1.2. Motivación	23
1.3. Objetivos.....	24
Referencias	26
Capítulo 2- Antecedentes.....	27
2.1. Arduino	27
2.1.1. Introducción de la plataforma.....	27
2.1.2. Proyectos con Arduino.....	28
Monitorización de temperatura y humedad con Arduino y GadgetKeeper	28
2.2. Plataformas móviles	28
2.2.1. Android	28
2.2.2. iOS.....	29
2.2.3. Windows Phone.....	29
2.2.4. Ubuntu Touch.....	29
2.2.5. Conclusiones	29
2.3 Conexiones inalámbricas	30
2.3.1. Bluetooth	30
2.3.2. Wi-Fi.....	30
2.3.3 NFC.....	30
2.3.4. Conclusiones	31
2.4. Sistemas comerciales actuales.....	31
2.4.1 Madge Tech Data Loggers.....	31
2.4.2. Hanwell Synergy Software	32
2.4.3 MHS - Monitoring Heritage System.....	32
2.4.4. Conclusiones	33
Referencias	34
Capítulo 3. Mota sensora para monitorización de obras de arte	35

3.1. Esquema general.....	35
3.2. Tecnologías y componentes.....	36
3.2.1. Arduino	36
3.2.2. Android	37
3.2.3. Bluetooth	37
3.2.4. Sensores	38
3.3. Flujo de la información	39
3.4. Funcionalidades del sistema	39
3.4.1. Funcionalidades de la placa Arduino	39
3.4.2. Funcionalidades de la aplicación Android: ArtSense	40
Referencias	43
II Proceso de desarrollo.....	45
Capítulo 4 - Estudio de requisitos y planificación.....	47
4.1. Especificación de requisitos.....	47
4.1.1. Requisitos funcionales	47
4.1.2. Requisitos no funcionales	49
4.2. Planificación	49
4.2.1. Planificación previa.....	50
4.2.2. Transcurso del proyecto	52
4.2.2.1. Primeros pasos	52
4.2.2.2. Elección de sensores	52
4.2.2.3. Toma de contacto con Arduino	53
4.2.2.4. Toma de contacto con Android	53
4.2.2.5 Diseño y desarrollo de la PCB.....	53
4.2.2.6 Adquisición y procesamiento de los datos.....	54
4.2.2.7 Desarrollo de la aplicación ArtSense.....	54
4.2.2.8. Integración de plataformas y finalización.	55
4.2.3. Reorganizaciones de la planificación	55
Referencias	57
Capítulo 5. Diseño e implementación.....	59
5.1. Diseño de la mota sensora.....	59
5.1.1. Sensor de luz	60
5.1.2. Sensor de temperatura y humedad.....	61
5.1.3. Placa Arduino Mega ADK	62

5.1.4. PCB de interconexión.....	63
5.1.5. Bluetooth USB dongle.....	63
5.1.6. Alimentación del dispositivo	64
5.1.7. Código Arduino.....	64
5.2. Diseño de la aplicación Android: ArtSense	65
5.3. Diseño del protocolo de comunicación	68
5.4. Consideraciones finales	70
Referencias	71
III Evaluación y Conclusiones.....	73
Capítulo 6. Evaluación y resultados	75
6.1. Grado de cumplimiento de los objetivos.....	75
6.2. Pruebas de alcance.....	76
6.3. Estudio de la capacidad de transmisión	78
6.4. Estudio de compatibilidad	79
6.5 Encuesta sobre la aplicación ArtSense	80
Capítulo 7. Conclusiones y trabajo futuro.....	83
7.1. Conclusiones	83
7.2 Trabajo futuro	83
Apéndice I. Manual de usuario.....	87
Apéndice 2. Presupuesto	91
Referencias	91

Índice de Figuras

Figura 2.3.1: Versiones de Bluetooth.....	30
Figura 3.1.1 Esquema general del sistema.....	35
Figura 3.2.1: Placa Arduino Mega ADK	37
Figura 3.2.2: (a) Sensor de humedad y temperatura HIH60XX; (b) Sensor de luz TSL2561	38
Figura 3.4.1: Pantalla principal de ArtSense. (a) Estado desconectado; (b) Estado conectado.....	41
Figura 3.4.2: Historial de Luz	42
Figura 4.2.1: Diagrama de Gantt de la planificación a priori.....	51
Figura 4.2.2: Diagrama de Gantt de la planificación a posteriori	56
Figura 5.1.1: Esquema general de la mota sensora.	59
Figura 5.1.2: Sensor de luz TSL2561 integrado en PCB y esquemático de la misma.	60
Figura 5.1.4. Sensor de humedad y temperatura HIH6130.....	61
Figura 5.1.5: Conexionado del sensor HIH6130 para su uso con I ² C	62
Figura 5.1.6: Esquemático del conexionado del sistema.	63
Figura 5.1.8: Diagrama de flujo del código para Arduino.	65
Figura 5.2.1: Esquema general de la aplicación ArtSense.....	66
Figura 5.2.2: Clases y actividades de Blueterm que se mantienen.....	66
Figura 5.2.3: Clases y actividades incluidas en ArtSense	68
Figura 5.3.1: Tramas originadas en Arduino.....	69
Figura 5.3.2: Tramas originadas en la aplicación ArtSense	69
Figura 5.3.4: Opciones de la trama de modificación de datos de configuración.	70
Figura 6.2.1: Resultados de las pruebas de alcance Bluetooth con el smartphone LG G3	78
Figura 6.5.1: Encuesta de valoración para ArtSense.....	81
Figura A1.1: Pantalla de inicio. Desconectado.	87
Figura A1.2: Pantalla principal. Conectado.	88
Figura A1.3: Historial de datos guardados. Humedad	89
Figura A2.1: Presupuesto para la construcción del prototipo de la mota sensora.	91

Parte I

Descripción del proyecto

Capítulo 1 - Introducción

Con este primer capítulo realizaremos una inmersión en el ámbito del presente proyecto. Con su lectura se facilitará la comprensión de la información contenida en la memoria. Realizaremos un breve resumen de los contenidos y su estructuración para permitir una lectura ágil del documento.

1.1. Descripción del proyecto

El proyecto desarrollado consiste en un sistema de medida portátil que permite la monitorización de los parámetros ambientales más importantes que inciden en el estado de conservación de obras de arte. En concreto y de acuerdo con la normativa del Ministerio de Educación y Cultura, interesa monitorizar de manera continua los valores de temperatura, humedad relativa y luminosidad de la atmósfera en contacto con la obra. [1]

Dado que algunas obras pueden ser más sensibles que otras, interesa tomar las medidas en un entorno cercano a la misma, de esta forma la monitorización puede adaptarse a los requisitos específicos de cada una. Para ello se ha diseñado un sistema de reducido tamaño, capaz de obtener y registrar todas las medidas necesarias sin suponer un añadido desproporcionado y que reste atención al objeto que supervisa.

La información tomada se transmite de forma inalámbrica a un dispositivo móvil, donde una sencilla aplicación permite el almacenaje de los datos, su consulta en tiempo real y el estudio posterior de las condiciones de conservación de la obra de arte.

1.2. Motivación

En lo que respecta al patrimonio cultural, España es un país privilegiado, ocupando el segundo lugar después de Italia por el enorme valor que representa. Indudablemente es nuestro deber cuidar este patrimonio que manifiesta, expresa y transmite la cultura que se ha ido desarrollando con el paso del tiempo. El cuidado del patrimonio cultural se apoya en muchos factores, desde la educación y acercamiento a los ciudadanos hasta los planes de conservación y restauración llevados a cabo por expertos. Una pieza clave del engranaje es la llamada conservación preventiva.

La conservación preventiva trata de aplicar una serie de medidas indirectas, que no interfieren con los materiales de los bienes ni modifican su apariencia. Su objetivo no es otro que garantizar el mantenimiento temporal del patrimonio en las mejores circunstancias conservativas posibles, creando para ello las condiciones ambientales óptimas que requieran en función de sus necesidades materiales y técnicas, sin olvidarnos de hacerlas compatibles con la función socio-cultural que desempeñe.

Aunque no ha sido habitual que las entidades encargadas de la gestión del patrimonio interactuaran con las de ciencia y tecnología, los avances tecnológicos actuales pueden aportar soluciones muy útiles para dar soluciones a estos requisitos. A pesar de su riqueza patrimonial, hasta la fecha España ha tenido una escasa participación en la investigación y desarrollo de proyectos en este ámbito. En este sentido, el Plan Nacional de Investigación en Conservación de Patrimonio marca una enorme diferencia, por primera vez se presta atención al desarrollo y fomento de iniciativas. Tenemos la capacidad humana y material para cambiar esta tendencia en lo que respecta a nuestro patrimonio, que es una fuente de riqueza tanto cultural como económica si se hace una buena gestión del mismo. [2]

La miniaturización de la tecnología permite mantener una vigilancia constante y detallada de forma discreta y económica. Además el desarrollo conjunto de una aplicación móvil asegura una lectura cómoda, versátil y flexible de la información obtenida, tanto en el punto de interés como a distancia.

1.3. Objetivos

Durante el desarrollo de este proyecto se planea diseñar e implementar un sistema de monitorización de obras de arte, tomando los valores de los parámetros ambientales más importantes y controlando que se mantengan dentro de un margen adecuado. Además los datos se transmitirán a un dispositivo móvil para permitir su visualización de una forma sencilla e intuitiva. El dispositivo móvil se encargará así mismo de almacenar los datos y permitir su estudio posterior. Los objetivos que marcamos para el desarrollo quedan enumerados de la siguiente forma:

- **Adquisición no invasiva de parámetros de medida:** Se utilizarán sensores capaces de adquirir las medidas necesarias sin alterar las condiciones del entorno.
- **Monitorización en tiempo real:** La conservación preventiva se basa en mantener estables las condiciones del entorno. La monitorización constante y en tiempo real, permite tomar medidas correctivas antes de que la variación de parámetros supere el umbral adecuado para una conservación óptima.
- **Control de los parámetros dentro de los márgenes establecidos:** Se establece un rango de valores permitidos para cada parámetro. El sistema deberá comprobar en primera instancia que los valores que se adquieren estén comprendidos en dicho rango.
- **Generación de alarmas:** En caso de que los valores escapen al rango de control, el sistema deberá generar alarmas tanto para avisar al personal humano como para permitir futuros controles de otros sistemas asociados.

- Transmisión inalámbrica independiente: Puesto que su utilización mayoritaria será en lugares abiertos al público y zonas de paso, se evitará el uso de cables. Añadiendo el valor de poder situarlo de forma discreta. Se debe evitar la dependencia de redes de cobertura o dispositivos de terceros.
- Visualización de los datos en tiempo real: El acceso a la información del estado actual de los parámetros de interés debe ser accesible en primera instancia. Esta información estará disponible en un dispositivo móvil conectado al sistema.
- Facilidad de uso: Se debe conseguir que el sistema sea sencillo e intuitivo para el usuario. La plataforma Arduino no exigirá más que conexión a una fuente de alimentación y una situación adecuada en el entorno de la obra de arte. La aplicación estará diseñada con una interfaz clara y sencilla dentro de la tendencia actual.
- Almacenamiento de datos: El dispositivo móvil deberá ser capaz de almacenar los datos recibidos de los distintos parámetros, contemplando un orden adecuado que permita su análisis posterior en conjunto.
- Consulta de datos: Se debe permitir la consulta de los datos almacenados en cualquier momento, independientemente de que el dispositivo esté o no conectado. Los datos deben mostrarse de forma sencilla y ordenada.
- Interpretabilidad de los datos: Se mostrarán los datos de forma que el usuario pueda interpretarlos de forma sencilla. Mostrando claramente mediante un código de colores que se han excedido los límites de control.
- Robustez: Los elementos que componen el sistema y su funcionamiento deberán carecer de cualquier tipo de fallo que dificulte su utilización por parte del usuario.

Referencias

[1] Normas de Conservación Preventiva para la Implementación de Sistemas de Control de Condiciones Ambientales en Museos, Bibliotecas, Archivos, Monumentos y Edificios Históricos. Instituto del Patrimonio Cultural de España. Ministerio de Educación y Cultura, Madrid, 2009. [Consulta 5-6-2015]. Disponible en: http://ipce.mcu.es/pdfs/IPCE_NormasClimatizacion.pdf

[2] Plan Nacional de Investigación en Conservación de Patrimonio Cultural . Instituto del Patrimonio Cultural de España. Ministerio de Educación y Cultura, Madrid, 2011. [Consulta 5-6-2015]. Disponible en: http://ipce.mcu.es/pdfs/PN_INVESTIGACION.pdf

Capítulo 2- Antecedentes

Este proyecto se enmarca dentro de la denominada conservación preventiva impulsada en la resolución de Vantaa (Finlandia) de 2010 y respaldada en 2011 por el Plan Nacional de Conservación Preventiva en España. La Unión Europea recomienda implementar la conservación preventiva como pilar fundamental para la conservación del patrimonio cultural. La base de la conservación preventiva es el estudio y mantenimiento de las condiciones ambientales ideales para evitar el deterioro de las obras de arte y otros elementos del patrimonio. La condición más crítica es la humedad relativa de su entorno, seguida por la temperatura y otros factores como la iluminación o los gases ambientales.

En este capítulo estudiaremos el estado del arte y detallaremos los antecedentes que repercuten en este proyecto. Como comienzo hablaremos de la plataforma Arduino y sus factores determinantes para la inclusión en este proyecto. Repasaremos las plataformas móviles del mercado y veremos brevemente las conexiones inalámbricas disponibles y sus posibilidades. Como cierre se mostrará un repaso a las tecnologías actuales en el mercado que marcan las líneas de desarrollo y negocio.

Al finalizar la lectura de este capítulo, se habrá logrado un conocimiento base del marco teórico del proyecto y el grado de progreso en el que se encuentra.

2.1. Arduino

2.1.1. Introducción de la plataforma

Arduino es una herramienta de desarrollo pensada para facilitar la interconexión de un micro controlador con otros periféricos electrónicos de entrada o salida. Implementa una placa base con un entorno de desarrollo de software libre y su objetivo es la flexibilidad tanto en software como en hardware. Permite programar el controlador y su entorno en un lenguaje muy similar a C. Obteniendo de forma sencilla objetos interactivos de una amplia gama, capaces de tomar datos o controlar salidas físicas. Esto y su facilidad de intercomunicación con otros dispositivos, hace de Arduino un sistema ideal para el desarrollo de prototipos. [1]

Existen múltiples alternativas a Arduino como Raspberry Pi o Beaglebone, pero tienen una capacidad mayor que el Arduino que no es necesaria para la función que cumple la plataforma en este proyecto. Claro está que se podrían aprovechar sus ventajas para delegar más tareas en el sistema, pero eso nos llevaría a resultado de mayor precio y similitud con las opciones actuales del mercado. Además Arduino tiene como punto fuerte la facilidad de comunicación con una amplia gama de sensores y el respaldo de la comunidad.

Por último, la versión de Arduino Mega ADK tiene un puerto USB incluido que nos permitirá usar un Bluetooth dongle para obtener fácilmente comunicación Bluetooth en lugar de tener que recurrir a opciones más complejas y costosas.

2.1.2. Proyectos con Arduino

Existen infinidad de proyectos basados en Arduino y adquisición de sensores. Dada la dificultad de mostrar un abanico que ilustre toda la capacidad de la plataforma nos limitaremos a un proyecto similar al que nos atañe en esta memoria.

Monitorización de temperatura y humedad con Arduino y GadgetKeeper

GadgetKeeper es una plataforma de desarrollo enfocada a facilitar la implantación del internet de las cosas. Este proyecto es uno de los ejemplos de su web de cómo utilizar la plataforma. La placa Arduino se encarga únicamente de recoger los datos de los sensores y enviarlos por cable mediante el ethernet shield de Arduino a un PC donde la API de GadgetKeeper, que se encarga de toda la gestión, almacenaje de datos y envío de alertas al exterior. [2]

Se trata de un sistema de muy bajo coste y escasa complejidad, que no aprovecha al máximo la capacidad de Arduino. Puede considerarse como un punto de partida a partir del cual puede desarrollarse un sistema más completo.

2.2. Plataformas móviles

La generación actual de dispositivos móviles tiene una amplia oferta de sistemas operativos dentro de los cuales destacamos cuatro.

2.2.1. Android

Es el sistema más popular con una cuota de mercado cercana al 85%. Está basado en Linux y diseñado para funcionar con pantallas táctiles de cualquier tamaño. Es versátil y personalizable para adaptarlo a la mayoría de dispositivos. La gran comunidad de usuarios y desarrolladores es uno de sus mayores puntos fuertes, dada la cantidad de ayuda, tutoriales y documentación que se puede encontrar. Las aplicaciones están escritas en Java y las librerías en C.

El sistema es de código abierto y su desarrollo está encabezado por Google. Tiene la desventaja de que existe una gran fragmentación de versiones consecuencia de su alta penetración en el mercado.

2.2.2. iOS

Propiedad de Apple. Está específicamente diseñado para sus dispositivos y no se permite su uso en otros terminales. Es necesario disponer de un equipo Mac para programar aplicaciones para este sistema, puesto que el entorno de desarrollo es exclusivo de esta plataforma. El lenguaje que se utiliza es Objective-C, escrito y compilado para la arquitectura ARM.

Desarrollar el proyecto en esta plataforma, además del coste de las herramientas necesarias, obligaría al usuario a disponer de un dispositivo de Apple.

2.2.3. Windows Phone

Windows Phone es la apuesta de Microsoft para los dispositivos móviles. A pesar del gran esfuerzo financiero de la compañía, sigue sin alcanzar una cuota de mercado significativa. Tiene una oferta de aplicaciones bastante más reducida que Android e iOS.

El desarrollo para Windows Phone puede llevarse a cabo mediante Microsoft Silverlight para aplicaciones con transiciones y efectos visuales. O mediante Microsoft XNA Framework, implementación nativa de .NET Compact Framework con una amplia gama de bibliotecas específicas para el desarrollo de juegos.

2.2.4. Ubuntu Touch

Este sistema, también basado en Linux, se presentó en 2013. Es un proyecto de Canonical que está empezando en el mercado de la mano de compañías como Bq. Su intención es lograr la máxima compatibilidad con la versión de escritorio, ya que comparten la misma tecnología. Utiliza el framework Qt 5 basado en la interfaz de usuario táctil.

Cualquier aplicación diseñada para Ubuntu Touch puede utilizarse sobre un escritorio de Ubuntu. Además los dispositivos táctiles pueden estar equipados con una sesión completa de Ubuntu y cambiar por completo el escritorio del sistema operativo cuando se conectan mediante una estación de acoplamiento, difuminando la línea entre sistemas portátiles y fijos.

2.2.5. Conclusiones

Android es con diferencia el sistema operativo más adecuado. La principal ventaja es la gran cantidad de usuarios que disponen de este sistema, seguido por la facilidad de desarrollo de las aplicaciones. Descartamos iOS por los requisitos de hardware propietario y Windows Phone y Ubuntu por su bajo impacto de mercado.

2.3 Conexiones inalámbricas

La comunicación de la plataforma Arduino con el exterior será un punto primordial en la funcionalidad del sistema. Es clave estudiar las opciones disponibles y tomar una decisión al respecto. Puesto que vamos a trabajar con dispositivos Android, tenemos que enfocarnos a las tecnologías disponibles para todos los terminales: Bluetooth y Wi-Fi.

2.3.1. Bluetooth

Con una difusión cada vez mayor en muchos periféricos y accesorios para los terminales móviles, bluetooth trabaja en el rango 2400 MHz a 2483,5 MHz. Según la versión de este protocolo de transmisión obtendremos una capacidad u otra.

Versión	Ancho de banda
Versión 1.2	1 Mbit/s
Versión 2.0 + EDR	3 Mbit/s
Versión 3.0 + HS	24 Mbit/s
Versión 4.0	24 Mbit/s

Figura 2.3.1: Versiones de Bluetooth

Bluetooth tiene la gran ventaja de un bajo consumo de potencia, crítico en el uso de los dispositivos móviles con baterías reducidas. Además cuenta con una estructura de protocolo que facilita las comunicaciones automáticas y opera en una banda de frecuencias que no requiere licencia.

2.3.2. Wi-Fi

Utilizando la misma banda de frecuencias de 2,4 GHz tenemos la tecnología más extendida para conexión de equipos en un área delimitada. Ofrece velocidades entre 11Mbit/s y 300 Mbit/s y permite conectarse a internet a través de un punto de acceso.

A cambio de la mayor velocidad, Wi-Fi no está diseñado para dispositivos móviles, y por tanto tiene un mayor consumo de potencia. Su uso requiere de una red de cobertura en el área de los dispositivos conectados.

2.3.3 NFC

NFC es una tecnología en alza que está cada vez más presente en los dispositivos móviles y que tiene gran cantidad de aplicaciones. Sin embargo, debemos descartarla para este proyecto dada su baja capacidad de transmisión de datos y los

aproximadamente diez centímetros a los que se reduce la distancia a la que se producen las comunicaciones

2.3.4. Conclusiones

La conexión Wi-Fi tiene unas ventajas muy claras frente a bluetooth entre las que destacan una mayor velocidad y facilidad de conexión a internet. Sin embargo, la capacidad de transmisión de bluetooth es suficiente para nuestras necesidades, y no nos obliga a disponer de una red de cobertura Wi-Fi. Aunque las redes Wi-Fi están muy extendidas, no tienen por qué ser necesarias o gozar de buena cobertura en todas las zonas donde el presente proyecto pueda ser utilizado. Además y puesto que es necesaria la conexión con un dispositivo exterior más complejo, este último puede encargarse con facilidad de la comunicación con el exterior de forma más eficiente.

La comunicación bluetooth, que puede implementarse de forma muy sencilla en Arduino mediante un bluetooth dongle de reducido tamaño, nos permitirá conseguir la comunicación que necesitamos con un coste muy reducido tanto económico como de potencia en el dispositivo.

2.4. Sistemas comerciales actuales.

Se realiza a continuación un breve repaso por algunos de los sistemas del mercado actual que ofrecen soluciones dentro del ámbito de este trabajo y que definen el estado del arte en este campo.

2.4.1 Madge Tech Data Loggers

Esta empresa se dedica a fabricar registradores de datos de muchos tipos. Trabaja por separado las distintas variables. Para monitorización de museos y obras de arte, ofrece distintos medidores de temperatura, humedad relativa y presión. Algunos de ellos son capaces de transmitir los datos via Wi-Fi y enviar correos o sms de alerta. Poseen una memoria interna no volátil para garantizar el almacenamiento de los datos.

Permite centralizar varios registradores mediante un PC que ejecuta un software asociado. De esta forma el sistema se puede completar con la adquisición de los datos que nos interesen en cada punto. Ofrece así mismo la posibilidad de añadir repetidores de señal específicos y con ello facilitar la conexión en distancias grandes o cuando existen interferencias u obstáculos que dificulten la comunicación.

Existe la opción adicional del envío los datos a la nube, a través del PC, de forma que sean consultables en un ordenador remoto o en un dispositivo móvil a través de su web.

El hecho de que cada sensor funcione por separado y tenga su propio emisor permite una gran versatilidad para el montaje del sistema y su ubicación, pero implica un coste elevado.

2.4.2. Hanwell Synergy Software

Este sistema, parecido al de Madge Tech, permite la sincronización de varios sensores individuales Hanwell en un solo ordenador. Se trata de un sistema de control de datos basado en navegador, que implementa una base de datos SQL.

Está desarrollado cuidando al detalle la experiencia del usuario final, facilitando la inclusión de nuevos sensores y el acceso a los datos, las alarmas y el historial. Contempla el envío de alarmas por correo y sms. Los sensores son inalámbricos e independientes.

No está específicamente diseñado para obras de arte, pero tiene un abanico de sensores muy amplio que lo hace apto para cualquier aplicación, desde temperatura y humedad, a sensores de presión, detectores de plagas y emisores GPRS.

2.4.3 MHS - Monitoring Heritage System

Este sistema desarrollado por Telefónica y la fundación Santa María la Real, permite monitorizar edificios completos para registrar, evaluar y controlar los parámetros clave en la conservación preventiva del propio edificio y su contenido.

El MHS separa la red de sensores del centro de gestión, de tal forma que puede adaptarse a cada edificio y sus necesidades. Previo a su instalación, se desarrolla una investigación y estudio técnico del edificio, que permite determinar la red de sensores necesaria. Los sensores se comunican con un servidor central en el edificio, que procesa los datos y los envía a una centralita de control y gestión donde se almacenan los datos de todos los proyectos. Los usuarios pueden acceder a toda la información a través de una página web.

Una vez gestionados los datos en la centralita, el sistema puede establecer recomendaciones de acción humana y/o accionar automáticamente dispositivos que corrijan los valores de algunos parámetros.

MHS se instala por un equipo especializado y adaptado a cada edificio. Cubre una gama de parámetros muy extensa, desde temperatura y humedad hasta fisurómetros o detectores de xilófagos. Se trata de proyectos de monitorización exhaustiva y tiene un coste muy elevado.

2.4.4. Conclusiones

Existen tres líneas muy marcadas en los sistemas comerciales. Por un lado están los sistemas más básicos de control de climatización que solo tienen en cuenta la humedad y la temperatura y por otro los grandes sistemas de monitorización exhaustiva donde se vigila hasta el último detalle. Como punto intermedio, tenemos los sistemas de sensores inalámbricos independientes que otorgan flexibilidad y un coste intermedio. En todos los casos, se obliga a disponer de un ordenador central y conexión a internet para la gestión de los datos. Lo más habitual es que el sistema realice una vigilancia global del entorno y no de cada obra de arte en particular.

El proyecto que se presenta, pretende desarrollar una solución de bajo coste con la capacidad de un sistema complejo. La placa Arduino no solo recolecta los datos de su entorno, sino que ejecuta una primera comprobación de los mismos y es capaz de reconocer alertas en primera instancia sin necesidad de conexión externa. Al mismo tiempo, la conexión Bluetooth permite enviarlas a un dispositivo Android que puede encargarse de notificar al personal humano al cargo de la obra, o incluso notificarlo a dispositivos remotos mediante su propia conexión, logrando así las mismas prestaciones que otros sistemas Wi-Fi más complejos.

Referencias

- [1] Documentación sobre la plataforma de desarrollo Arduino. [Consulta 3-6-2015]. Disponible en: <http://www.arduino.cc/>
- [2] GadgetKeeper. Temperature/Humidity monitoring. [Consulta 3-6-2015]. Disponible en: <http://wiki.gadgetkeeper.com/pages/viewpage.action?pageId=10551608/>
- [3] Documentación sobre el sistema Android. [Consulta 3-6-2015]. Disponible en: <http://www.android.com/>
- [4] Documentación sobre iOS. [Consulta 3-6-2015]. Disponible en: <https://developer.apple.com/>
- [5] Documentación sobre el sistema Windows Phone [Consulta 3-6-2015]. Disponible en: <https://www.windowsphone.com/>
- [6] Documentación sobre implementación de Ubuntu touch. [Consulta 3-6-2015]. Disponible en: <http://www.ubuntu.com/phone>
- [7] Documentación técnica de la Tecnología Bluetooth. [Consulta 3-6-2015]. Disponible en: <http://www.bluetooth.com/>
- [8] Wi-Fi Alliance.[Consulta 3-6-2015]. Disponible en: <http://www.wi-fi.org/>
- [9] Madge Tech Data Loggers. [Consulta 3-6-2015]. Disponible en: <http://www.madgetech.com/data-loggers/applications/museums-art-preservation.html>
- [10] Hanwell Synergy Software.[Consulta 3-6-2015]. Disponible en: <http://www.the-imcgroup.com/product/item/hanwell-synergy-software>
- [11] Monitoring Heritage System. [Consulta 3-6-2015]. Disponible en: <http://www.mohst.es/>

Capítulo 3. Mota sensora para monitorización de obras de arte

En este tercer capítulo se describirá de forma conceptual el sistema construido y los elementos que lo componen; Arduino y aplicación Android. Se mostrarán las funciones que realiza, las relaciones entre las dos partes del sistema y las características del flujo de información entre ellos y el usuario.

3.1. Esquema general

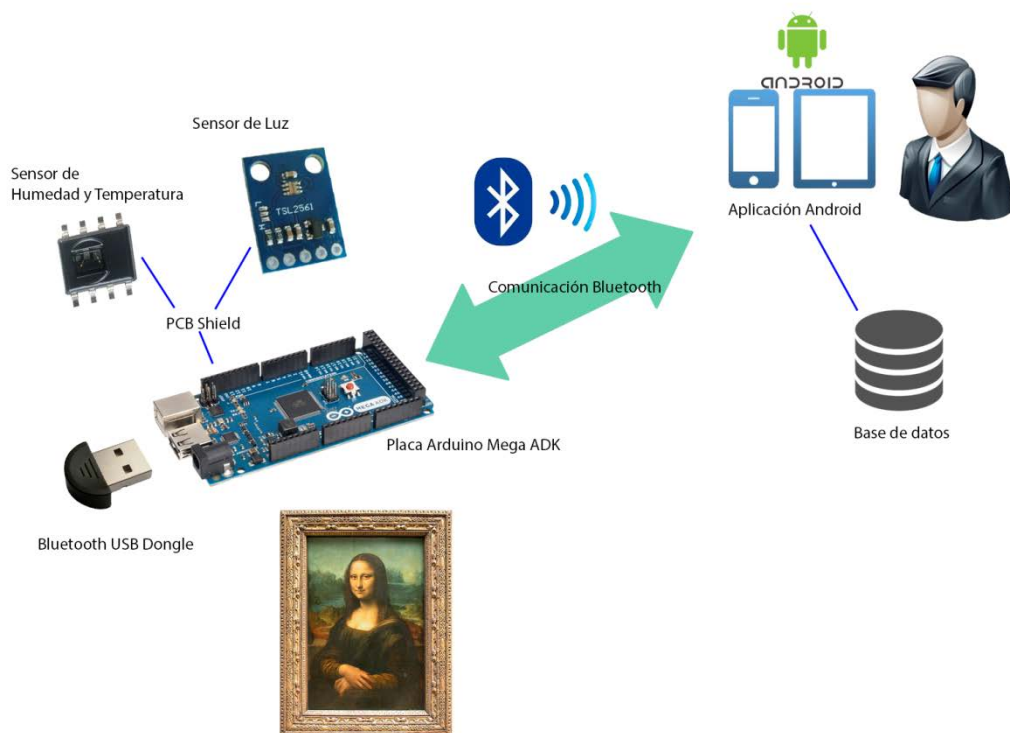


Figura 3.1.1 Esquema general del sistema

En la Figura 3.1.1 se muestra el esquema general del sistema. Se aprecian claramente las dos partes que lo componen. La parte local, basada en la placa Arduino, encargada de interactuar con los sensores y captar las medidas del entorno de la obra de arte. Y una parte remota, consistente en una aplicación Android que recibe, almacena y muestra los datos de forma sencilla e intuitiva para el usuario. Entre ambas partes se establece una intercomunicación inalámbrica mediante tecnología Bluetooth.

Los parámetros que definen el ambiente de la obra de arte: humedad, temperatura y luz, son captados por dos sensores que transmiten la información al microcontrolador de la placa Arduino. El microcontrolador procesa los datos y hace las comprobaciones de

estado antes de mandar la información completa. El envío se realiza a través de un USB dongle instalado en el puerto USB de la placa.

Una aplicación Android, instalada en un dispositivo móvil con una conexión de Bluetooth activa, se encarga de recibir los datos y mostrarlos en tiempo real de una forma cómoda y de fácil interpretación. La información recibida se almacena en una base de datos que permite la consulta de cada parámetro por separado en el momento que se desee, independientemente del estado de la conexión.

3.2. Tecnologías y componentes

En este sistema intervienen diferentes tecnologías que deben funcionar al unísono, se detalla a continuación el modo en que interconexionan entre sí y con los distintos componentes.

Las principales tecnologías en este proyecto son Arduino y Android, si bien Bluetooth también juega un papel importante gestionando la comunicación entre ambas.

3.2.1. Arduino

En el capítulo anterior se hizo mención a las ventajas de la plataforma Arduino. Es abierta, cuenta con un lenguaje y entorno de programación sencillos y dispone de una amplia variedad de placas que se adaptan a las distintas necesidades. Para este proyecto interesan fundamentalmente dos de sus características, estas son:

- Variedad de sensores compatibles y la facilidad de interconexión con los mismos, así como la amplia oferta de librerías existentes gracias a la comunidad de desarrollo Arduino.
- La placa Arduino Mega ADK, que dispone de un conector USB que nos permite disponer de la tecnología Bluetooth mediante un dongle sencillo y económico en lugar de tener que recurrir a un módulo Bluetooth específico.

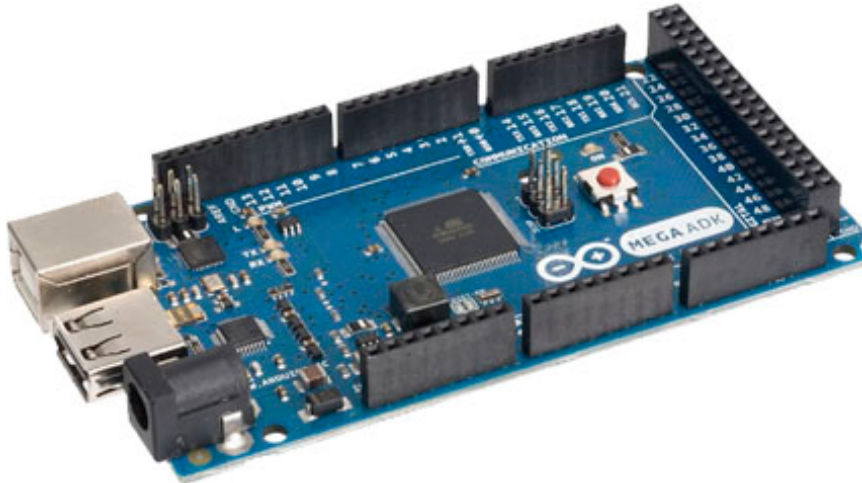


Figura 3.2.1: Placa Arduino Mega ADK

La placa cuenta con un microcontrolador ATmega2560 y diversas entradas analógicas y digitales. Especialmente interesantes resultan las entradas SDA y SCL para tecnología I²C mediante la librería Wire, dado que nos permiten una interconexión muy sencilla de los sensores adecuados.

El microcontrolador que incluye la placa será el encargado de recoger los datos de los sensores y llevar a cabo un control de los mismos antes de empaquetarlos y enviarlos junto con su análisis, a través del USB dongle.

Estas características hacen que dispositivo posibilite cubrir ampliamente los objetivos marcados en el primer capítulo.

3.2.2. Android

En los objetivos se indicaba que el sistema debía disponer de un dispositivo móvil capaz de mostrar y almacenar los datos de forma sencilla y accesible al mayor número de usuarios. Para ello, la mejor opción disponible en el mercado es Android. Se trata de una plataforma ampliamente utilizada, versátil y de fácil desarrollo. Los dispositivos Android tienen un mayor porcentaje de usuarios que otras plataformas y una amplia comunidad de desarrollo.

La aplicación Android desarrollada para este proyecto se ha denominado ArtSense y se detallaran sus funcionalidades más adelante.

3.2.3. Bluetooth

En el Capítulo 2 se realizó una breve comparación entre los sistemas inalámbricos disponibles para dispositivos móviles Android. La conveniencia de evitar una relación de dependencia con redes de terceros excluía la tecnología Wi-Fi limitando las posibilidades

a Bluetooth y NFC. Dado que este último, debido a su reducido alcance no constituye una verdadera opción, la elección es clara.

Bluetooth ofrece como ventajas un bajo consumo, una tasa de transmisión que sobrepasa con un amplio margen las necesidades previsibles y una gran facilidad de implementación: su total disponibilidad en dispositivos móviles, junto a la existencia de los llamados USB dongles, dispositivos apenas más grandes que el propio conector USB que permiten dotar de funcionalidad Bluetooth a casi cualquier dispositivo con un puerto USB disponible.

3.2.4. Sensores

Como se ha mencionado previamente, existe una amplia gama de sensores diseñados específicamente para Arduino, tecnología que además es compatible con otros muchos más que no están específicamente diseñados para ella.

Para lograr los objetivos de este proyecto es preciso monitorizar tres parámetros ambientales clave: humedad, temperatura y luz ambiente. Como condición previa se requiere que los sensores no sean invasivos y que tomen las correspondientes medidas sin interferir en el entorno, de forma que siempre que sea necesario los parámetros monitorizados puedan ser corregidos con el uso de otras tecnologías externas.

Finalmente se ha optado por incluir en el proyecto un sensor conjunto de temperatura y humedad, dado que estos dos parámetros están muy relacionados, y un sensor de luz propio de Arduino. Ambos sensores digitales permiten la comunicación I²C de los datos, lo cual nos permite trabajar con ambos simultáneamente y sin problemas de compatibilidad.

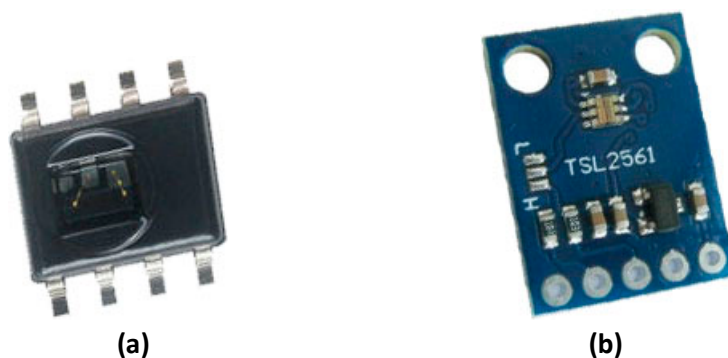


Figura 3.2.2: (a) Sensor de humedad y temperatura HIH60XX; (b) Sensor de luz TSL2561

3.3. Flujo de la información

El flujo de información mayoritario, compuesto por los datos de temperatura, humedad y luz viajará en un solo sentido. La información es captada por los sensores y se solicita periódicamente desde el microcontrolador de la placa Arduino. Tras recibir la información de los sensores y procesarla, se transmite mediante Bluetooth al dispositivo Android conectado.

La aplicación Android recibe los datos y realiza un nuevo procesamiento para mostrar y almacenar los datos de manera clara y con una fácil interpretación, junto con la información de fecha y hora del momento en el que se reciben. De esta forma se puede acceder de forma intuitiva y sencilla a los datos almacenados siempre que se desee.

Así pues, los datos capturados viajan en un solo sentido. Sin embargo, vemos en la figura 3.1.1 que la comunicación también puede realizarse desde la aplicación Android a la placa Arduino. ArtSense se permite que el usuario modifique ciertas variables del programa de medida: Intervalo entre medidas, Luz máxima y valores centrales de humedad y temperatura, parámetros que marcaran los límites establecidos para el control del ambiente y que podrían personalizarse en función de las condiciones deseadas para el óptimo mantenimiento de cada obra en particular.

3.4. Funcionalidades del sistema

La mota sensora para monitorización de obras de arte desarrollada en este proyecto tiene como fin ofrecer una forma sencilla y económica para monitorizar el entorno de la obra de arte, facilitando además su control para la óptima conservación de la misma. El reducido tamaño de la placa Arduino permite llevar a cabo la tarea sin suponer una agresión visual ni interferir de manera alguna en el entorno de exposición.

Explicaremos por separado las funcionalidades de las dos partes constituyentes de este proyecto y que en conjunto permiten el adecuado cumplimiento de los objetivos del mismo.

3.4.1. Funcionalidades de la placa Arduino

En el mismo momento en el que se conecta la alimentación, el microcontrolador se prepara para recibir una petición de conexión Bluetooth y comienza a tomar datos con los ajustes preestablecidos por defecto, realizando las siguientes funciones cada vez que se cumple el intervalo de tiempo entre medidas:

1. Lectura de los datos de cada sensor.

2. Adecuación de los datos y conversión al formato correspondiente en unidades de medida del sistema internacional mediante las librerías de cada sensor.
3. Procesamiento y control: Se realiza una comprobación de cada uno de los datos para saber si está dentro del margen de seguridad establecido por la normativa:
 - Temperatura entre 17 °C y 27 °C con una variación máxima de ± 2 °C.
 - Humedad relativa entre 30 % y 70% con una variación máxima de $\pm 5\%$.
 - Luminosidad máxima establecida según el tipo de obra de arte.

En caso de existir conexión Bluetooth se procede a los puntos 4 y 5.

4. Formación de una trama de transmisión conformada por identificador de medida, valor y estado según los límites establecidos.
5. Envío de las tramas a través del USB dongle.
6. A las 24h de funcionamiento, se realiza una revisión de los valores centrales que establecen los límites de los parámetros.

Todo esto no sería posible sin las funciones de cada uno de los sensores. Se trata de sensores digitales con conectividad I²C que funcionan como esclavos en el mismo bus de datos y envían directamente al microcontrolador los valores de sus respectivos parámetros bajo petición. Ambos se conectan a la placa Arduino mediante una placa PCB diseñada a tal efecto.

La alimentación del prototipo se realiza mediante el conector USB tipo B del que dispone la placa Arduino.

3.4.2. Funcionalidades de la aplicación Android: ArtSense

En cualquier momento se puede conectar el dispositivo móvil al sistema, accediendo desde ese momento a los datos que se toman y posibilitando el almacenamiento de los mismos. La aplicación presenta una pantalla central desde donde directamente se puede acceder a todas las funcionalidades de las que dispone. Puede verse en la Figura 3.4.1.

Pulsando el botón de conexión, se accederá a una lista de dispositivos disponibles, tras elegir el dispositivo Arduino se solicita una contraseña para evitar el acceso de usuarios no autorizados.

Una vez establecida la conexión se muestra en pantalla el valor de los datos en tiempo real. Acorde con dichos valor, el color del icono correspondiente a cada parámetro cambia para hacer aún más intuitiva la lectura de los datos: Rojo para un valor

demasiado alto, azul para un valor demasiado bajo, y verde si el valor está dentro del rango permitido.

En segundo nivel tenemos tres botones que nos permiten acceder al historial de datos guardado para cada valor. Esta función está disponible independientemente del estado de conexión.

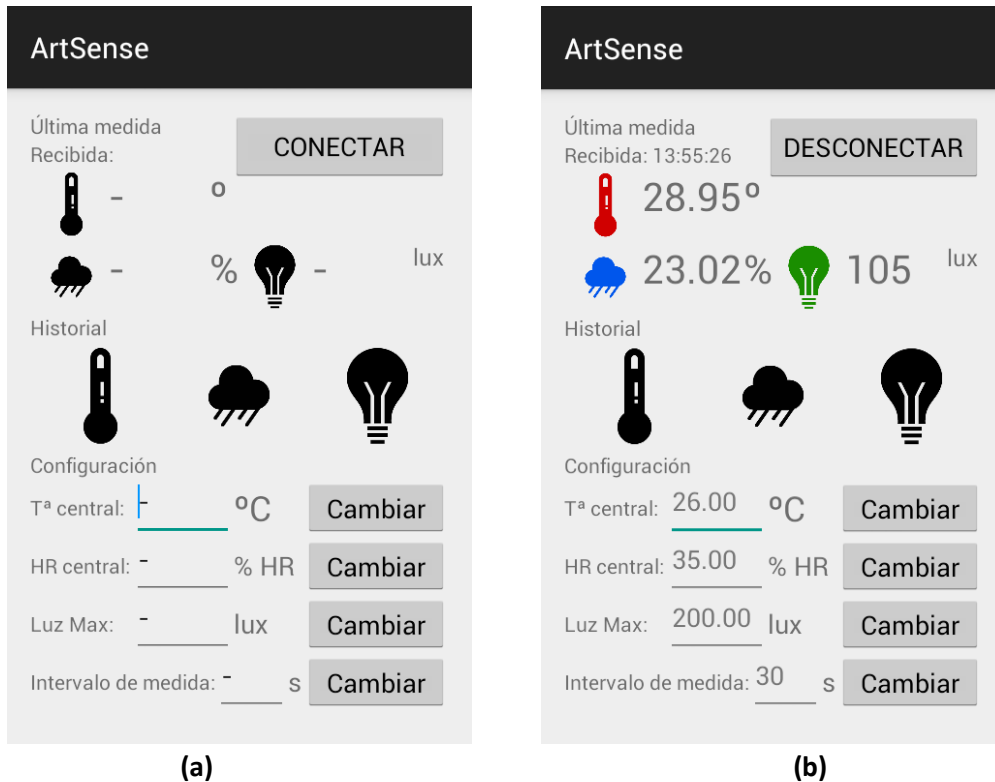


Figura 3.4.1: Pantalla principal de ArtSense. (a) Estado desconectado; (b) Estado conectado

En la Figura 3.4.2. se muestra uno de los historiales con algunas medidas. En las tablas de historial se utiliza el mismo código de colores que en la pantalla principal. Rojo para valores excesivos y azul para valores demasiado bajos. En este caso se prescinde del color verde para los valores adecuados por tener una lectura más incomoda que el color negro.

Por último, tenemos la zona más baja de la pantalla principal reservada para los valores de configuración. Aquí se muestran los valores que marcan los límites de control de los distintos parámetros. Dichos valores se envían desde Arduino al comenzar la conexión, y pueden variarse desde la aplicación.

En caso de desconexión por cualquier motivo, todos los iconos de la pantalla principal de la aplicación vuelven a color negro y dejan de mostrarse las medidas. De forma que el usuario pueda darse cuenta rápidamente de la situación.



FECHA	HORA	VALOR (lux)
2015-06-08	14:06:05	122
2015-06-08	14:05:35	122
2015-06-08	14:05:04	121
2015-06-08	14:04:34	121
2015-06-08	14:04:03	120
2015-06-08	14:03:33	119
2015-06-08	14:03:02	121
2015-06-08	14:02:32	119
2015-06-08	14:02:02	118
2015-06-08	14:01:31	117
2015-06-08	14:01:01	118
2015-06-08	14:00:30	119
2015-06-08	14:00:00	117
2015-06-08	13:59:29	118
2015-06-08	13:58:59	117
2015-06-08	13:58:29	119
2015-06-08	13:57:58	115
2015-06-08	13:57:28	113
2015-06-08	13:56:57	1874
2015-06-08	13:56:27	1876
2015-06-08	13:55:56	107
2015-06-08	13:55:26	105

Figura 3.4.2: Historial de Luz

Referencias

- [1] Documentación sobre la plataforma de desarrollo Arduino. [Consulta 8-6-2015]. Disponible en: <http://www.arduino.cc/>
- [2] Documentación técnica de la Tecnología Bluetooth. [Consulta 8-6-2015]. Disponible en: <http://www.bluetooth.com/>
- [3] Datasheet HIH6130-021-001 Humidity/Temperature Sensors. Honeywell International Inc., Golden Valley, MN, 2013.
- [4] Datasheet TSL2561 Light-to-Digital Converter. Texas Advanced Optoelectronic Solutions Inc., Plano, TX, 2005.
- [5] Documentación técnica de la tecnología I²C. [Consulta 8-6-2015]. Disponible en: <http://www.i2c-bus.org/>

Parte II

Proceso de desarrollo

Capítulo 4 - Estudio de requisitos y planificación

Un estudio previo de los requisitos que tendrá el sistema que se va a desarrollar es imprescindible para identificar los puntos clave que debe cumplir para alcanzar los objetivos planteados. Si el estudio se lleva a cabo de forma detallada, permitirá disminuir los imprevistos y realizar una buena planificación que facilite y aligere el desarrollo del proyecto.

4.1. Especificación de requisitos

En el primer Capítulo se enumeraron una serie de requisitos de alto nivel que definen los requisitos funcionales y no funcionales que ha de cumplir la mota sensora. Dividiremos el estudio en requisitos funcionales y no funcionales. Delimitando los primeros las funciones que deberá ser capaz de ejecutar el sistema y los segundos el nivel de calidad del mismo.

4.1.1. Requisitos funcionales

1. Placa Arduino y microcontrolador.

- RF 1.1: Debe realizar lecturas periódicas de los sensores a una frecuencia que podrá ser determinada por el usuario. La lectura se realizará mediante las entradas SDA y SCL de I²C.
- RF 1.2: La información generada por los sensores debe ser adaptada y procesada. Se comprobará su estado en relación a los límites fijados y se preparará para permitir su envío al dispositivo móvil.
- RF 1.3: Se conformarán tramas de datos legibles por el dispositivo móvil conteniendo los datos de los parámetros de interés.
- RF 1.4: El envío de las tramas se realizará a través del USB dongle.
- RF 1.5: En caso de no existir conexión, el dispositivo debe seguir llevando a cabo su ciclo de funcionamiento para permitir la gestión de alertas externas.
- RF 1.6: Todo lo anterior se llevará a cabo de forma automática y sin necesidad de que el usuario interactúe con el sistema.

- RF 1.7: La alimentación de la placa debe llevarse a cabo mediante un sistema que ofrezca una tensión fija y constante que permita satisfacer las necesidades de corriente del dispositivo completo.

2. Sensores.

- RF 2.1: Deben realizar medidas correctas de los parámetros correspondientes siempre que las condiciones de funcionamiento sean las adecuadas.
- RF 2.2: Deben realizar las medidas de forma automática y sin necesidad de que el usuario interaccione con el sistema.
- RF 2.3: Con una calibración adecuada deben ofrecer mediciones coherentes.

3. USB Dongle.

- RF 3.1: Debe permitir el envío en orden de las tramas que el microcontrolador de la placa le haga llegar.
- RF 3.2: Cuando esté conectado, ha de mostrarse visible para la aplicación Android y consentir el establecimiento de conexión Bluetooth mediante una contraseña introducida por el usuario. Siempre y cuando las condiciones de visibilidad entre dispositivos lo permitan.

4. Aplicación Android: ArtSense.

- RF 4.1: La aplicación debe permitir la gestión de la conexión y desconexión al modulo Arduino a petición del usuario.
- RF 4.2: La aplicación proporcionara una monitorización en tiempo real siempre y cuando la conexión Bluetooth este activa.
- RF 4.3: El sistema deberá guardar todos los valores recibidos en la base de datos del dispositivo móvil.
- RF 4.4: Se permitirá la consulta de todos los datos almacenados (temperatura, humedad y luz) en cualquier momento, independientemente del estado de la conexión.

4.1.2. Requisitos no funcionales**1. Placa Arduino y microcontrolador.**

- RNF 1.1: El sistema debe ser escalable de forma que se puedan añadir nuevos sensores o funcionalidades. Los nuevos añadidos no deben afectar al rendimiento ni al funcionamiento del sistema.
- RNF 1.2: Una vez procesados los valores de los parámetros deben enviarse vía Bluetooth con tal frecuencia que la monitorización en tiempo real resulte efectiva.
- RNF 1.3: Se debe implementar un protocolo de comunicación con el dispositivo remoto basado en tramas de datos y sincronización que sea capaz de soportar fallos sin afectar a todo el sistema.
- RNF 1.4: El usuario no debe ocuparse en ningún momento del funcionamiento o configuración directa de la placa. Únicamente deberá conectar la alimentación y ubicar el sistema en un lugar adecuado.

2. Sensores.

- RNF 2.1: Deben permitir en todo momento el acceso a la medida. Siempre que se respeten los tiempos de acceso de los mismos.

3. USB Dongle.

- RNF 3.1: El envío de tramas debe realizarse en el menor tiempo posible siempre que el dispositivo móvil se encuentre conectado.

4. Aplicación Android: ArtSense.

- RNF 4.1: Debe ser escalable, de forma que sea posible añadir nuevas funcionalidades de forma sencilla.
- RNF 4.2: La interfaz de usuario debe ser sencilla, clara e intuitiva.
- RNF 4.3: Debe proporcionar los datos de una manera rápida, precisa y de fácil interpretación
- RNF 4.4: Al ejecutarse sobre un dispositivo móvil, la aplicación debe consumir el mínimo posible de recursos (memoria, CPU y batería).

4.2. Planificación

Afrontar un proyecto de este tipo en solitario, tras haber pasado la mayor parte de la carrera trabajando en equipo supone un reto de planificación. Es necesario saber gestionar el tiempo propio y marcarse objetivos a corto plazo para que el trabajo global pueda llevarse a cabo dentro del plazo adecuado.

4.2.1. Planificación previa

El calendario previo de planificación tiene que realizarse teniendo muy presentes los imprevistos que puedan surgir a lo largo del proceso, un calendario excesivamente ajustado que no considere la posibilidad de retraso, demoras o imprevistos de cualquier índole, puede hacer fracasar un proyecto, dado que podría no ser posible finalizarlo antes de la fecha de entrega. En el caso que nos ocupa, la planificación inicial fue muy optimista. Sin embargo, acababa con una antelación suficiente a la fecha tope de entrega que permitía un margen de seguridad suficiente para cubrir posibles imprevistos. El diagrama de Gantt [1] puede verse en la Figura 4.2.1.

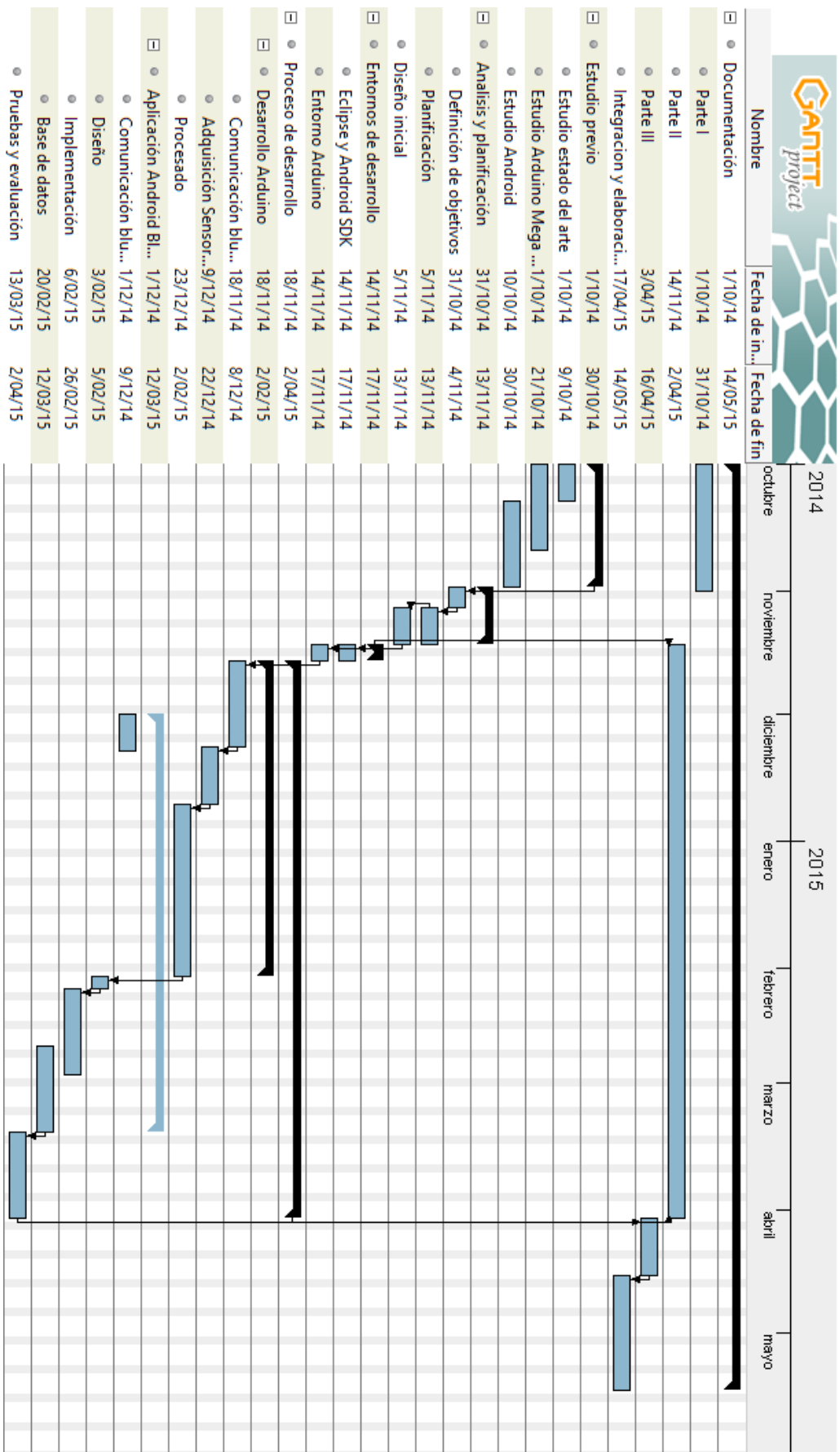


Figura 4.2.1: Diagrama de Gantt de la planificación a priori

4.2.2. Transcurso del proyecto

Este apartado refleja el desarrollo real del proyecto y detalla los acontecimientos que se produjeron durante el mismo. En particular se detallan las dificultades e imprevistos que surgieron durante el transcurso del proyecto y como se han ido solucionando.

4.2.2.1. Primeros pasos

Para comenzar se realizó un estudio sobre el estado del arte. Concretamente se consideró la necesidad de conocer en detalle el desarrollo de las normativa nacional e internacional relacionada con el proyecto y las distintas alternativas tecnológicas existentes en el mercado que pudieran aportar soluciones similares a la aquí planteada. En vista a los conocimientos obtenidos, se fijaron los objetivos generales que el sistema debía cumplir y se consultó la documentación de las distintas plataformas que podían servir para llevarlos a buen término.

Durante una fase avanzada del estudio de posibilidades, cuando Android y Arduino eran la opción más fuerte, se instalaron los entornos de desarrollo para tener una primera toma de contacto y comprobar de primera mano las bondades descritas en la documentación. En un primer momento se instalaron los siguientes entornos de desarrollo:

- Eclipse Indigo + Android SDK: Un paquete conjunto, disponible en la web de Android que permitía el desarrollo y prueba en emulador de aplicaciones para Android. Ambos de código abierto. [2]
- Entorno Arduino: Un software sencillo que permite escribir código en lenguaje Arduino y cargarlo en las distintas placas del fabricante.

4.2.2.2. Elección de sensores

Se ha mencionado en capítulos anteriores la gran variedad de sensores compatibles con Arduino. La elección del protocolo I²C como vía de comunicación acotaba un poco el espectro. Encontrar un sensor de humedad y temperatura que cumpliera los requisitos de funcionamiento y precisión y que contara con dicho protocolo fue una tarea sencilla. El caso del sensor de luz fue más complicado. Los proveedores con los que trabaja la universidad tenían una menor oferta para este parámetro. Finalmente se optó por el BH1710FVC-TR.

El tamaño de los sensores no era uno de los factores importantes en la elección de los mismos, así que no se tuvo especialmente en cuenta. Fallo que resultó crítico más adelante, como se explica en el apartado 4.2.2.5.

4.2.2.3. Toma de contacto con Arduino

La plataforma Arduino no me era totalmente desconocida, si bien había desarrollado poco trabajo en la misma. Para afrontar el proyecto, se comenzó desde cero, probando algunos de los ejemplos e introduciendo ligeras modificaciones.

El siguiente paso fue hacer pruebas con la librería Bluetooth que permitía la interconexión de la placa mediante un USB dongle. La librería no está completamente depurada y hay una amplia oferta de dongles en el mercado, haciendo muy difícil conseguir compatibilidad con todos los modelos. Las especificaciones que dan los fabricantes con respecto a la mayoría de los dongles son escasas, por lo que fue difícil encontrar el modelo adecuado.

Las pruebas se realizaron con la aplicación Android de código libre Blueterm, que permite emular un terminal y enviar los datos directamente en código ASCII. [3]

Como añadido, los usb dongle no se comportan igual con todos los dispositivos móviles, por lo que finalmente se ha desarrollado el proyecto con un Samsung Galaxy S3 mini que presentaba una comunicación óptima.

4.2.2.4. Toma de contacto con Android

Existe una página web para desarrolladores Android que cuenta con un completo entrenamiento para nuevos desarrolladores [4]. Gracias a esta guía, los primeros pasos con Android son muy sencillos y permiten realizar fácilmente aplicaciones sencillas. Con esto finalizo la breve toma de contacto con la plataforma móvil, que en las primeras fases del desarrollo no planteaba mayores exigencias.

4.2.2.5 Diseño y desarrollo de la PCB

Los dos sensores elegidos eran de reducido tamaño, especialmente el de luz. Por ello fue necesario el diseño de una PCB para llevar a cabo el conexionado de los mismos. Se utilizó la plataforma abierta KiCad que permite el desarrollo de esquemáticos y diseño de PCB de manera sencilla. Al trabajar tan solo con dos sensores fue posible hacer el diseño en una sola capa, recurriendo a las distintas patillas de tierra de las que dispone la placa Arduino para conectar las distintas áreas de la placa.

El primer problema apareció al intentar soldar los componentes a la placa. Ambos sensores eran componentes de tipo SMD, cuya soldadura manual requiere una mayor destreza. En el caso del sensor de luz fue especialmente difícil y posteriormente a su soldadura fue imposible hacerlo funcionar. Es difícil decir si el sensor se deterioró durante los diversos intentos o finalmente no se consiguió un conexionado correcto.

Este acontecimiento obligo a dar un paso atrás y volver a la elección de sensor. Finalmente se decidió recurrir a un proveedor externo y adquirir un sensor de luz ofertado para la plataforma Arduino. El TSL2561 se trata en realidad de una pequeña PCB que conecta un pequeño sensor de luz muy similar al elegido anteriormente junto con los componentes necesarios para adecuación del sensor y la comunicación I²C de forma que solamente es necesario conectar las patillas al punto correspondiente. Resultando mucho más sencillo su conexión dado su mayor tamaño.

Finalmente resultó las ventajas de este nuevo sensor sobrepasaron ampliamente al inconveniente generado por el cambio de la previsión inicial. Se trata de un sensor muy utilizado con Arduino que dispone incluso de una librería para el mismo, facilitando el trabajo posterior de adquisición de los datos. Este es un magnífico ejemplo de los problemas que pueden evitarse cuidando los detalles del estudio previo del proyecto.

4.2.2.6 Adquisición y procesamiento de los datos

Con ambos sensores acompañados de su librería correspondiente, junto con la propia librería Wire de Android que gestiona el protocolo I²C, la adquisición de datos fue una tarea sencilla.

Durante los periodos de espera que se dieron con la adquisición de nuevas piezas como el dongle y el segundo sensor de luz, se hicieron algunos avances con el procesamiento. Partiendo de una variable que almacenaría la futura medida, se diseñó e implementó el sistema cíclico de medidas y comprobaciones en base a un intervalo. Este trabajo previo junto con la parte de adecuación del dato a sistema internacional realizado por las propias librerías, hizo que la adquisición y procesamiento de datos se llevaran a cabo sin problema desde el momento que se conectaron ambos sensores.

4.2.2.7 Desarrollo de la aplicación ArtSense

Los periodos de espera también sirvieron para avanzar en el desarrollo de la aplicación. Desde un principio el trabajo se llevó a cabo mediante la aplicación Blueterm, que permite emular un terminal para enviar y recibir texto desde Android. Se desarrollaron una serie de comandos a los que el Arduino era capaz de responder con el dato adecuado.

Al ser Blueterm un proyecto de código abierto, existía la posibilidad de crear una nueva aplicación con una capa de personalización adecuada al proyecto que aprovechara las funcionalidades de Blueterm para la conexión Bluetooth. En la fase temprana del proyecto el ordenador principal donde se desarrollaba sufrió un duro revés y fue necesario un formateo completo. Aunque no se perdieron los avances desarrollados gracias a las copias de seguridad y el repositorio Git, sí fue necesario reinstalar las herramientas y entornos de desarrollo. Este problema coincidió en el tiempo con el reconocimiento y apoyo de Google a la plataforma Android Studio como entorno de

desarrollo oficial para Android. Así pues, se decidió realizar la mudanza desde Eclipse a la nueva herramienta.

Fue necesario familiarizarse con la nueva plataforma, así como limar ciertas asperezas para conseguir que proyectos desarrollados en Eclipse pudieran modificarse en Android Studio. Una vez salvados estos problemas entre versiones, fue cuestión de limpiar el código de Blueterm eliminando las funcionalidades que no eran necesarias y añadir la capa de personalización que estaba a medio desarrollar.

4.2.2.8. Integración de plataformas y finalización.

Una vez integradas las dos plataformas, es esfuerzo se concentró en el pulido de los detalles de su funcionamiento. Para una cuidada unión de la nueva aplicación con la parte hardware de Arduino, fue necesario crear un protocolo que permitiera el intercambio de datos correctamente. Finalmente se modificaron los comandos que se habían implementado en origen, que obligaban al usuario a solicitar los datos y se modificaron por envíos automáticos realizados al unísono de la toma de medidas.

Por último se abordó la implementación de la base de datos, que salvo algunos detalles concretos, fue bastante sencilla gracias a la comunidad de desarrollo Android y los tutoriales disponibles para el aprendizaje.

Tan solo restaba realizar un intervalo de pruebas y evaluación de resultados, en el que entraremos en detalle más adelante.

4.2.3. Reorganizaciones de la planificación

Finalmente se ha recogido en el diagrama de Gantt de la Figura 4.2.2, el calendario con las reorganizaciones sufridas a lo largo del desarrollo del proyecto. Una planificación inicial excesivamente optimista junto con los cambios obligados en el material del proyecto, implicaron retrasos en muchas de las tareas y reorganización de otras, de forma que pudiera aprovecharse los intervalos de tiempo de espera para la recepción de materiales que obligaban a paralizar la ejecución de algunas actividades. La flexibilidad en la planificación y la ejecución en paralelo de algunas de las tareas inicialmente previstas ha sido crucial para el desarrollo a tiempo del presente proyecto.

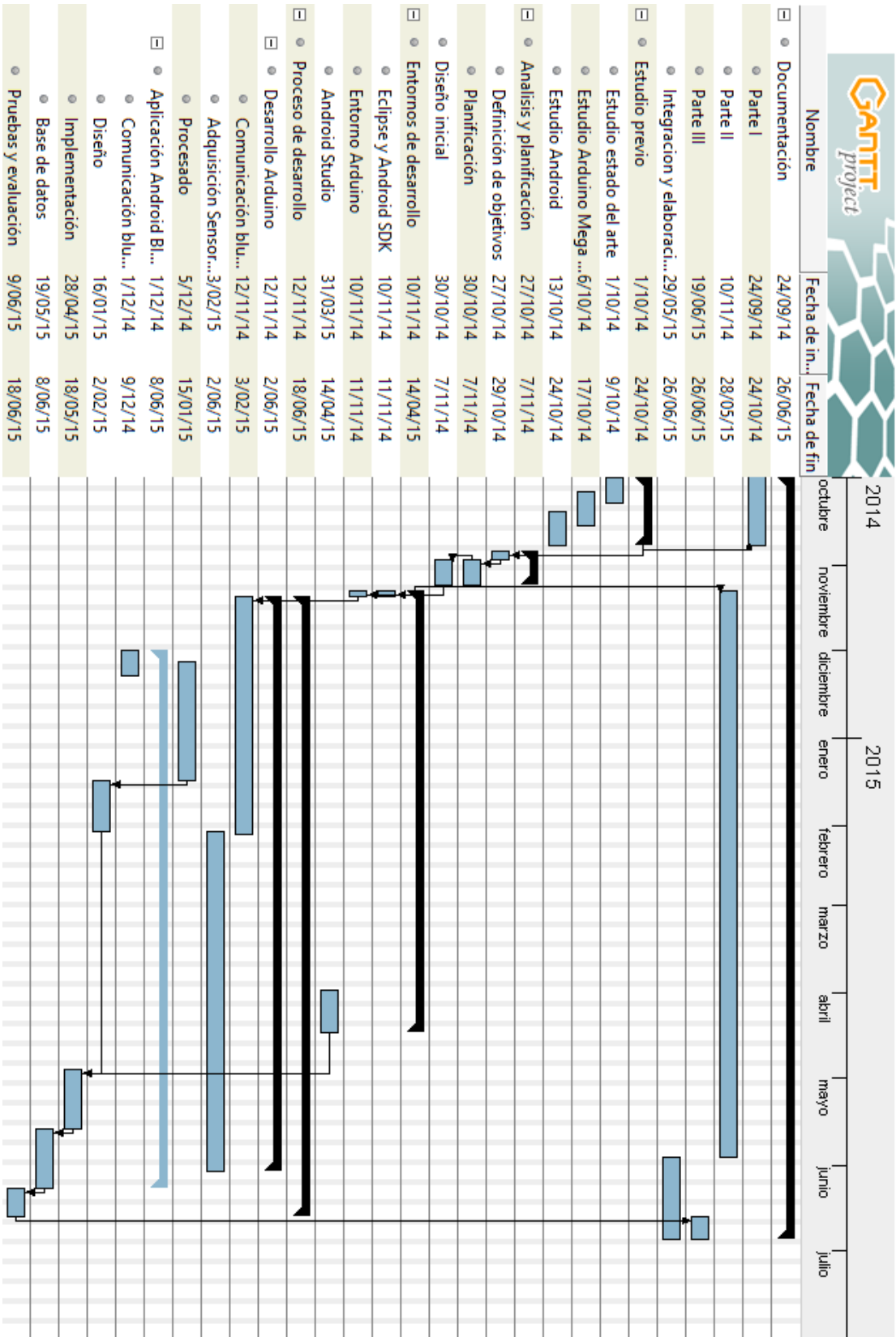


Figura 4.2.2: Diagrama de Gantt de la planificación a posteriori

Referencias

[1] Gantt chart. Artículo en Wikipedia. [Consulta: 9-06-2015]. Disponible en: http://en.wikipedia.org/wiki/Gantt_chart

[2] Información sobre la plataforma Eclipse. [Consulta: 9-06-2015]. Disponible en: <https://eclipse.org/>

[3] Aplicación Blueterm en Google play. [Consulta: 9-06-2015]. Disponible en: <https://play.google.com/store/apps/details?id=es.pymasde.blueterm&hl=es>

[4] Android developers. [Consulta: 9-06-2015]. Disponible en: <https://developer.android.com/index.html>

Capítulo 5. Diseño e implementación

El siguiente consistió en el diseño del sistema, eligiendo entre las distintas alternativas posibles aquella que permitiera garantizar la respuesta más satisfactoria a los distintos objetivos marcados. El sistema se realizará de forma acorde a las decisiones que se tomen en la fase de diseño sobre su estructura, interconexión de las partes que lo componen y funcionalidades de las mismas. Solo cuando se disponga de un diseño consistente, se podrá proceder a la implementación del mismo.

En este capítulo se detallaran los aspectos claves y consideraciones de diseño para cada uno de los componentes de la mota sensora, junto con el diseño de la aplicación ArtSense y la integración del sistema en su conjunto.

5.1. Diseño de la mota sensora

La mota sensora está compuesta por una placa Arduino Mega ADK, que será alimentada por el puerto USB tipo B. A las entradas correspondientes al protocolo I²C [1] se conectan dos sensores que captarán tres variables del entorno: Uno de ellos tomará medidas de temperatura y humedad y el otro se encargara de tomar medidas de la luz ambiente. Los sensores se conectan a la placa por medio de una PCB que incluye los componentes necesarios para su correcto funcionamiento. La comunicación con el dispositivo móvil se realiza mediante tecnología Bluetooth a través de un USB dongle conectado al puerto USB tipo A. El esquema general de la mota se muestra en la Figura 5.1.1.

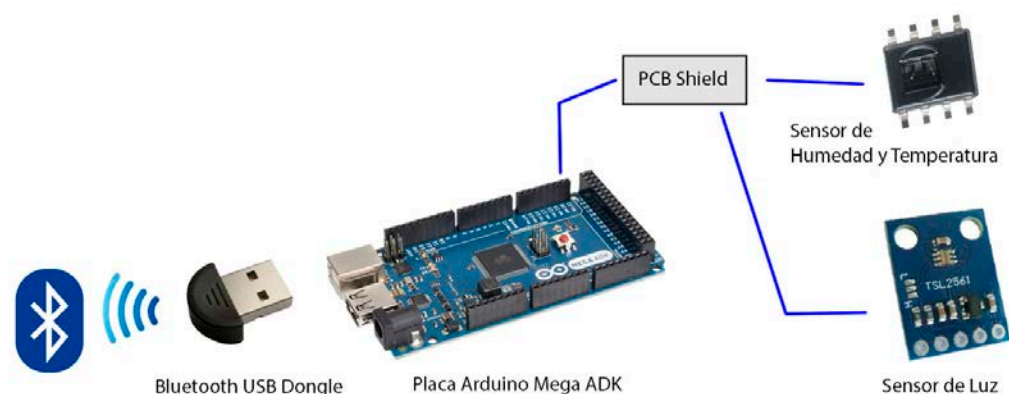


Figura 5.1.1: Esquema general de la mota sensora.

El dispositivo se sitúa próximo a la obra de arte que se desea controlar, permitiendo la monitorización de los parámetros de temperatura, humedad e iluminación del

ambiente. El conjunto de los componentes puede encapsularse en una caja de diseño discreto y no más de 11x5,5x2cm de forma que no suponga una invasión visual excesiva en el entorno de exposición de la obra.

5.1.1. Sensor de luz

El TSL2561 es un convertidor de luz, capaz de transformar la intensidad de luz a una señal digital I²C [2]. Combina un fotodiodo de banda ancha (luz visible e infrarroja) con un fotodiodo de respuesta infrarroja. El sensor en sí tiene unas medidas ínfimas de 3,8x2,60x0,7mm, con tecnología de montaje SMD, lo cual dificulta el trabajo de implementación sin las herramientas adecuadas. Sin embargo, existen proveedores que sirven el sensor junto con los componentes necesarios para su interconexión, previamente montado sobre una pequeña PCB de adaptación a tecnología DIP para facilitar su integración en sistemas Arduino. Ver Figura 5.1.2.

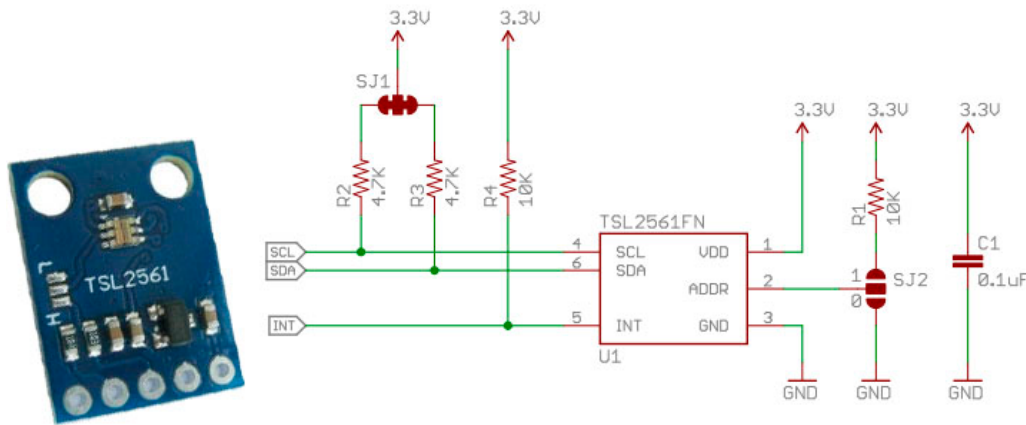


Figura 5.1.2: Sensor de luz TSL2561 integrado en PCB y esquemático de la misma.

El sensor cubre el rango espectral entre los 300 y 1000 nm, utilizando ambos fotodiodos para aproximar la respuesta del ojo humano, que se mueve en el rango entre 400 y 700 nm. Por tanto el sensor es completamente capaz de captar todo el espectro visible, y parte del infrarrojo, que puede ser utilizado en ampliaciones futuras, puesto que también influye en la conservación de la obra de arte.

Es importante tener en cuenta la situación del sensor si se realiza el empaquetado del sistema, puesto que debe estar libre de obstáculos y preferiblemente en la misma orientación que la obra a monitorizar, de forma que pueda captar la luz incidente. La respuesta del sensor varía según el ángulo de incidencia como se ve en la Figura 5.1.3.b.

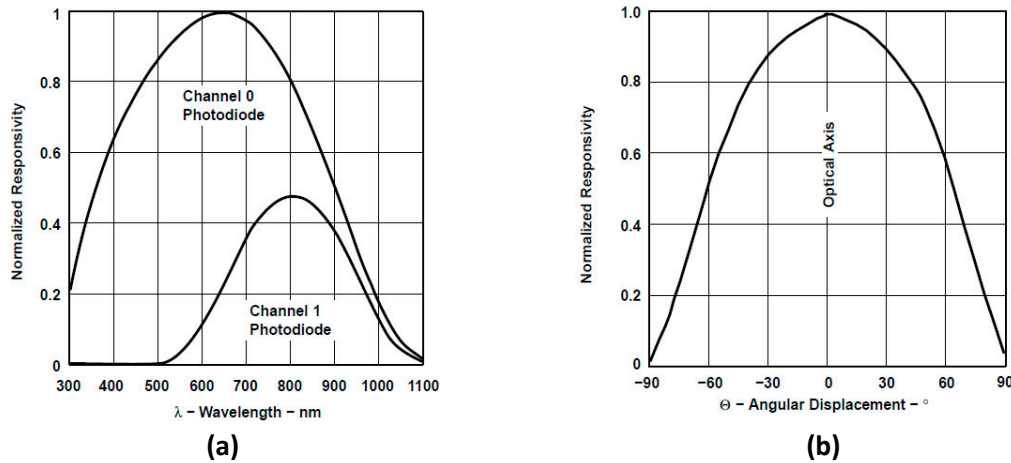


Figura 5.1.3: (a) Respuesta del sensor de luz en el espectro; (b) Respuesta del sensor de luz según el ángulo de incidencia.

El sensor requiere una alimentación entre 2,7 y 3,6 V, por lo que podemos alimentarlo sin problema con el pin de 3,3V del que dispone la placa Arduino. Las conexiones de datos se realizan a las líneas SCL y SDA preparadas para I²C.

5.1.2. Sensor de temperatura y humedad

En este caso, el sensor utilizado es el HIH6130 [3], se trata en realidad de dos sensores con salida digital combinados en el mismo empaquetado. Se utiliza el sensor de temperatura para incrementar la precisión del sensor de humedad hasta $\pm 4\%$ HR y se permite la lectura individual de temperatura para incrementar las funcionalidades del sensor. Permite utilizar I²C o SPI para la lectura de la salida.



Figura 5.1.4. Sensor de humedad y temperatura HIH6130

Como en el caso anterior, este sensor también utiliza tecnología de montaje SMD. Sin embargo tiene unas dimensiones más asequibles para su soldadura manual, 4,90x3,90x2,05mm. Por ello en este caso no fue necesario recurrir a sensores específicos para Arduino, que tienen un mayor precio.

El sensor está preparado para trabajar en óptimas condiciones entre 5 y 50 °C con una precisión de ± 1 °C y entre 10 y 90 % HR con la ya mencionada precisión de $\pm 4\%$ HR. Esto cubre ampliamente nuestras necesidades, puesto que las condiciones que vamos a

medir deben encontrarse entre 17 y 27 °C con una variación máxima de ± 2.5 °C en el caso de la temperatura, y entre 30 y 70% HR con una variación máxima de ± 5 -10 HR en el caso de la humedad. [4]

Ha sido necesario preparar el circuito de adaptación del sensor, que se implementará en una PCB siguiendo las especificaciones del fabricante. El conexionado típico para el uso del sensor mediante I²C puede verse en la Figura 5.1.5.

Este sensor permite un rango de alimentación entre 2,3 y 5,5 V más amplio que el caso anterior. La patilla de tensión 3,3 V del Arduino es perfecta, dado que corresponde a la tensión típica recomendada por el fabricante.

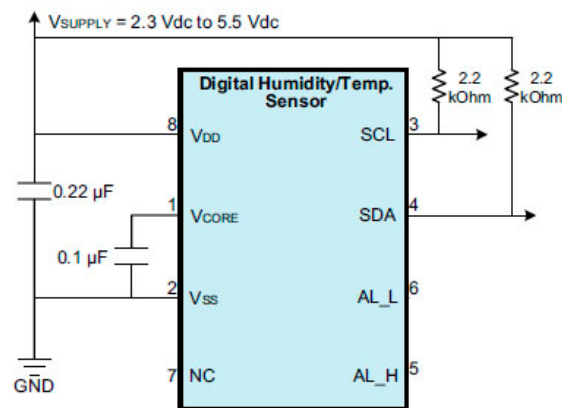


Figura 5.1.5: Conexionado del sensor HIH6130 para su uso con I²C

5.1.3. Placa Arduino Mega ADK

En la placa se integra el microcontrolador ATmega2560 [5], que será el encargado de llevar a cabo todo el procesamiento de información, construcción de tramas con los datos tomados y de la sincronización y envío a través del Bluetooth USB dongle. Es el punto principal en el que se interconexiónan todos los elementos de la mota sensora. Las especificaciones que mayor interés y repercusión tienen para el desarrollo del presente proyecto son las siguientes:

- Alto rendimiento con bajo consumo, 8 Bits
- 256 KBytes de memoria flash programable
- 16 MHz de velocidad de reloj
- Voltaje de operación 5V
- Interfaz I²C. Utilizada para comunicar los sensores
- USB Host Chip, que nos permitirá el uso del Bluetooth USB dongle.

La placa dispone de un total de 54 pines de entrada/salida digital y otros 16 de entrada analógica. Para el desarrollo de este proyecto tan solo necesitaremos los pines digitales correspondientes al protocolo I²C, número 20 para SDA y 21 para SCL. Usaremos

también dos de los pines de tierra de los que dispone, uno a cada lado de la placa y el pin que suministra 3,3V regulados desde la placa.

5.1.4. PCB de interconexión

La conexión de dichos pines con los sensores, se realizará mediante una PCB con el diseño mostrado en la Figura 5.1.6. Nótese que el USB dongle no aparece en el esquema dado que se conecta al puerto USB tipo A del que dispone la placa Arduino Mega ADK.

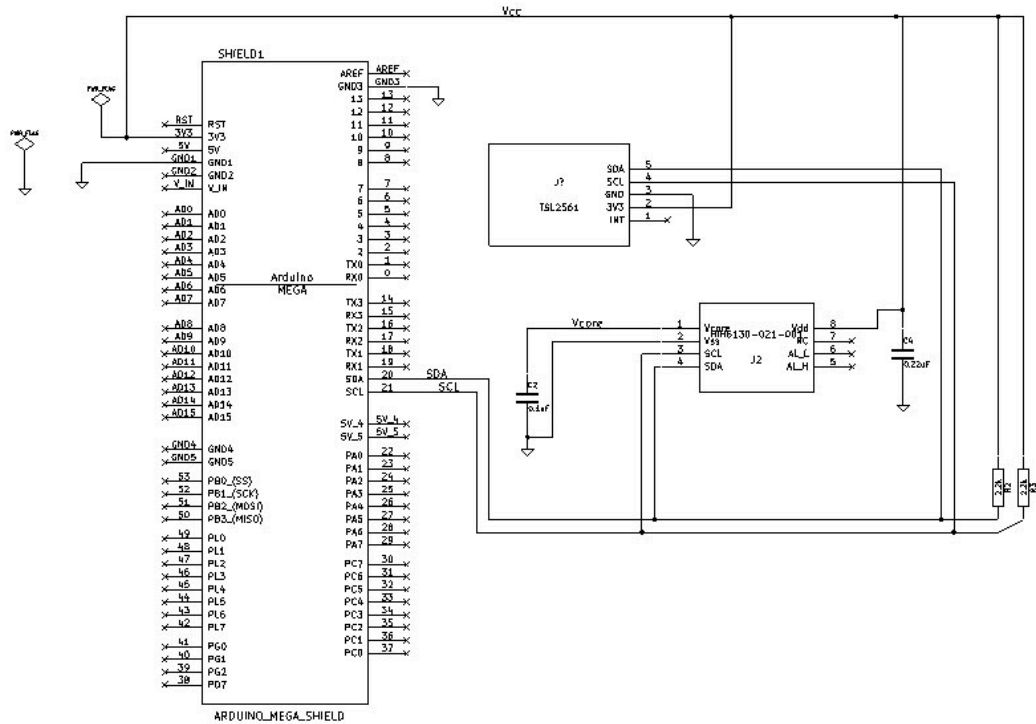


Figura 5.1.6: Esquemático del conexionado del sistema.

A partir del esquemático, la herramienta PcbNew de KiCad nos permite realizar el diseño físico de la PCB. Que posteriormente se fabricará sobre una placa de baquelita recubierta de cobre.

5.1.5. Bluetooth USB dongle

El uso de un Bluetooth USB dongle es una manera sencilla y económica de añadir las funcionalidades de la tecnología Bluetooth a un dispositivo que no ha sido construido para funcionar con este tipo de comunicación. La librería de Arduino que gestiona el USB Host está en fase de desarrollo y no es compatible con todos los dongles. El principal problema estriba en que no existen unos requisitos claros que deba cumplir el dongle para ser compatible, por lo que la elección del mismo se reduce al ensayo y error. Existe una lista de dispositivos compatibles, por desgracia se basa en enlaces directos a los vendedores, que muchas veces varían el producto sin reflejarlo en las especificaciones.



Figura 5.1.7: Bluetooth USB dongle.

El dongle USB que se ha utilizado en este proyecto es del fabricante CSR- Cambridge Silicon Radio. Con versión 2.0 + EDR. La mejora EDR (Enhanced Data Rate) permite reducir el consumo de energía mediante un ciclo de trabajo reducido, y aumenta la tasa de transmisión hasta 2,1 Mbit/s.

Mediante la tecnología Bluetooth usaremos el rango de frecuencias entre 2,4 y 2,485 GHz dentro de la banda libre de licencia ISM (Industrial, Scientific and Medical). Las especificaciones de diseño de Bluetooth dictan un rango mínimo de conexión de 10 metros, si bien existen dispositivos de mayor potencia de emisión capaces de alcanzar los 100 metros. [6]

5.1.6. Alimentación del dispositivo

Al ser un dispositivo fijo, la alimentación del dispositivo no es un punto crítico. Puede alimentarse mediante el puerto USB tipo B del que dispone la placa mediante uno de los múltiples transformadores para dispositivos móviles que existen en el mercado y un cable USB tipo A - tipo B macho macho. Esta solución es muy versátil, e incluso permitiría utilizar el sistema de forma portátil mediante una batería externa USB, tremendamente popularizadas gracias a los dispositivos móviles.

5.1.7. Código Arduino

Para llevar a cabo la implementación de todas las funcionalidades mencionadas, y la interacción con todos los componentes conectados a la placa, es necesario dotar a la misma de un código bien estructurado que realice las siguientes funciones:

- Integración de las librerías de los distintos componentes: Facilitaran la lectura de los datos y el correcto funcionamiento de cada elemento.
- Peticiones de comunicación a los sensores y lectura de los datos.
- Comprobación de los datos en función de los valores de configuración y los márgenes establecidos.
- Formación y envío de las tramas de datos a través del Bluetooth USB dongle.

El proceso de petición, comprobación y envío se realiza de forma cíclica y constante con una frecuencia establecida por el usuario. Esta frecuencia se introduce como intervalo de tiempo entre medidas, directamente en segundos para mayor comodidad del usuario. Este intervalo está fijado por defecto en 30 segundos, pero el usuario puede modificarlo a voluntad, hasta un máximo de 3600 segundos, espaciando las medidas hasta 1 hora.

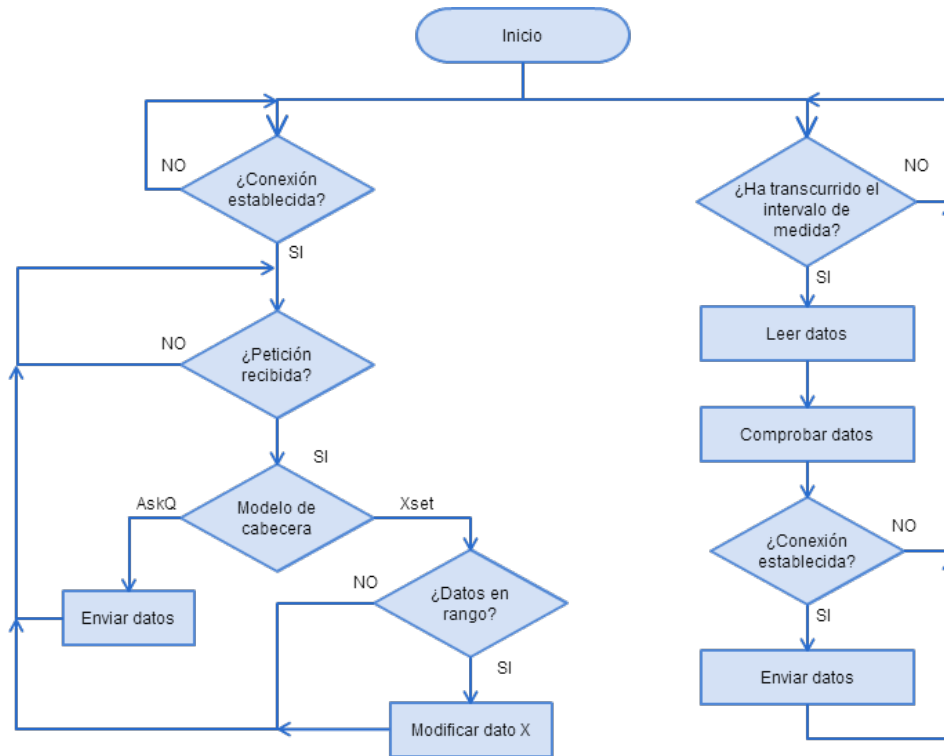


Figura 5.1.8: Diagrama de flujo del código para Arduino.

Para la programación y carga de este código en la placa Arduino, es necesario el entorno de desarrollo Arduino, libre y gratuito. Para el desarrollo de este proyecto se ha utilizado la versión 1.6.4.

5.2. Diseño de la aplicación Android: ArtSense

La aplicación ArtSense está basada en el código de la aplicación Blueterm [7], no es más que una capa de personalización para aprovechar la gestión de Bluetooth, ya implementada. Se busca mostrar de forma sencilla e intuitiva los datos recibidos desde la mota sensora y permitir su almacenaje para posterior consulta y análisis. Es necesario además, permitir al usuario la modificación de ciertos valores de configuración. El esquema de las modificaciones sobre Blueterm puede verse en la Figura 5.2.1.

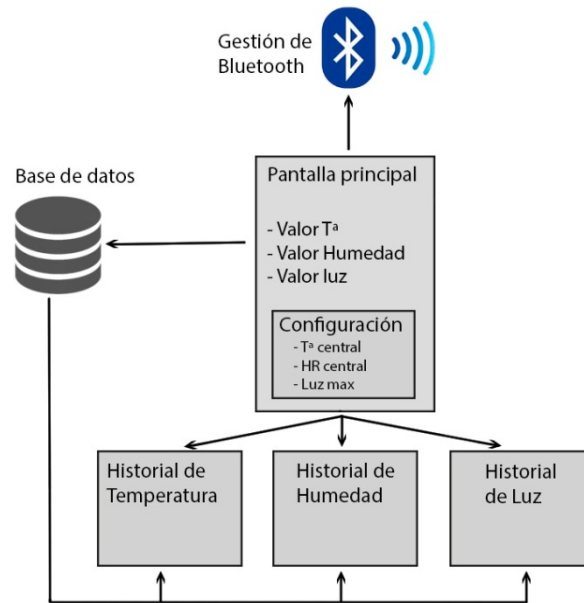


Figura 5.2.1: Esquema general de la aplicación ArtSense.

Las aplicaciones Android requieren de clases y actividades para su funcionamiento. Las actividades gestionan las distintas ventanas de la aplicación por las que el usuario puede navegar, mientras que las clases dotan de funcionalidad al contenido de las mismas. Para llevar a cabo la aplicación ArtSense, se mantienen las clases y actividades de Blueterm enfocadas a la gestión de Bluetooth, tanto transmisión como conexión, y se eliminan o sustituyen el resto.

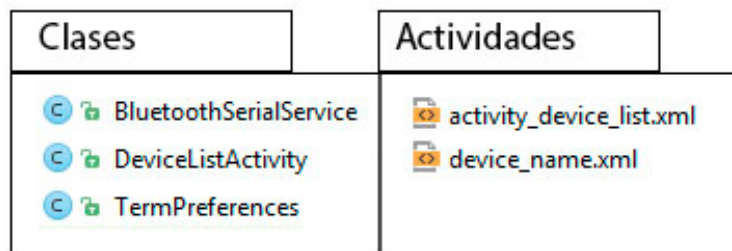


Figura 5.2.2: Clases y actividades de Blueterm que se mantienen.

Para conseguir las funcionalidades propuestas, se realizan las actividades y clases que se detallan a continuación. Pueden verse de forma resumida en la Figura 5.2.3:

Actividades:

- Activity_main: Actividad principal que se encargará de centralizar todas las funcionalidades. Se mostrarán los valores actuales de temperatura, humedad y luz. Al mismo tiempo debe facilitar el acceso a los historiales de datos guardados y a la configuración, así como mostrar el estado de la conexión y permitir su modificación.

- Activity_historico_temperatura, _humedad, _luz: Serán necesarias tres actividades de este tipo, una para cada parámetro guardado. En ellas se mostraran las tablas de datos de cada variable.
- lista_temperatura, _humedad, _luz: Si bien no aparecerán como ventanas independientes, las actividades de lista, de nuevo una por cada parámetro, facilitan la integración de los valores de la base de datos en la Activity_historico correspondiente.

Clases:

- MainActivity: Dota de funcionalidad a la actividad principal, contiene las funciones necesarias para llevar a cabo las peticiones del usuario a través de la interfaz. Llamadas a otras actividades, uso de botones y actualización de valores de medida y estado de la conexión.
- Luz, Temperatura y Humedad: Definen cada uno de los parámetros y las variables que contendrán de cara a su almacenamiento en la base de datos para su posterior procesado.
- HistoricoHumedad, HistoricoLuz e HistoricoTemperatura: Estrechamente relacionadas con las actividades del mismo nombre puesto que les proporcionan el contenido. Cada una de estas clases será la encargada de realizar la lectura de la base de datos y gestionar la actualización de datos.
- CustomAdapterTemperatura, Humedad y Luz: Estas clases se encargan de organizar el contenido de las listas de datos. Se realiza una llamada a la misma desde la clase Histórico correspondiente para ordenar los datos y modificar el color en el que se muestran según su estado.
- ConsultasBD: Contiene los métodos y funciones necesarias para crear la base de datos de cada parámetro con sus respectivos valores, así como para almacenar o acceder a la información cuando sea necesario.

Para una sencilla interpretación de los datos, tanto en la actividad principal como en los historiales, se utiliza un código de colores. Rojo para valores demasiado altos y azul para valores por debajo del mínimo. En la actividad principal se incluye también el color verde para los valores que cumplen con las condiciones de conservación. En los historiales se prescinde del verde, manteniendo el color negro para los datos en rango, presumiblemente la mayor parte, dado que tiene una mejor visibilidad y hace que resulte más cómoda la lectura.

Clases	Actividades
 ConsultasBD  CustomAdapterHumedad  CustomAdapterLuz  CustomAdapterTemperatura  HistoricoHumedad  HistoricoLuz  HistoricoTemperatura  Humedad  Luz  MainActivity  Temperatura	 activity_historico_humedad.xml  activity_historico_luz.xml  activity_historico_temperatura.xml  activity_main.xml  lista_hum.xml  lista_luz.xml  lista_temp.xml

Figura 5.2.3: Clases y actividades incluidas en ArtSense

La aplicación se ha desarrollado completamente en Android mediante la plataforma Android Studio. Ambas plataformas permiten una inmersión especialmente cómoda gracias a la comunidad de desarrollo, su cuidada documentación y el amplio abanico de guías y ejemplos para implementar las funcionalidades básicas. [8]

5.3. Diseño del protocolo de comunicación

Puesto que se han utilizado las funcionalidades de la aplicación Blueterm y la biblioteca HostShield de Arduino, no ha sido necesario el diseño de un protocolo de sincronización, o de un procedimiento de control de errores en la transmisión y/o perdidas. Las funcionalidades utilizadas se ocupan de esos menesteres. Por el contrario si fue necesario tomar decisiones en lo relativo a las tramas de datos que deben enviarse a través de este enlace. Dado que no controlamos las cabeceras utilizadas por Blueterm y Arduino en su comunicación, es necesario incluir una subtrama que indique al receptor el tipo de datos que le lleguen.

Se trata de una trama de alto nivel, donde los datos se envían como cadenas de caracteres ASCII. Este tipo de trama exige un carácter de separación para poder distinguir los campos. Se ha elegido como separación el símbolo de asterisco "*". Existen dos tipos de trama que se envían desde el Arduino a la aplicación, y otros dos que se envían en sentido contrario.

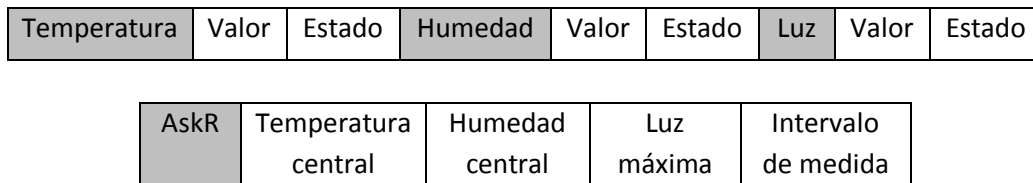


Figura 5.3.1: Tramas originadas en Arduino

Las tramas que se envían desde Arduino, mostradas en la Figura 5.3.1, tienen como finalidad hacer llegar a la aplicación los datos que necesita el usuario. Las dos tramas existentes son para la transmisión de las lecturas de los sensores y para la transmisión de los valores de configuración de la mota sensora.

La trama de lecturas contiene a su vez tres subtramas. Cada una de las cuales corresponde a un parámetro ambiental y se identifica con una cabecera que lo indica. A continuación se envía el valor medido de dicho parámetro y una variable de estado determinada por las comprobaciones realizadas en el Arduino en base a los límites establecidos. La cabecera de temperatura cumple una doble función, pues sirve también para identificar la trama completa.

La trama de datos de configuración es más sencilla, se identifica por la cabecera AskR y a continuación contiene los datos de temperatura central, humedad central, luz máxima e intervalo de medida que la mota tiene fijados en el momento de construir la trama.

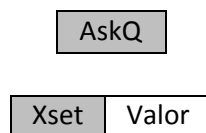


Figura 5.3.2: Tramas originadas en la aplicación ArtSense

Las tramas enviadas desde la aplicación con destino al Arduino son mucho más sencillas como se aprecia en la Figura 5.3.2. Tienen como finalidad consultar los datos de configuración o bien modificarlos.

La trama de solicitud de consulta se limita a enviar la cabecera "AskQ". La recepción en el Arduino de esta cabecera iniciara el proceso de respuesta con una trama de datos de configuración.

La trama de modificación de datos se inicia con una cabecera que indica el tipo de valor que se desea modificar, seguido del nuevo valor. Dichos valores se comprueban en origen para evitar que el usuario introduzca valores fuera de rango, y se vuelven a

comprobar en el destino para evitar fallos en el funcionamiento de la aplicación causados por problemas en el envío. Las cabeceras y márgenes de cada valor se muestran en la Figura 5.3.4.

Cabecera	Valor modificado	Rango aceptado
Tset	Temperatura central	18 - 26 °C
Hset	Humedad central	35-65 % HR
Lset	Luz máxima	Hasta 5000 lux
Iset	Intervalo de medida	30-3600 s

Figura 5.3.4: Opciones de la trama de modificación de datos de configuración.

5.4. Consideraciones finales

A lo largo de este capítulo se han detallado las actuaciones requeridas para cumplir los requisitos planteados en el Capítulo 4. Queda patente por la variedad de tecnologías y componentes utilizados, que se trata de un proyecto multidisciplinar. El verdadero reto del presente proyecto ha sido integrar y hacer funcionar al unísono las distintas partes que lo componen, de forma que resulten provechosos los recursos que ofrece cada una de ellas. Las tecnologías utilizadas se enmarcan entre las más actuales, con un futuro prometedor que hace pensar que tanto este proyecto como los conocimientos adquiridos durante el desarrollo del mismo tendrán una extensa vida útil.

Referencias

- [1] Información técnica sobre I²C. [Consulta: 10-6-2015]. Disponible en: <http://www.i2c-bus.org/>
- [2] Datasheet TSL2561 Light-to-Digital Converter. Texas Advanced Optoelectronic Solutions Inc., Plano, TX, 2005.
- [3] Datasheet HIH6130-021-001. Honeywell International Inc., Golden Valley, MN, 2013.
- [4] Normas de Conservación Preventiva para la Implementación de Sistemas de Control de Condiciones Ambientales en Museos, Bibliotecas, Archivos, Monumentos y Edificios Históricos. Instituto del Patrimonio Cultural de España, Madrid, 2009.
- [5] Información técnica sobre Arduino Mega ADK. [Consulta: 10-6-2015]. Disponible en: <http://www.arduino.cc/en/Main/ArduinoBoardMegaADK>
- [6] Información técnica sobre Bluetooth. [Consulta: 10-6-2015]. Disponible en: <http://www.bluetooth.com/Pages/Basics.aspx>
- [7] Blueterm en Google play. [Consulta: 10-6-2015]. Disponible en: <https://play.google.com/store/apps/details?id=es.pymasde.blueterm&hl=es>
- [8] Comunidad de desarrollo Android. [Consulta: 10-6-2015]. Disponible en: <https://developer.android.com/index.html>

Parte III

Evaluación y Conclusiones

Capítulo 6. Evaluación y resultados

Una vez finalizado el desarrollo del proyecto, es un requisito imprescindible realizar una valoración de los resultados obtenidos. Para ello se parte de los objetivos inicialmente planteados y se analiza el grado de consecución de cada uno de ellos. Simultáneamente se han realizado varias pruebas para comprobar la validez del sistema y su implementación.

6.1. Grado de cumplimiento de los objetivos

- **Adquisición no invasiva de parámetros de medida:** Los sensores utilizados no alteran las condiciones del entorno. Es más, el sistema tiene un tamaño reducido que puede ser empaquetado de forma discreta para que no suponga un agravio visual en el entorno de exposición de la obra.
- **Monitorización en tiempo real:** Se ha establecido un intervalo de medida que puede ser modificado por el usuario entre 30 segundos y una hora, dependiendo de las necesidades del mismo. Los parámetros de temperatura y humedad se pueden considerar monitorizados en tiempo real con un intervalo de 30 segundos entre medidas, dado que su variación es más lenta. En el caso de la luz, que obviamente admite variaciones más rápidas, se admiten modificaciones posteriores del sistema para realizar su monitorización en mayor detalle.
- **Control de los parámetros dentro de los márgenes establecidos:** La mota sensora comprueba tras cada medida el estado de los datos con respecto a los márgenes a priori establecidos. Esta comprobación se realiza previa al envío, lo que permitiría utilizar la propia mota para el control activo de los parámetros, independientemente del dispositivo móvil que completa el sistema, y de su estado de conexión.
- **Generación de alarmas:** El sistema avisa mediante un código de colores al usuario de la aplicación Android. Sin embargo, dado que la comprobación se realiza en origen, la placa Arduino dispone de la información correspondiente y podría gestionarla como se menciona en el punto anterior.
- **Transmisión inalámbrica independiente:** La transmisión se realiza vía Bluetooth, sin depender de cableado ni redes de cobertura o dispositivos de terceros. Para garantizar un funcionamiento correcto se requiere tan solo la mota sensora y un dispositivo móvil. La conexión con la mota sensora está protegida por una contraseña a fin de evitar posibles injerencias de personal no autorizado.

- Visualización del estado actual: La aplicación ArtSense muestra en su pantalla principal y en un lugar privilegiado, los datos de la última medida recibida.
- Facilidad de uso: La mota sensora solo requiere de alimentación para comenzar a funcionar por sí sola. La aplicación ArtSense tiene un diseño sencillo que permite al usuario interactuar sin dificultad.
- Almacenamiento de datos: La aplicación instalada en el dispositivo móvil almacena en una base de datos la información de cada parámetro, incluyendo la fecha y hora de su recepción, así como el estado de la medida en función de los límites establecidos.
- Consulta de datos: Los historiales guardados están accesibles desde la aplicación en cualquier momento. El acceso a la base de datos se realiza con facilidad y rapidez desde la pantalla principal.
- Interpretabilidad de los datos: La inclusión de la fecha y la hora de recepción de la medida, así como el código de colores que identifica el estado de la misma, permite interpretar los datos de forma rápida y concisa.
- Robustez: Utilizar sistemas de código libre con una amplia comunidad de desarrollo, garantiza que se han testeado más allá de lo que puede realizar una sola persona en un periodo de tiempo limitado. Además, se ha realizado un desarrollo cuidado y una serie de pruebas, en condiciones forzadas, buscando la identificación y corrección de cualquier aspereza que pudiera ocasionar fallos en el sistema.

Consideramos que se han alcanzado los objetivos propuestos con un amplio grado de cumplimiento de los mismos. Además se ha dejado abierta la puerta a realizar trabajos futuros que añadan características o mejoras al sistema y su funcionamiento, de forma que la inclusión de estos no suponga graves cambios en la estructura actual.

6.2. Pruebas de alcance

Se han realizado pruebas de alcance de conexión con dos dispositivos móviles distintos. Un Samsung Galaxy S3 mini con Android 4.1.2 y versión de Bluetooth 4.0 y un LG G3 con Android 5.0 y Bluetooth 4.0 + LE. Para realizar las pruebas se establecieron las siguientes pautas:

- Escenario al aire libre, con el menor número de obstáculos posible. A última hora de la tarde.

- Las pruebas se realizaron alimentando la mota sensora con una batería externa con salida USB de 5V y 1A.
- Se fijó el intervalo de medidas en 1 segundo, de forma que se recibieran medidas constantes.
- Se mantuvo la mota sensora en un punto fijo, y se hizo variar la distancia del dispositivo móvil.
- El movimiento fue rectilíneo, manteniendo la línea visual entre los dispositivos en todo momento, y aumentando la distancia progresivamente con una velocidad constante.
- Se analizaron las medidas recibidas y perdidas en intervalos de diez segundos.

Con el dispositivo Samsung se tuvieron problemas de conexión para distancias mayores a 10 metros, pero cuando previamente estaba conectado se mantenía la conexión con una calidad óptima de recepción hasta un máximo de 34 metros, donde se producía la desconexión.

El caso del LG es más interesante. Permitía el establecimiento de conexión con la mota sensora en un radio máximo de 30 metros y una media de 27 metro tras cinco intentos. Permitted mantener la conexión hasta una distancia máxima de 80 metros con una media de 75 metros. Esta mayor extensión permitió ver en detalle el comportamiento de la conexión.

Se ha resumido el funcionamiento de la conexión Bluetooth del sistema en tres zonas distintas:

- Zona óptima: Se reciben todas las medidas con una tasa de fallos mínima. El sistema funciona correctamente.
- Zona parcial: La transmisión presenta un comportamiento errático, recibiendo unas veces gran parte de las medidas y sufriendo severas pérdidas en otras.
- Zona de desconexión: Dejan de recibirse datos y finalmente se pierde la conexión entre los dispositivos.

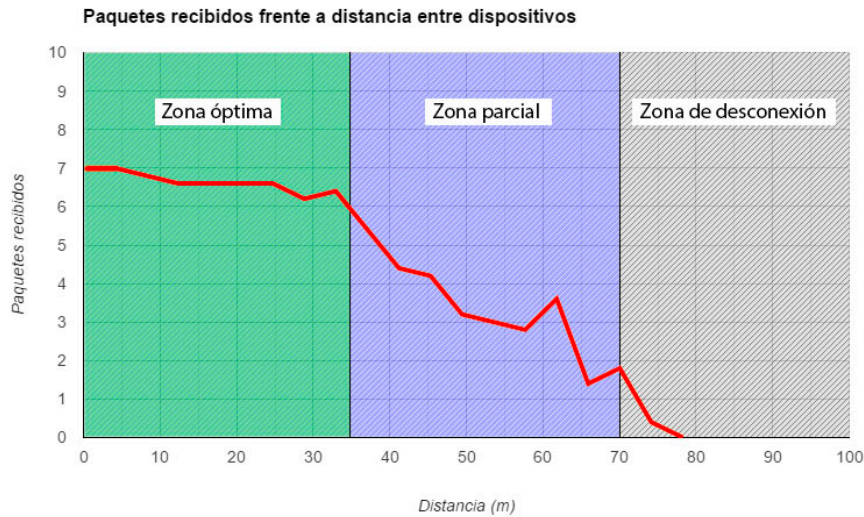


Figura 6.2.1: Resultados de las pruebas de alcance Bluetooth con el smartphone LG G3

En la Figura 6.2.1. Puede verse la media de resultados de las pruebas con el LG, así como la división por zonas de la misma. Se realizaron cinco iteraciones de la prueba.

Queda patente que la capacidad de conexión del sistema depende del dispositivo al que se conecte. Existen tres clases de dispositivos Bluetooth según su potencia de emisión y alcance. La clase 1, con 100 mW de potencia máxima permitida y un alcance aproximado de 100 metros; la clase 2, que se reduce a 2,5 mW y unos 10 metros y por último la clase 3 con 1 mW y apenas un metro de alcance. Desconocemos la potencia de emisión del Bluetooth USB dongle ya que el fabricante no la especifica, pero a la vista de los resultados suponemos que se trata de un dispositivo de clase 2, en el que cuya distancia efectiva se vería incrementada por la interacción con un dispositivo de mayor potencia y sensibilidad de recepción como el smartphone de LG.

Las pruebas arrojaron un resultado secundario que si bien no era el que se buscaba también resulta interesante. Como se ve en la Figura 6.2.1, la recepción máxima de medidas en intervalos de diez segundos se redujo a siete medidas. El intervalo de medida estaba fijado en un segundo, con lo cual deberían haberse recibido diez. Esto llevo a realizar un estudio más detallado sobre las capacidades del sistema para encontrar el problema, que se detalla en el apartado siguiente.

6.3. Estudio de la capacidad de transmisión

Ante la recepción máxima de las pruebas realizadas en el apartado 6.2, hubo que plantearse el análisis de las posibles causas de este resultado.

Tanto el sensor de luz como el de temperatura tienen tiempos de acceso inferiores a un segundo. En concreto el sensor de luz es el más lento, ya que necesita 400 ms para la conversión de los datos a digital. Por tanto, los sensores no eran los culpables del fallo,

pero sí que marcan un intervalo mínimo de 400 ms necesario para obtener una segunda medida, en intervalos inferiores enviaría una medida repetida.

Al reducir el intervalo de medida a 400 ms, y comprobar por puerto serie que el Arduino era perfectamente capaz de llevar a cabo la tarea, esperábamos recibir en la aplicación al menos dos medidas por segundo. La recepción conseguida fue de una medida por segundo. Ante tal resultado, se redujo el intervalo a 200 ms. La medida obtenida del sensor de luz fue repetida pero se permitió su lectura. Sin embargo, en la aplicación la tasa de recepción seguía siendo de una medida por segundo.

La mayor longitud de trama de datos enviada, consta de 51 caracteres ASCII de 8 bits cada uno, por lo que su peso conjunto es de 408 bits más el añadido de la cabecera necesaria para su envío por Bluetooth. La capacidad del enlace de 2,1 Mbits por segundo permite enviar hasta 5397 tramas de este tipo por segundo si despreciamos la cabecera Bluetooth, por lo que la conexión de Bluetooth no debería ser la barrera para la recepción de datos con intervalos de tiempo menores.

Finalmente, se consideró la estructura de la aplicación. El registro de los datos se configuró de forma que son necesarios tres accesos a la base de datos antes de estar preparada para aceptar otra medida. El sistema podría optimizarse guardando los datos en un solo acceso, pero se tomó la decisión de hacerlo por separado en el diseño para mantener individualizados los distintos parámetros y aumentar la robustez del sistema ante posibles fallos de uno de los sensores. Esta se consideró la causa más probable del límite observado en la velocidad de recepción de valores.

Se trata de un resultado que no es en absoluto crítico para el funcionamiento del proyecto, a priori se había planteado un intervalo mínimo de medida de 30 segundos, dado que la humedad y la temperatura ambientales no sufren variaciones tan rápidas. Podría considerarse influyente para trabajos futuros en los que se incremente la sensibilidad del sistema a variaciones de luz, para poder detectar el uso de flashes o iluminaciones temporales sobre la obra de arte monitorizada. Sin embargo, dado el tiempo de 400 ms requerido por el sensor de luz, creemos que una capacidad máxima de una medida por segundo no es un defecto considerable.

6.4. Estudio de compatibilidad

La compatibilidad del sistema con los distintos dispositivos móviles queda determinada por varios factores, que se exponen a continuación.

- Sistema operativo Android: La aplicación ArtSense está diseñada y implementada en Android. Por tanto, es un requisito imprescindible que el dispositivo móvil disponga de este sistema operativo. La aplicación es compatible con Android 4.0 y superiores, lo cual la hace compatible con el 90,4% de los dispositivos activos en Google play.

- **Compatibilidad Bluetooth:** A priori, todas las versiones de Bluetooth son retro compatibles, por lo que no debería haber problema en este sentido. Sin embargo, existen unas especificaciones de alto nivel para Bluetooth llamadas perfiles que definen los comportamientos generales que los dispositivos pueden utilizar para comunicarse. Que ambos dispositivos sean capaces de utilizar el mismo perfil es un punto clave para la compatibilidad de ambos. La mota sensora emite con el protocolo RFCOMM que emula los parámetros de un puerto serie, por lo que los dispositivos móviles deben ser capaces de soportar este protocolo. Los dispositivos de las grandes compañías no suelen dar problema en este aspecto, pero es habitual encontrar problemas de conexión en los dispositivos móviles de bajo precio que provienen de fabricantes chinos menos reconocidos.

Las pruebas empíricas que se han llevado a cabo hasta el momento nos dejan como dispositivos compatibles el Samsung Galaxy S3 mini, LG G3 y el Nexus 4.

Por otro lado, se han probado varios móviles de procedencia china como el POMP W88 y el MPIE 809T, así como el Aquaris 5 de la compañía española BQ. Con ninguno de ellos se consiguió establecer conexión con la mota sensora a pesar de tener la aplicación instalada. Presumiblemente debido que estos móviles no cumplen las condiciones de Bluetooth anteriormente mencionadas.

6.5 Encuesta sobre la aplicación ArtSense

Cuando se desarrolla una aplicación, incluso teniendo como objetivo principal la sencillez y facilidad de uso, es fácil perder de vista aspectos que no resultan evidentes a otros usuarios. Por ello se ha realizado una breve encuesta a usuarios ajenos al desarrollo de la misma para valorar los aspectos que la componen. Se ha utilizado el sistema de puntuación sobre 5 utilizado en la Google play para la valoración general de aplicaciones. La encuesta y sus resultados pueden consultarse en la figura 6.5.1.

Si bien la encuesta ha tenido una participación reducida y sus resultados no son concluyentes, se aprecia un buen comienzo en el contacto con los usuarios. La verdadera utilidad de la valoración de la aplicación surgirá al mantener contacto con un mayor número de usuarios e ir incluyendo los cambios o modificaciones más solicitadas.

La principal crítica aportada por los usuarios encuestados ha sido relativa a los botones de acceso a los historiales, que para muchos pasaban desapercibidos, ya que no los identificaban como botones con los que pudieran interactuar, sino como un adorno de la pantalla principal.

Diseño	Puntuación
Me gusta el aspecto visual de la aplicación	★★★★★
Es fácil encontrar la información	★★★★★
Funcionalidad	
Los datos son fáciles de interpretar	★★★★★
El código de colores facilita la interpretación	★★★★★
Es fácil establecer la conexión entre los dispositivos	★★★★★
Es fácil acceder a los datos almacenados	★★★★★
Uso	
Me ha sido fácil usar la aplicación en general	★★★★★
La aplicación tiene una buena velocidad de respuesta	★★★★★

Figura 6.5.1: Encuesta de valoración para ArtSense

Frente al código de colores aparecieron opiniones encontradas, puesto que aunque la mayoría de usuarios encuestados lo encontraban muy útil, otros lo encontraban inútil o incluso confuso.

Capítulo 7. Conclusiones y trabajo futuro

Con este capítulo finaliza la presente memoria. Se resumirán las conclusiones obtenidas a lo largo del desarrollo del proyecto y las actividades relacionadas con el mismo. Se realizara asimismo un análisis de posibles líneas de trabajo futuro que aporten mejoras o cubran aspectos débiles del desarrollo.

7.1. Conclusiones

Utilizando componentes disponibles en el mercado y de bajo coste, ha sido posible diseñar e implementar un sistema de monitorización de obras de arte. El sistema permite monitorizar los valores de temperatura, humedad relativa y luminosidad, y controlar que se mantengan en las condiciones óptimas de conservación de obras de arte dictadas por la normativa del Ministerio de Educación y Cultura gracias a la primera parte del mismo: la mota sensora. Posteriormente, mediante una conexión Bluetooth se envían los datos a un dispositivo móvil, en el que se puede acceder de forma sencilla e intuitiva tanto al valor en tiempo real como a los registros históricos. Esto es factible gracias a la aplicación ArtSense, diseñada en entorno Android, que además permite modificar tanto los rangos de valores deseables en los parámetros monitorizados, como la periodicidad con la que se registran los datos. El sistema diseñado cumple todos los requisitos establecidos al definir los objetivos: no altera las condiciones del entorno, ofrece información en tiempo real, comprueba si los datos están o no dentro del rango establecido, genera alarmas y ofrece la posibilidad de asociar sistemas de control automático. La transmisión de datos es autónoma e independiente, permitiendo la consulta en tiempo real y el almacenamiento de los registros transmitidos, todo ello de forma fácil e intuitiva y sin alterar el medio en el que se expone la obra bajo control.

La viabilidad del sistema se ha logrado trabajando al unísono en las plataformas Android y Arduino para aunar las partes de hardware y software. Ambas libres, lo que permite que el sistema sea extensible por terceras personas.

El resultado es un sistema de monitorización sencillo y económico, que permite un cuidado detallado de las obras de arte y su entorno de conservación. Sin suponer un agravio al entorno de exposición de la obra, y que permite la comodidad de acceder en todo momento a la información desde un dispositivo móvil.

7.2 Trabajo futuro

Desde un primer momento del desarrollo, se planteó un sistema escalable que permitiera añadir funcionalidades o mejoras de forma sencilla, y de esta manera dar respuesta a las necesidades posteriormente detectadas. A lo largo de la memoria se han

mencionado brevemente algunas, que se resumen a continuación junto con otras posibles vías de trabajo.

- **Nuevos sensores:** Aunque se han tomado los valores más influyentes en la conservación preventiva, se pueden añadir sensores a la plataforma. La inclusión de estos sensores incrementaría el nivel de detalle en el cuidado de la obra, alargando su vida en perfecto estado. El uso del protocolo de comunicación I²C entre los sensores y la placa Arduino, permite su incorporación sin problemas de compatibilidad. También existe capacidad de transmisión suficiente entre la mota sensora y el dispositivo móvil para dar cabida a la nueva información. Se recomienda comenzar este trabajo con sensores que permitan medir y controlar la calidad del aire, tanto en materia particulada (polvo, hollín, etc) como en contenido de gases contaminantes (SO₂, NO₂ y O₃).
- **Gestión activa de parámetros:** El sistema está preparado para acometer la gestión activa de los parámetros que monitoriza. Este trabajo probablemente deba ser llevado a cabo de forma especializada para cada instalación, pero tanto la comunicación I²C que permite interactuar con otros microcontroladores que puedan estar presentes en los sistemas de climatización, como el control de las medidas antes de su envío al dispositivo móvil, hacen posible el añadido de esta funcionalidad de forma sencilla.
- **Mejora del protocolo de comunicación:** Se puede optimizar y mejorar el sistema de comunicación. Por un lado es conveniente reducir al mínimo el peso de las tramas de datos, excluyendo la información que no sea estrictamente necesaria. Por otro lado, se plantea la inclusión de un mecanismo de control, recuperación y/o corrección de errores.
- **Incremento de la distancia de conexión:** Un factor importante de mejora sería permitir el acceso a los datos independientemente de la distancia. Para ello sería necesario recurrir a una conexión del dispositivo a internet. Esta conexión, que probablemente requiera añadir un servidor al sistema, permitiría además la inclusión de notificaciones en la aplicación móvil.
- **Aplicación para múltiples plataformas:** Es importante permitir el acceso al sistema al mayor número de usuarios posible. Por ello se plantea el desarrollo de la aplicación para otros sistemas operativos móviles como pueden ser iOS o Windows Phone.
- **Traducción a otros idiomas:** La aplicación se ha presentado en español e inglés para permitir su uso de forma internacional. Se pueden añadir otros idiomas si se requiere.

- Muestreo gráfico de los datos: Permitir la visualización grafica de los datos en la aplicación ArtSense, supondría una mejora en la facilidad de interpretación de los datos.
- Exportación de datos: La aplicación podría permitir exportar la base de datos. La variable más sencilla es utilizar un fichero de texto o el formato csv, haciéndolo compatible con múltiples programas de análisis de datos.
- Organización de los datos por fechas y franjas horarias: Permitir el acceso a secciones concretas de la base de datos, para facilitar su consulta y gestión. Esta información ya se recoge en la aplicación actual, solo falta implementar la funcionalidad.
- Apartado de configuración más detallado: Desligar el apartado de configuración de la pantalla principal, de forma que se puedan mostrar de forma más detallada los valores de configuración y los rangos válidos de medida.

Apéndice I. Manual de usuario

Bienvenido al sistema Artsense.

ArtSense es un sistema de ayuda para el cuidado de obras de arte y su perfecta conservación. El sistema mide temperatura, humedad y luz, y comprueba que se mantengan en un rango adecuado. Se compone de dos partes: el aparato sensor, que debe ser colocado junto a la obra y la app ArtSense que puede instalar en su dispositivo móvil Android.

Sitúe la mota sensora en un lugar adecuado en el entorno de la obra de arte, y conecte su alimentación. Instale en su dispositivo móvil la aplicación ArtSense.



Figura A1.1: Pantalla de inicio. Desconectado.

Para comenzar, conecte su dispositivo móvil con la mota sensora. Esta conexión se realiza mediante Bluetooth, por lo que será necesario que active Bluetooth en los ajustes de su dispositivo. Pulse el botón conectar en la pantalla principal.

Se ofrecerá una lista de los dispositivos vinculados y la opción de buscar nuevos dispositivos. Si es la primera vez que utiliza el sistema, pulse buscar nuevos dispositivos. Entre los dispositivos disponibles, seleccione "**MotaSensora**".

Es necesario introducir un pin de acceso. Por defecto, el pin está fijado a "0000".

Una vez establecida la conexión, se mostrará la pantalla como en la Figura A1.2.

1. Lectura de medidas:

El sistema toma las medidas automáticamente. En la parte superior se muestran los valores medidos más recientes de Temperatura, Humedad y Luz. El color del icono correspondiente cambia según el estado de la medida:

- **Rojo:** Valor demasiado alto.
- **Verde:** Valor en el rango adecuado
- **Azul:** Valor demasiado bajo.



Figura A1.2: Pantalla principal. Conectado.

En el ejemplo reflejado en la figura A1.2 se incluye un valor de temperatura excesivamente alto (termómetro en rojo), con un valor de humedad relativa por debajo de lo establecido (la nube en azul). La luz ambiente está dentro del rango prefijado. En condiciones ideales los tres iconos deben aparecer en color verde.

2. Historial de datos

El sistema guarda automáticamente las medidas recibidas. Para acceder a los datos guardados, pulse el icono en negro de la medida correspondiente en el centro de la pantalla principal.

En el historial se muestran primero las últimas medidas recibidas. También se utiliza el código de colores para facilitar su interpretación. En la Figura A1.3 se muestra un ejemplo de historial de humedad.



FECHA	HORA	VALOR (% HR)
2015-06-11	18:14:38	59.66
2015-06-11	18:14:36	65.90
2015-06-11	18:14:35	64.39
2015-06-11	18:14:34	59.90
2015-06-11	18:14:32	54.04
2015-06-11	18:14:31	47.09
2015-06-11	18:14:29	38.66
2015-06-11	18:14:28	35.92
2015-06-11	18:14:26	36.32
2015-06-11	18:14:25	36.77
2015-06-11	18:14:23	37.60
2015-06-11	18:14:22	37.84
2015-06-11	18:14:20	38.46
2015-06-11	18:14:19	37.70
2015-06-11	18:14:18	35.78
2015-06-11	18:14:16	33.92
2015-06-11	18:14:15	33.04
2015-06-11	18:14:13	32.85
2015-06-11	18:14:12	32.55
2015-06-11	18:14:10	32.57
2015-06-11	18:14:09	32.57
2015-06-11	18:14:07	32.62

Figura A1.3: Historial de datos guardados. Humedad

3. Configuración

En la parte baja de la pantalla principal, se muestran los valores de configuración que determinan el intervalo válido de medidas. El usuario puede modificar estos valores y pulsar el botón correspondiente "Cambiar" para modificar el comportamiento del sistema.

- T° central: Punto medio del intervalo de temperatura. El sistema permite temperaturas dos grados arriba y abajo de la temperatura central. Puede fijarse a cualquier valor entre 18°C y 26 °C.
- HR central: Punto medio del intervalo de humedad. El sistema permite humedades un 5% arriba y abajo de la humedad central. Puede fijarse a cualquier valor entre 35% y 65%.
- Luz Max: Iluminación máxima permitida por el sistema. Puede fijarse a cualquier valor entre 0 y 5000 lux.
- Intervalo de medida: Determina cada cuanto tiempo se toman las medidas. Puede fijarse a cualquier valor entre 30 segundos y 3600 segundos. (Nota: 3600 segundos corresponden a una hora entre medidas).

Apéndice 2. Presupuesto

En la Figura A2.1. se muestra el presupuesto elaborado para la construcción del prototipo de la mota sensora basada en Arduino.

El presupuesto no tiene en cuenta el coste de la mano de obra necesaria, al tratarse de un proyecto de formación. Tampoco incorpora el coste de las herramientas necesarias para llevar a cabo la implementación del sistema. Solo se tiene en cuenta el coste de los componentes que integran el prototipo.

Puesto que el coste de los componentes puede variar según el proveedor, se indica la casa comercial a la que corresponde el precio del componente utilizado en la elaboración del presupuesto.

Componente	Cantidad	Distribuidor	Referencia	Coste (€)
Placa Arduino Mega ADK	1	RS	A000069	59,95
Sensor de luz para placa Arduino TSL2561	1	Sparkfun	SEN-12055	5,95
Sensor de temperatura y humedad	1	RS	HIH6130-021-001	14,19
Bluetooth USB dongle*	1	-	-	5,00
Resistencia 2,2 K Ω	2	Sonytel	P220K50V	0,42
Condensador 0.1 μ F	1	Sonytel	P1M50V	0,54
Condensador 0.22 μ F	1	Sonytel	P22M50V	0,54
Placa fotorresistente para PCB 160x100x1,6mm	1	RS	FB2-100x160S	7,65
TOTAL				94,24

Figura A2.1: Presupuesto para la construcción del prototipo de la mota sensora.

**Nota: La elección del dongle es un tema delicado como se explicó en el Capítulo 5. Al no poder ofrecer un proveedor estable, se ha tomado un precio orientativo y se recomienda visitar la referencia [1] de este apéndice.*

Referencias

[1] Lista de dongles compatibles con la librería Bluetooth. Wiki. [Consulta 12-06-2015]
Disponible en: https://github.com/felis/USB_Host_Shield_2.0/wiki/Bluetooth-dongles