

**Universidad Autónoma de San Luis Potosí**  
**Facultad de Ingeniería**  
**Área de Computación e Informática**

**Proyecto Integrado**

“Monitoreo y mantenimiento de plantas decorativas  
en edificios verdes”

**Profesor:**

M. en I. Omar Vital Ochoa

**Asesor:**

Dr. Carlos Soubervielle Montalvo

**Alumnos:**

Alfredo Orozco de la Paz

Daniel Varela Varela

Daniel Sánchez Ovalle



# INDICE

<b>1</b>	<b>Introducción .....</b>	<b>1</b>
1.1	Planteamiento del problema .....	1
1.2	Justificación .....	2
1.3	Requerimientos de Hardware-Software .....	3
<b>2</b>	<b>Marco teórico .....</b>	<b>4</b>
2.1	Matriz de materias .....	4
2.2	Bluetooth 4.0 .....	5
<b>3</b>	<b>Metodología .....</b>	<b>7</b>
3.1	Hardware .....	7
3.1.1	Requerimientos .....	7
3.1.2	Diseño .....	10
3.1.3	Implementación .....	11
3.1.4	Pruebas .....	12
3.2	Firmware .....	13
3.2.1	Requerimientos .....	14
3.2.2	Controladores implementados .....	13
3.2.3	Diagrama de flujo .....	15
3.2.4	Pruebas .....	16
3.3	Aplicación .....	17
3.3.1	Objetivo .....	17
3.3.1	Descripción .....	17
3.3.1	Requerimientos .....	16
3.3.2	Diagramas .....	18
3.3.3	APP .....	20
<b>4</b>	<b>Pruebas .....</b>	<b>23</b>
<b>5</b>	<b>Costos .....</b>	<b>25</b>
<b>6</b>	<b>Conclusiones y trabajo a futuro .....</b>	<b>26</b>
<b>7</b>	<b>Anexos .....</b>	<b>27</b>
7.1	Diagrama esquemático del hardware .....	27
<b>8</b>	<b>Referencias .....</b>	<b>28</b>
<b>9</b>	<b>Glosario de Términos .....</b>	<b>29</b>

# 1 Introducción

## 1.1 Planteamiento del problema

---

Actualmente existe una necesidad creciente por adoptar formas de vivir más apegadas a la naturaleza, de cuidar nuestro medio ambiente, así como del cuidado de nuestro planeta. Todo esto derivado de la inminente huella dejada por los humanos debido al desarrollo de la industria, el consumismo, la globalización, la urbanización y la deforestación que propicia a el calentamiento global, pero la que más afecta es la falta de conciencia hacia nuestro planeta. Con esta nueva conciencia que se origina a partir del cuidado de nuestro planeta es que se crean nuevas necesidades, tanto humanas como ambientales, como son el cuidado del lugar donde habitamos, la conservación de la flora y la fauna, minimizar la degradación ambiental en nuestra actividad diaria e integración de la naturaleza en el día a día de nuestra vida. Estas necesidades surgen a partir de la falta de tiempo y atención para cuidar las plantas en edificios u hogares debido al ritmo de vida actual y al estrés que generan los empleos en la actualidad.

## 1.2 Justificación

---

En esta creciente integración de nuestra vida diaria con la naturaleza es donde se han propuesto soluciones para reducir el impacto ambiental y al mismo tiempo nos permite convivir más de cerca con ella. Las nuevas tendencias de hogares y edificios verdes, autosustentables e inteligentes están necesitando y proponiendo soluciones innovadoras de todo tipo para el cuidado de la vegetación que la integra y para la reducción de consumo de energía y recursos de los que depende.

Con base en estas necesidades es donde nos hemos propuesto hacer innovación, con una solución para el cuidado y conservación de plantas ornamentales en edificios verdes y en el hogar, que permitirá ahorrar recursos, energía y tiempo en su cuidado y mantenimiento para lograr un mayor beneficio de los recursos disponibles.

### 1.3 Requerimientos de Hardware-Software

---

Este proyecto es realizable mediante un sensor que se incruste en la tierra cerca de la planta, mida sus variables y las envíe mediante Bluetooth a un celular para que la información pueda ser procesada. Este concepto está basado en el Internet de las Cosas, que dice que cualquier cosa puede ser conectado a una red informática, inclusive a internet.

Lo que requerimos para realizar el proyecto son 2 cosas, una es el diseño e implementación un sistema embebido prototipo que se encargue del sensado de las plantas y pueda comunicarse por Bluetooth, y la otra es el desarrollo de una aplicación en Android que se encargue de recibir e interpretar los datos enviados por el sistema embebido para probar su funcionamiento. El sistema embebido se divide en 2 partes: la parte electrónica que integra los sensores con algún microcontrolador y la parte del firmware que ejecutará las tareas de sensado, comunicación Bluetooth y sincronización en el microcontrolador. La Aplicación en Android se encargará de buscar y listar estos sistemas por medio de Bluetooth para posteriormente mostrar su información relacionada con la planta.

En la figura 1.1 se muestra gráficamente como es el funcionamiento del sistema, donde las plantas que tengan el sistema embebido enviarán los datos al celular y este mediante la aplicación reportará la información al usuario.

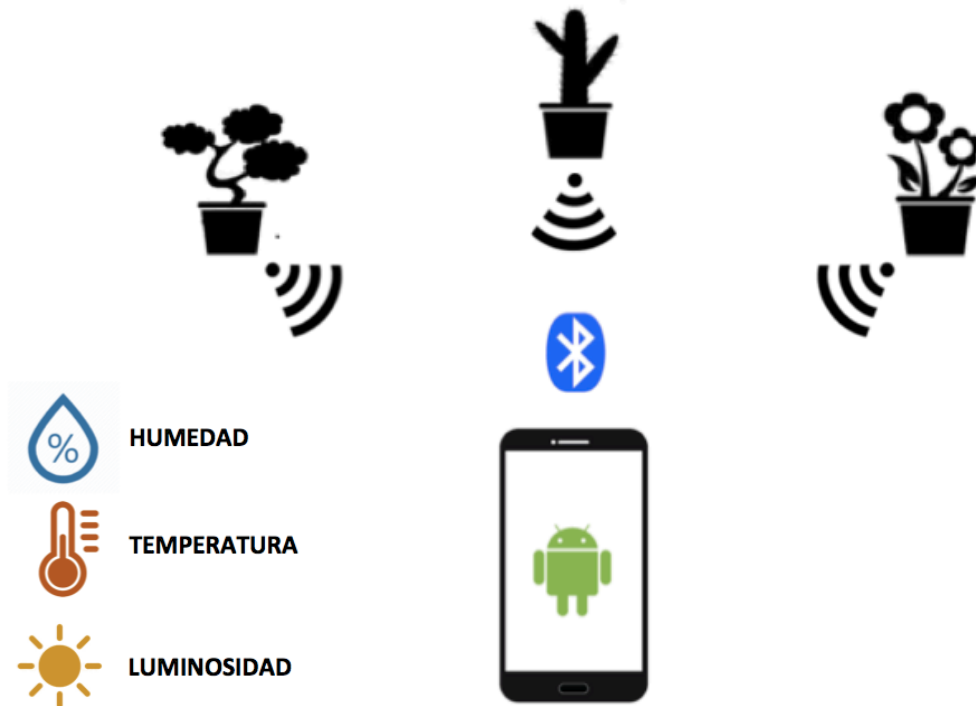


Figura 1.1 Funcionamiento del Sistema

## 2 Marco Teórico

En este capítulo se presenta el marco teórico necesario para llevar a cabo este proyecto, de los conocimientos previos que se necesitan saber para el desarrollo.

### 2.1 Matriz de materias.

Para el desarrollo del proyecto se requiere tener conocimiento en diferentes áreas, la mayor parte de este, se obtiene durante la carrera. En las tablas 2.2.1 y 2.2.2 se muestra la matriz de las materias que se aplican al proyecto.

Tabla 2.2.1. Materias del nivel I al nivel V del plan de estudios.

Área	Nivel				
	I	II	III	IV	V
<b>Formación Básica</b>	Algebra A	Algebra B Calculo A			
<b>Formación Básica Profesional</b>		Electrónica A			
		Estructuras de datos y algoritmos A		Programación Orientada a Objetos	Electrónica B
		Estructuras de datos y algoritmos B			
<b>Formación Profesional</b>					Sistemas Operativos
<b>Formación Socio Humanística</b>					
<b>Formación Investigativa</b>			Estadística	Análisis Numérico	

Tabla 2.2.2. Materias del nivel VI al nivel X del plan de estudios.

	Nivel				
Área	VI	VII	VIII	IX	X
<b>Formación Básica</b>					
<b>Formación Básica Profesional</b>					
<b>Formación Profesional</b>			Bases de Datos	Redes A	Telemática
				Computación Ubicua	
				Tecnologías Web	
<b>Formación Socio Humanística</b>					
<b>Formación Investigativa</b>		Modelos Matemáticos			

La justificación de cada una de estas materias se muestra en el siguiente listado:

- **Electrónica A.** Para el acondicionamiento de los sensores analógicos.
- **Física A.** Para saber el funcionamiento de los sensores que miden variables ambientales.
- **Electrónica B.** para el desarrollo del sistema embebido.
- **Bases de datos.** Para la gestión y almacenamiento de la información de plantas.
- **Redes.** Para la implementación de una conexión P2P.
- **Telemática.** Para la comunicación inalámbrica.
- **Programación orientada a objetos**
- **Estructuras de datos y algoritmos.** Para la programación del sistema embebido.
- **Computación ubicua.** Para la programación de dispositivos móviles.
- **Tecnologías Web.** para el desarrollo de la interfaz web.
- **Sistemas Operativos.** para el desarrollo del sistema operativo del sistema embebido en tiempo real (RTOS).
- **Modelos Matemáticos.** para la gestión de tiempos y procesos.
- **Álgebra y Cálculo.** para el desarrollo de las ecuaciones que describen el comportamiento de los sensores.
- **Estadística.** Para el análisis de datos obtenidos por los sensores (graficar, promediar, pronosticar, etc.)
- **Análisis Numérico.** Para la programación de las ecuaciones matemáticas necesarias.

## 2.1 Bluetooth 4.0

---

El protocolo bluetooth 4.0, también llamado bluetooth Smart o BLE fue desarrollado por Nokia en 2010 y posteriormente se fue adoptando en toda clase de dispositivos.

Los dispositivos que cuentan con bluetooth 4.0 pueden comunicarse con dispositivos que lo implementan así como también con dispositivos que cuentan con bluetooth clásico (anterior a 4.0) pero no a la inversa, es decir que los dispositivos con bluetooth clásico no pueden comunicar con los que tienen 4.0.

Un dispositivo con Bluetooth Low Energy (BLE), puede comunicarse con otros dispositivos de 2 maneras: con broadcast o con conexión. El método utilizado en el proyecto es broadcast.

Como se muestra en la figura 2.1.1, la comunicación por Broadcast tiene 2 roles principales, que son el Broadcaster y el Observer. El primero es el dispositivo que transmite la información a todos los dispositivos que se encuentren dentro del alcance y el Observer recibe la información, pero no se establece una conexión entre ellos.

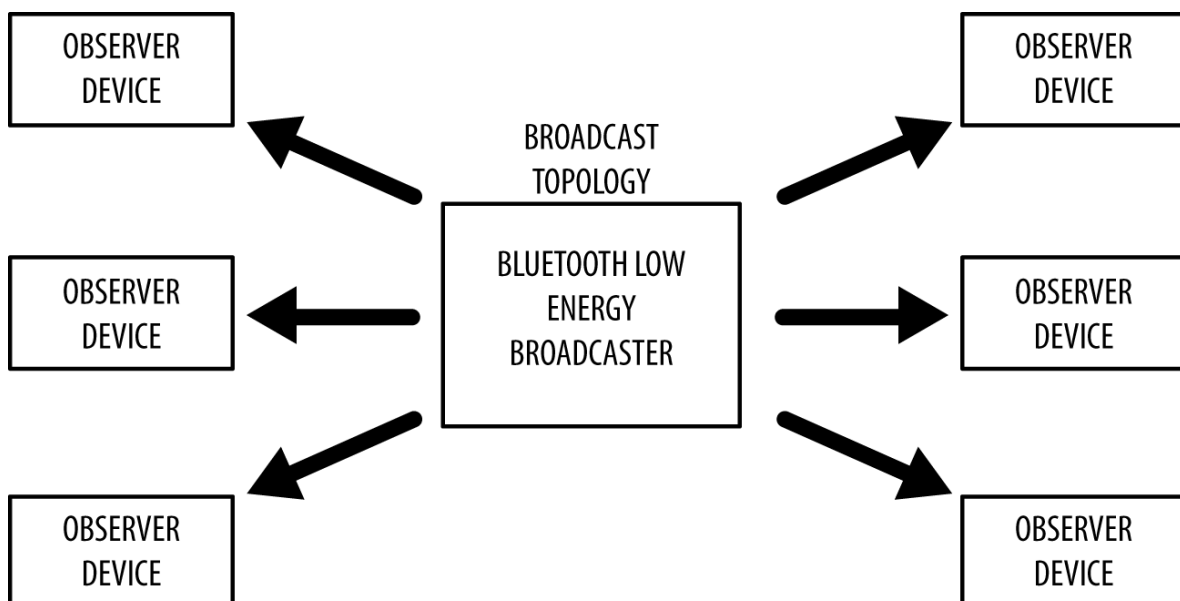


Figura 2.1.1 Comunicación Broadcast.



## 3 Metodología

En este capítulo se presenta el diseño del proyecto, conformado por sus partes de Hardware, de Firmware y de Aplicación.

### 3.1 Hardware

---

Una de las partes que conforman este proyecto es el hardware, que es la parte electrónica que se conecta directamente a la planta y al celular para enviar las lecturas de los sensores. En este apartado se describe el diseño e implementación del hardware propuesto.

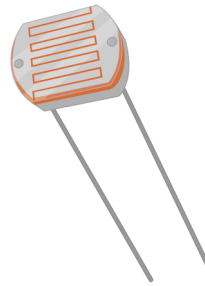
#### 3.1.1 Requerimientos

Para poder sensar las diferentes variables de las plantas y comunicarse con el celular se necesita de un dispositivo y una serie de sensores que midan temperatura, humedad de la tierra y la cantidad de luz, establecer la comunicación Bluetooth , un indicador para el estado del dispositivo y la batería para alimentar al sistema, por lo que se diseño un prototipo de un sistema embebido que incluya todos estos sensores y realice las tareas de comunicación y lectura de datos de los sensores.

De entre gran variedad de sensores existentes en el mercado, se eligieron 3 que son muy populares, los cuales se muestran en la figura 3.1.1. Estos tienen buena precisión, son económicos y fáciles de programar. Para medir la cantidad de luz se eligió una fotorresistencia, que es muy económica y es suficiente para medir qué tanta luz está recibiendo la planta. El sensor de humedad de la tierra es el sensor YL-69, este sensor se eligió gracias a que es muy fácil de remplazar, ya que es el sensor que más desgaste tiene al estar en constante contacto con la tierra, a parte de que es muy económico. Para medir la temperatura se usa el sensor DS18B20, un sensor digital de razonable precisión, bajo consumo y bajo costo, que por sus características es el adecuado para medir la temperatura ambiente. A parte de elegir los sensores por su bajo coste y popularidad, se escogieron por también ser de bajo consumo, lo que es muy benéfico para una aplicación que deba consumir poca energía.



Sensor de temperatura



Sensor de Luz



Sensor de Humedad

Figura 3.1.1 Sensores del sistema

Para la parte del microcontrolador se eligió el SOC (System On Chip) NRF51822 de la empresa Nordic Semiconductors® mostrado en la figura 3.1.2. Este cumple con todas las características y es el indicado para este sistema. El chip integra un procesador ARM, un módulo especializado para implementar el protocolo Bluetooth Smart, módulos analógicos y digitales para leer los sensores y consume una cantidad considerablemente baja de energía, con lo cual puede ser alimentado con baterías. Este chip a parte de ser muy potente es muy económico comparado con otros sistemas que utilizan varios componentes para cumplir la misma tarea. Otra razón por la que se eligió el chip es por sus herramientas de desarrollo, que son gratis bajo la licencia Open Source y no se necesita pagar nada para utilizarlas.



Figura 3.1.2 SOC NRF51822

Algo básico en un sistema es indicar al usuario el estado en el que se encuentra, para así en que estado se encuentra o para saber si está funcionando correctamente. Esta tarea esta a cargo de un Led RGB (Figura 3.1.3) integrado en el dispositivo, con el cual, mediante código de colores, indica al usuario lo que está pasando con el sistema.



Figura 3.1.3 Led RGB

Como todo el sistema esta diseñado para se de bajo consumo, la alimentación del para hacer funcionar todo el sistema se hace mediante una batería CR2032 (de tipo moneda) de 3v, mostrada en la figura 3.1.4, suficiente para alimentar al chip, sensores y led, teniendo una duración aproximada de 2 a 3 meses.



Figura 3.1.4 Batería CR2032

### 3.1.2 Diseño

El diseño del prototipo de hardware esta conformado por una pequeña placa de circuito impreso (PCB) sobre la cual se montan los sensores y el microcontrolador de tal forma que el sensor de humedad de tierra quede totalmente libre para ser insertado cerca de la planta. En la misma placa se agrega el porta pilas, el sensor de luz, el led RGB y un conector para el sensor de temperatura. A demás de los sensores se agrega un botón para encender el dispositivo y sensor después de entrar en estado de bajo consumo de energía.

Los sensores de luz y de humedad de tierra son analógicos, por lo que lleva una resistencia cada uno como componente externo para hacerlo funcionar y poder conectarlo al microcontrolador. El sensor de humedad mide solamente el porcentaje de humedad de la tierra, teniendo un rango representativo de humedad de 0% a 100%, igualmente el sensor de luz representa un nivel de iluminación de 0% a 100%.

El sensor de temperatura es digital y sólo necesita alimentación y un pin de entrada/salida del microcontrolador para hacerlo funcionar, mide la temperatura en grados centígrados (°C).

El led RGB se conecta a 3 salidas digitales del microcontrolador para controlar sus colores. Los diagramas de conexión de los sensores se muestran en la figura 3.1.5.

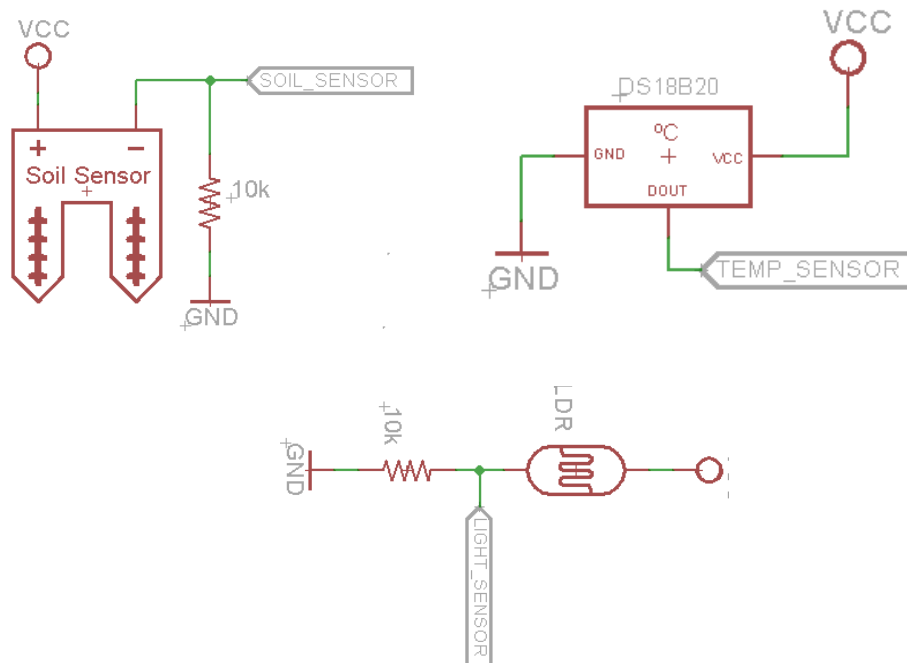
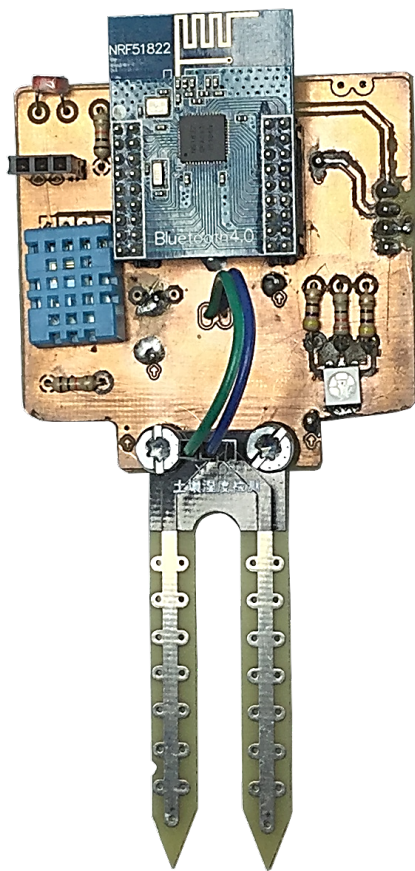


Figura 3.1.5 Diagramas de conexión.

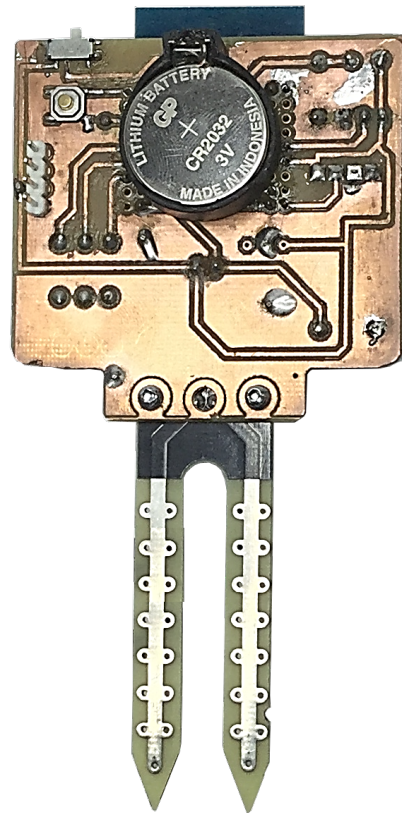
### 3.1.3 Implementación

Para la implementación del prototipo, se realizó el diseño y fabricación de una placa de circuito impreso para montar los componentes electrónicos para posteriormente soldar todos estos componentes en ella para probar que el diseño y los componentes no tuvieran problemas y todo funcionara correctamente.

Las imágenes del prototipo se muestran en la figura 3.1.6.



Vista Frontal del prototipo



Vista trasera del prototipo

Figura 3.1.6 Sensores del sistema

### **3.1.4 Pruebas**

Se realizaron varias pruebas con el prototipo para asegurar que el diseño y la placa no tuvieran errores y que los sensores funcionaran adecuadamente.

Entre las pruebas que se hicieron están las pruebas eléctricas, donde se prueba que todos los circuitos estén correctamente soldados al PCB, que no existan cortos circuitos, que los sensores no estén dañados, que el led encienda y que la alimentación sea adecuada.

Otras pruebas que se hicieron fueron de funcionamiento del microcontrolador, para verificar que los sensores estuvieran correctamente conectados al él y pudiera comunicarse con ellos.

Estas pruebas son esenciales y muy necesarias, ya que si el dispositivo no funciona correctamente será imposible programarle las tareas necesarias para la lectura de sensores y la comunicación Bluetooth, y por ende la aplicación en Android no se podría realizar ni probar correctamente.

## 3.2 Firmware

---

La segunda parte de este proyecto corresponde al firmware para el sistema embebido diseñado en la etapa de hardware. El firmware es el que se encarga de controlar todos los elementos de hardware para darle la funcionalidad necesaria al sistema. En este apartado se describe el desarrollo e implementación del firmware.

### 3.2.1 Requerimientos.

Para la etapa de desarrollo de firmware fueron necesarias herramientas para la programación y carga del firmware al SOC, las cuales son libres y multiplataforma. Estas herramientas se listan enseguida:

- **Lenguaje de programación C.** El lenguaje utilizado para generar el firmware fue el lenguaje C, con un compilador libre para arquitecturas ARM, corriendo en un sistema operativo Linux.
- **NRF5 SDK 10.** Conjunto de bibliotecas proporcionadas por la empresa Nordic para el manejo del protocolo Bluetooth 4.0.
- **Softdevice.** Es un firmware pre compilado, desarrollado por Nordic, el cual contiene el código ejecutable de las bibliotecas del protocolo bluetooth. Este debe ser cargado al SOC para su funcionamiento.
- **Herramienta de depuración de SEGGER.** Programador y depurador utilizado para cargar firmware al SOC, tanto el Softdevice como el firmware desarrollado. Esta herramienta es producida por la empresa SEGGER, y las utilerías necesarias para trabajar con ella pueden descargarse de su página de internet de manera gratuita.

### 3.2.2 Controladores Implementados.

Dentro de la etapa de desarrollo de firmware, se implementaron los controladores necesarios para cada uno de los sensores, generando una biblioteca para cada sensor, así como también, la programación del bluetooth para el envío de la información de los sensores y el control del LED RGB para notificar al usuario de diferentes situaciones.

### ➤ **Controlador para Sensor de luminosidad y Sensor de humedad.**

Para el sensor de luminosidad y el sensor de humedad, se utilizó el Convertidor analógico a digital (ADC) del SOC. Este fue configurado para trabajar a una resolución de 10 bits. Cada uno de estos sensores está conectado a una terminal del SOC, y este toma una cantidad específica de muestras del sensor y promedia los datos para obtener el valor del sensor, el cual es almacenado en una variable de 2 bytes para ser enviada posteriormente.

### ➤ **Controlador para sensor de temperatura.**

Para este sensor se utiliza el protocolo 1-Wire para comunicación con el SOC. Este protocolo requiere solo de una terminal del SOC para comunicarse, ya que lo hace de forma serial y los datos de temperatura son recibidos en formato digital.

### ➤ **Bluetooth.**

Dentro de las características que se programaron en el firmware para la comunicación por bluetooth, está la forma de enviar los datos, esto se hace en la etapa de “advertising” o publicidad del dispositivo, y es por esta característica que el sistema es no conectable, esto quiere decir que ningún otro dispositivo podrá establecer conexión con él. Esto implica que cualquier dispositivo que cuente con bluetooth 4.0 podrá visualizar los datos emitidos por el sistema, pero solo podrán interpretarse con la aplicación Android desarrollada, ya que estos datos se envían con una trama predefinida conocida por la aplicación.

Cada sensor tiene asignado un identificador, el cual es reconocido por la aplicación para leer el dato. En la tabla 5.2.1 se muestra la información de los identificadores.

*Tabla 5.2.1. Identificadores de sensores (bluetooth)*

Sensor	Identificador
Sensor de temperatura	UUID definido por Nordic.
Sensor de luminosidad	UUID = 0xDEDO
Sensor de humedad	UUID = 0xDADO



### 3.2.3 Diagrama de flujo.

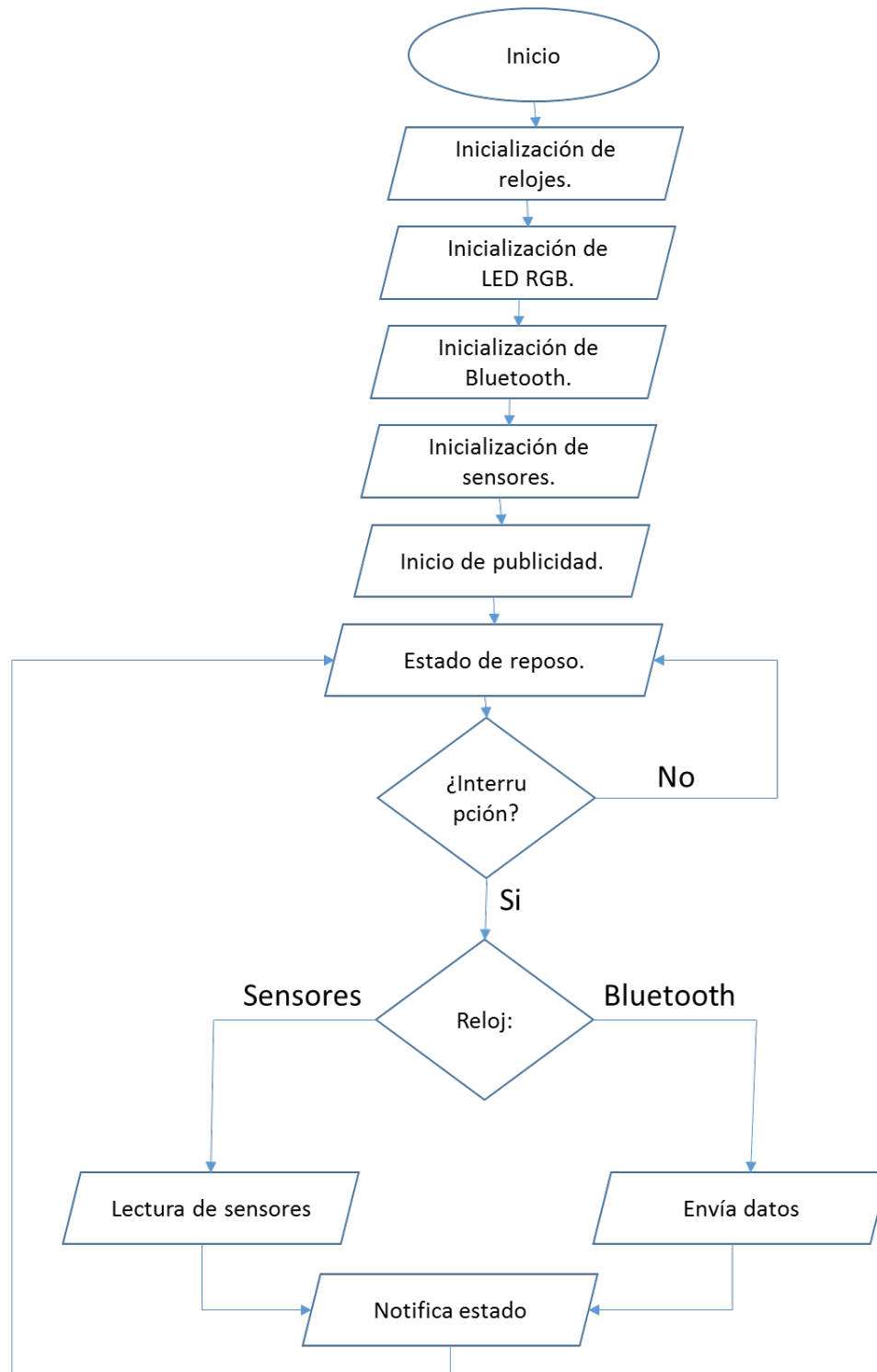


Figura 3.2.1. Diagrama de flujo del firmware

En la figura 3.2.1 se muestra el diagrama de flujo del firmware, en este diagrama se puede apreciar que el proceso comienza con la inicialización de los periféricos del SOC, como lo son los relojes, las terminales para el control del LED RGB, el protocolo Bluetooth y la inicialización para los sensores.

Una vez que se han inicializado los periféricos, el sistema comienza su etapa de advertising, y posteriormente entra en un estado de reposo en el que el consumo de energía se reduce al mínimo.

El firmware, inicializa 2 relojes, con los que sincroniza las actividades de lectura de sensores y el envío de los datos por bluetooth, estos relojes se configuran de tal manera que al transcurrir un tiempo definido, generan una interrupción provocando que el SOC salga del estado de reposo para realizar la acción que corresponde al reloj que genera la interrupción. Después de realizar la tarea, el sistema notifica al usuario lo que ha ocurrido, a través del LED RGB, con color verde se indica que el valor de los sensores se actualizó correctamente, con color rojo se indica que ocurrió un error con alguno de los sensores y con color azul se indica que el bluetooth se encuentra en funcionamiento.

Una vez que se termina ésta acción, el SOC regresa a su estado de reposo, en espera de otra interrupción.

### **3.2.4 Pruebas.**

Las pruebas realizadas al firmware, se hicieron separadas en 3 categorías, una por cada sensor. Para el sensor de luminosidad, se colocó el sensor en un ambiente de completa oscuridad, así como en un ambiente con abundante luz, para corroborar que efectivamente se diera un cambio en los datos transmitidos por el sistema.

Para el sensor de humedad se colocó el sensor, en tierra seca, y en agua, y de esta forma revisar que los datos transmitidos tuvieran cambios significativos, de acuerdo al ambiente en que trabaja el sensor.

Con el sensor de temperatura, se midió la temperatura ambiente, y se utilizó un encendedor como fuente de calor, para aumentar la temperatura del sensor y revisar que se diera el cambio en los datos de acuerdo al cambio de temperatura en el sensor.

Para realizar estas pruebas, se configuraron los relojes para que generaran interrupciones en lapsos de tiempo cortos, y así poder verificar con mayor facilidad el funcionamiento de los tres sensores.

## 3.3 Aplicación

---

### 3.3.1 Objetivo

El objetivo de la interfaz de usuario es crear un medio por el cual el usuario entrara en contacto el dispositivo, que facilite el cuidado de las plantas mediante indicadores visuales del estado de la humedad, temperatura, e iluminación

### 3.3.2 Descripción

La aplicación tiene cuenta con solo dos pantallas, esto tiene como objetivo mantener la simpleza del manejo, evitando componentes innecesarios (Figura 3.3.1).

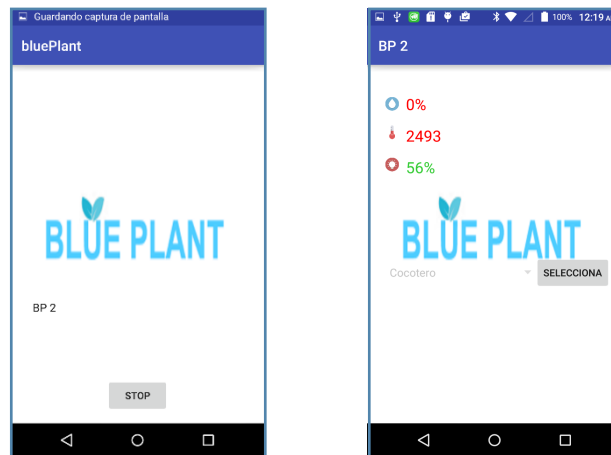


Figura 3.3.1 Interfaz de la aplicación.

### 3.3.3 Requerimientos

Requerimientos funcionales

- La aplicación debe de tener facilidad de uso
- Deberá adaptarse de acuerdo a el tipo de planta que se tenga
- La adquisición de datos se realizará por medio de Bluetooth
- La aplicación deberá recordar los dispositivos previamente utilizados
- La aplicación tendrá un catálogo de plantas en las que se incluya los valores aceptables de humedad, iluminación, y temperatura
- La aplicación deberá ser capaz de seleccionar una clase de planta para cada dispositivo, y en caso de ser requerido caviar la planta que se ha seleccionado
- La información deberá ser reflejada de manera que indique si los valores son aceptables

## Requerimientos del Smartphone o Tablet

- Sistema operativo Android 5.0 o superior
- Bluetooth 4.0

### 3.3.4 Diagramas

En la figura 3.3.2 se muestra el diagrama de la secuencia de uso de la aplicación BluePlant:

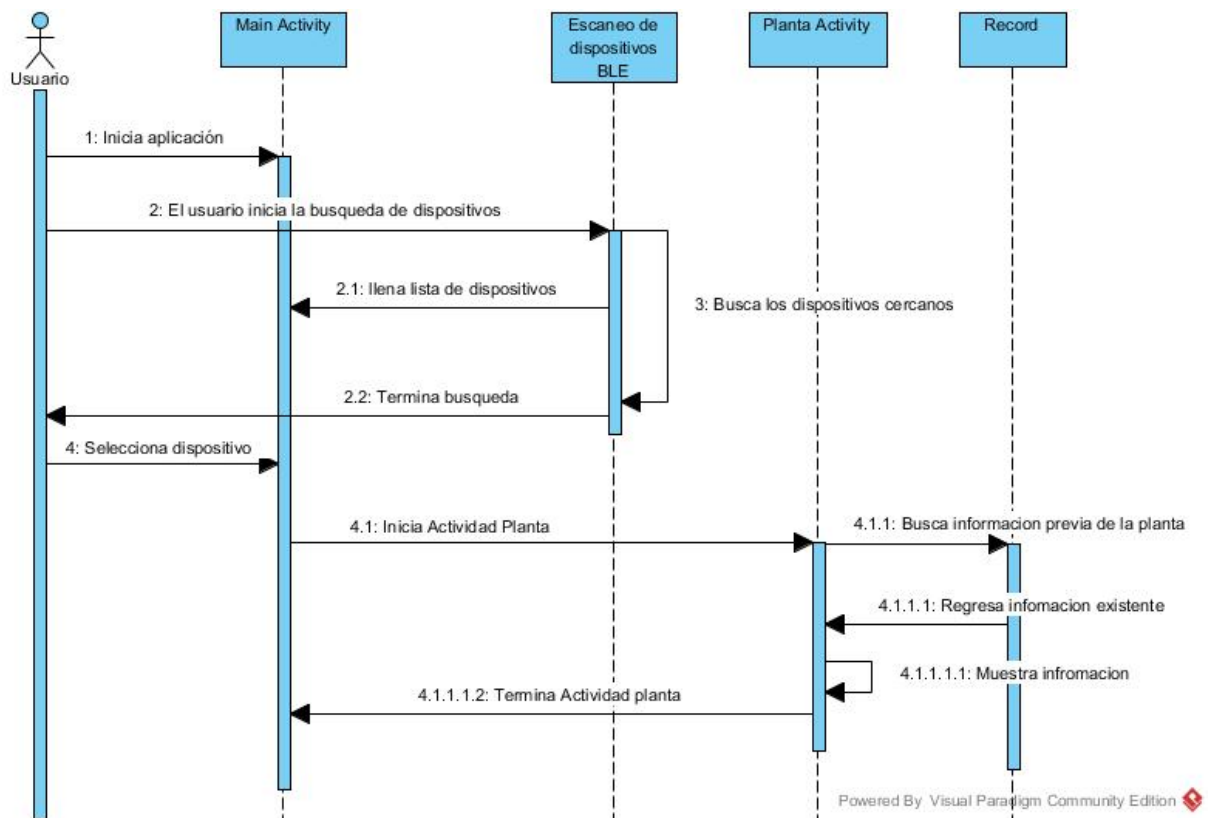


Figura 3.3.2 Diagrama de secuencia.

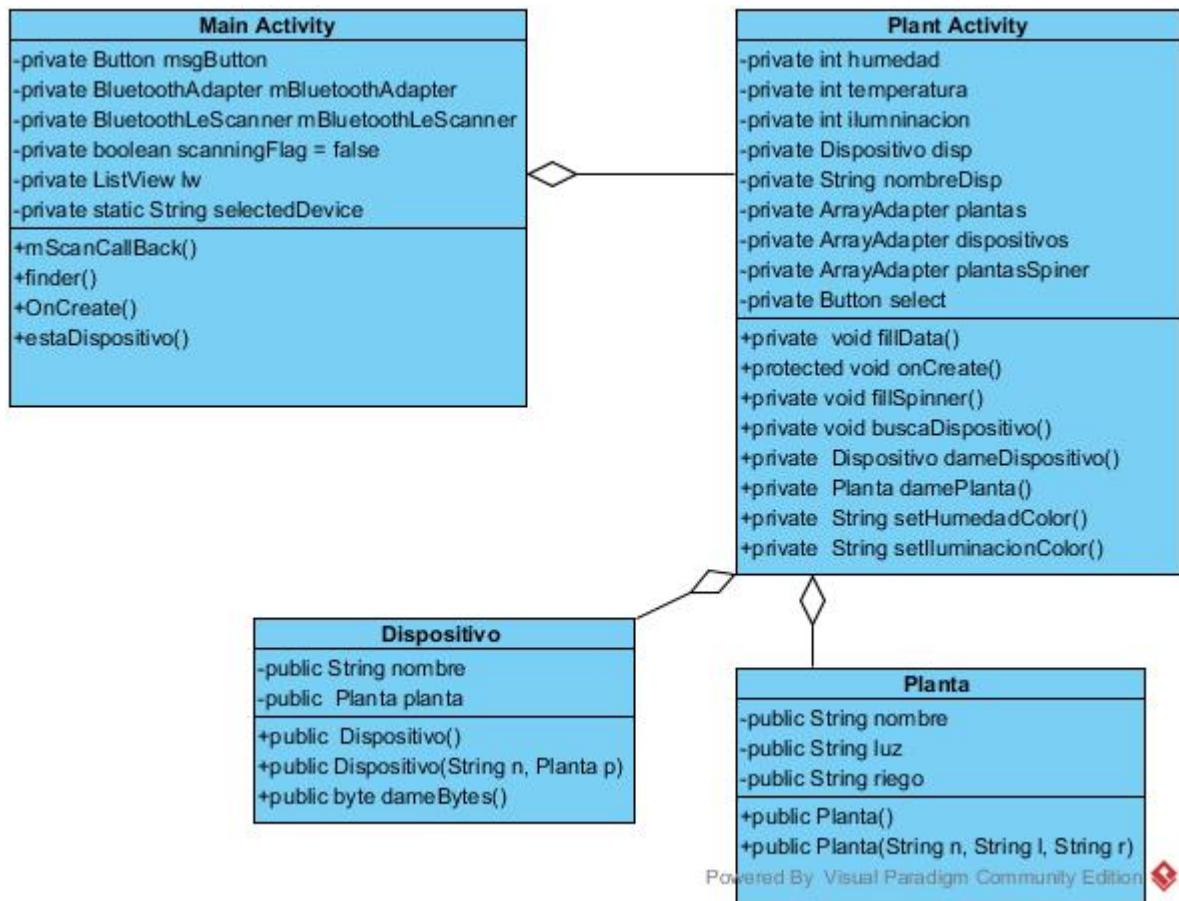


Figura 3.3.3 Diagrama de clases.

Casos de uso:

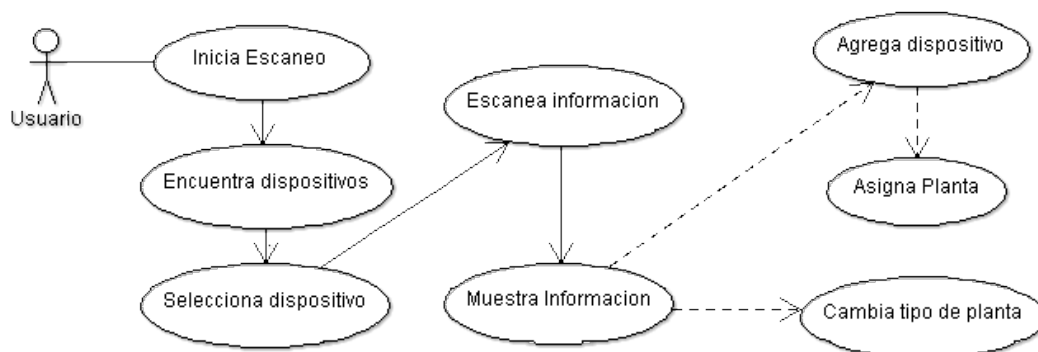


Figura 3.3.4 Diagrama de casos de uso.

### 3.3.5 App

Distribución:

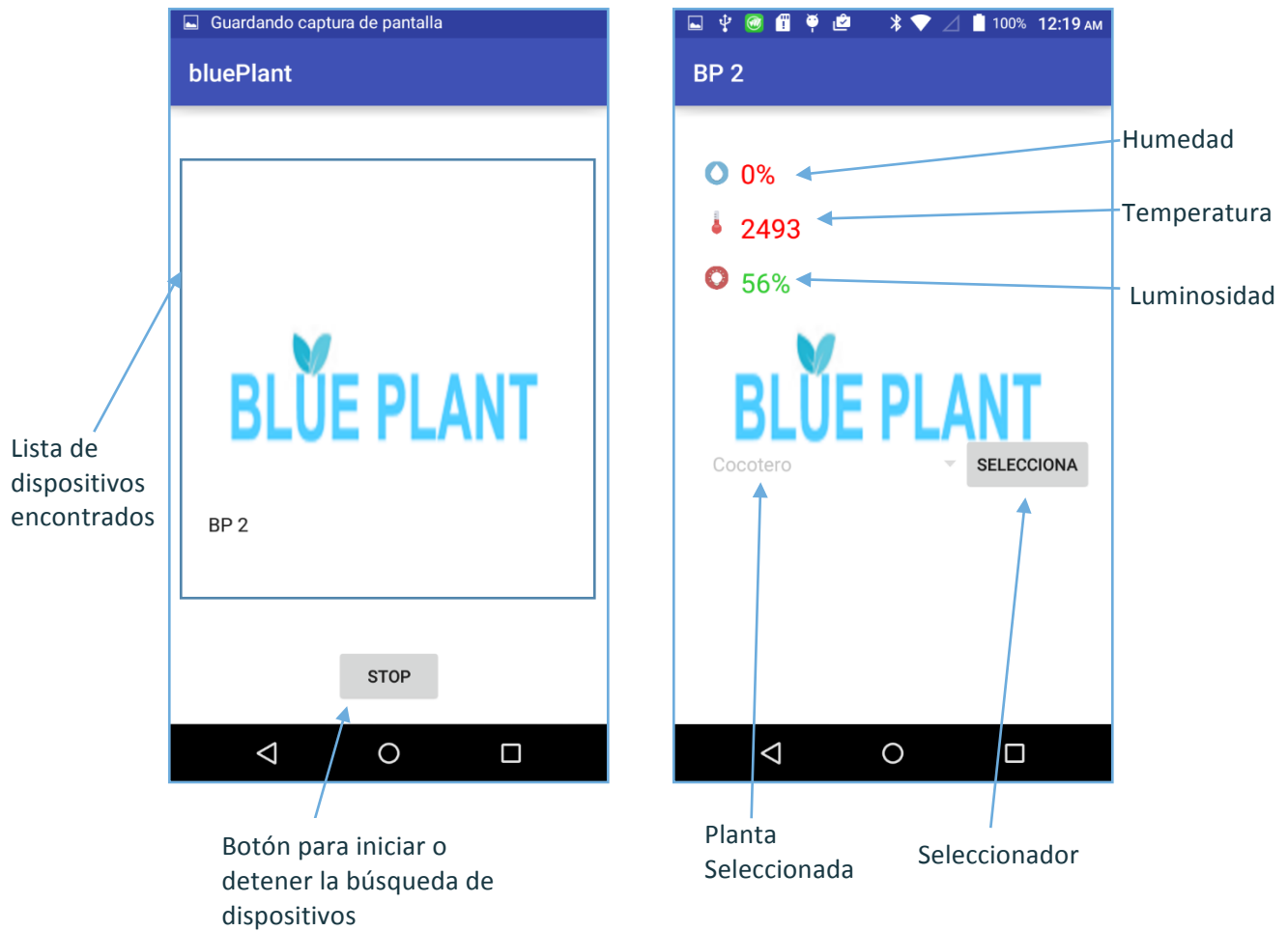


Figura 3.3.5 Distribución de los elementos de la aplicación.

Funcionamiento:

1. Iniciar la aplicación Blueplant
2. Presionar el botón ubicado en la parte inferior de la pantalla para iniciar la búsqueda de dispositivos cercanos



Figura 3.3.6 Búsqueda de dispositivos.

3. Los dispositivos aparecerán por encima de el botón.
4. Seleccionar el nombre del dispositivo o cualquier parte a la altura de el nombre del dispositivo para seleccionarlo.
5. Una vez seleccionado pasara a la siguiente pantalla, el significado de los valores ha sido definido en el punto anterior.



Figura 3.3.7 Valores de la planta.

6. En caso de que el dispositivo este siendo escaneado por primera vez por la Tablet o Smartphone el selectbox con las plantas aparecerá desbloqueado. Si el dispositivo ya fue escaneado antes solo presione el botón que dice “SELECCIONA” para liberar el selectbox y seleccionar una nueva planta, recuerda presionar nuevamente el botón “SELECCIONA” para guardar tu elección



Figura 3.3.8 Selección de planta.



## 4 Pruebas

Las pruebas realizadas se hicieron en una casa, un ambiente controlado en el cual se podía estar al pendiente de las mediciones a lo largo de las pruebas. Se le entrego a un usuario el dispositivo para su uso a lo largo de una semana, durante la cual se le dio la tarea de monitorear tres veces al día una planta apoyada por el dispositivo Blueplant. El dispositivo almaceno de manera automática los datos de las lecturas

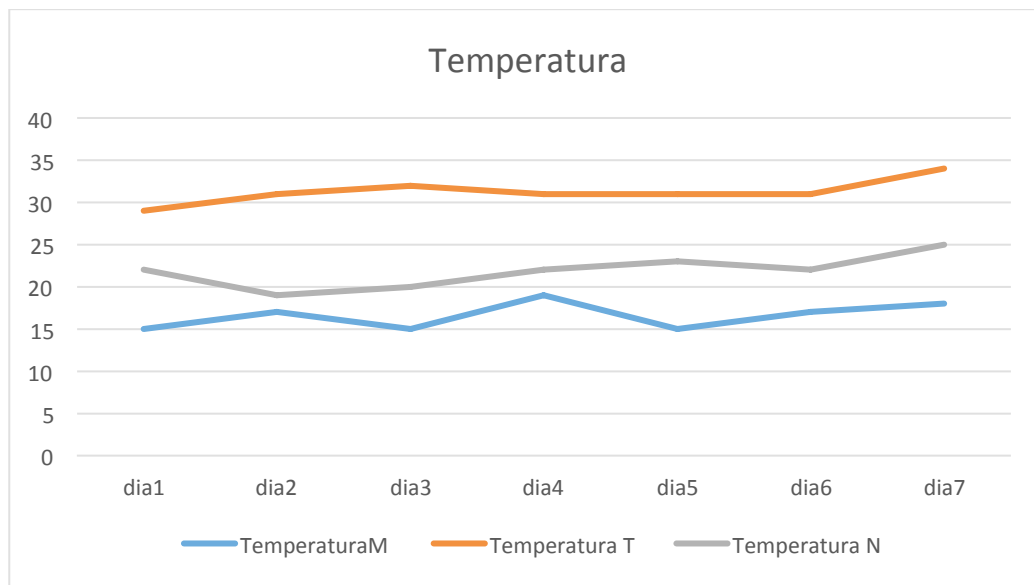


Figura 3.3.9 Grafica de temperatura de mañana tarde y noche.

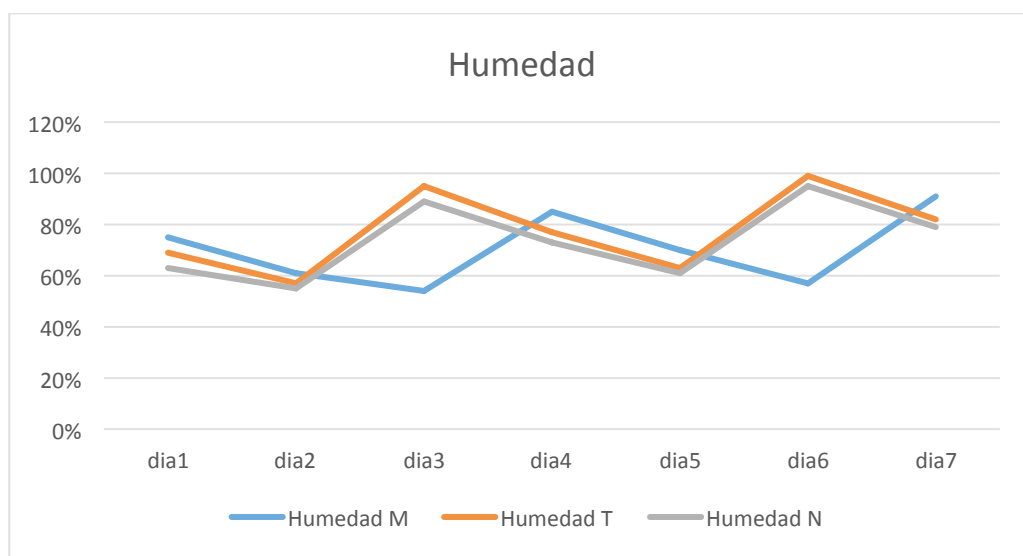


Figura 3.3.10 Grafica de humedad de mañana tarde y noche.

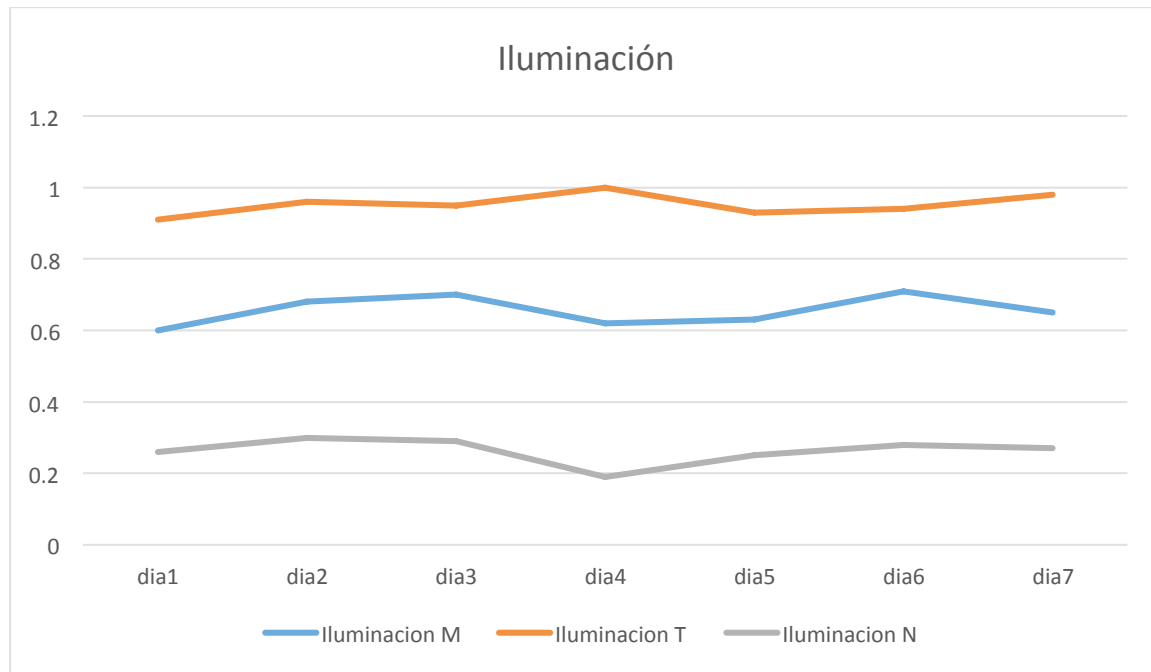


Figura 3.3.11 Grafica de iluminación de mañana tarde y noche.

Las gráficas muestran la humedad, la temperatura y la cantidad de iluminación de la planta puesta a prueba durante un periodo de una semana, tomando 3 mediciones cada día, una en la mañana otra en la tarde y la última en la noche.

El objetivo de las pruebas es evaluar el funcionamiento del dispositivo Blueplant y la experiencia de uso de la aplicación, tanto para estimar un buen estado de la planta, como para saber las condiciones de funcionamiento del sensor, así como para detectar los problemas que se pudieran presentar.

### Experiencia de uso

#### Pros

- La aplicación resulta fácil de usar
- El dispositivo es fácil de usar
- Los indicadores de uso son de gran ayuda

#### Contras

- No es resistente al agua
- Sería bueno tener una foto de la planta
- Le gustaría poder nombrar la planta/dispositivo

## 5 Costos

En este apartado se presentan los costos del desarrollo del prototipo, tanto de material como de ingeniería. Estos costos están en base al desarrollo de 2 prototipos de hardware.

Tabla 5.1.1 Costos de Material

Costos de Material			
Elemento	Cantidad	Precio unitario	Subtotal
Sensor de Humedad	2	\$25.00	\$50.00
Sensor de Temperatura	2	\$20.00	\$40.00
Sensor de Luz	2	\$3.00	\$6.00
SOC NRF51822	2	\$165.00	\$330.00
Resistencia	4	\$0.50	\$1.00
Push Button	2	\$1.00	\$2.00
Interruptor	2	\$2.00	\$4.00
Fabricación del PCB	2	\$45.00	\$90.00
Conectores	4	\$15.00	\$30.00
Programador JLink	1	\$130.00	\$260.00
Costos de envío	1	\$500.00	\$500.00
Total Material:			\$1313.00

Tabla 5.1.2. Costos de ingeniería.

Costos de ingeniería			
Razón	Horas trabajadas	Precio por hora	Subtotal
Diseño del Hardware	200	\$180.00	\$36,000.00
Programación del Firmware	200	\$180.00	\$36,000.00
Programación de la App en Android	200	\$180.00	\$36,000.00
Total Ingeniería:			\$108,000.00

El costo total del desarrollo del proyecto fue de **\$109,313.00** pesos MXN.

El costo de los materiales es relativamente bajo, si se toma en cuenta que sólo se desarrollaron un par de prototipos, y que en producción a alta escala, los costos disminuyen significativamente, abaratando el costo del producto final.

## 6 Conclusiones y trabajo a futuro

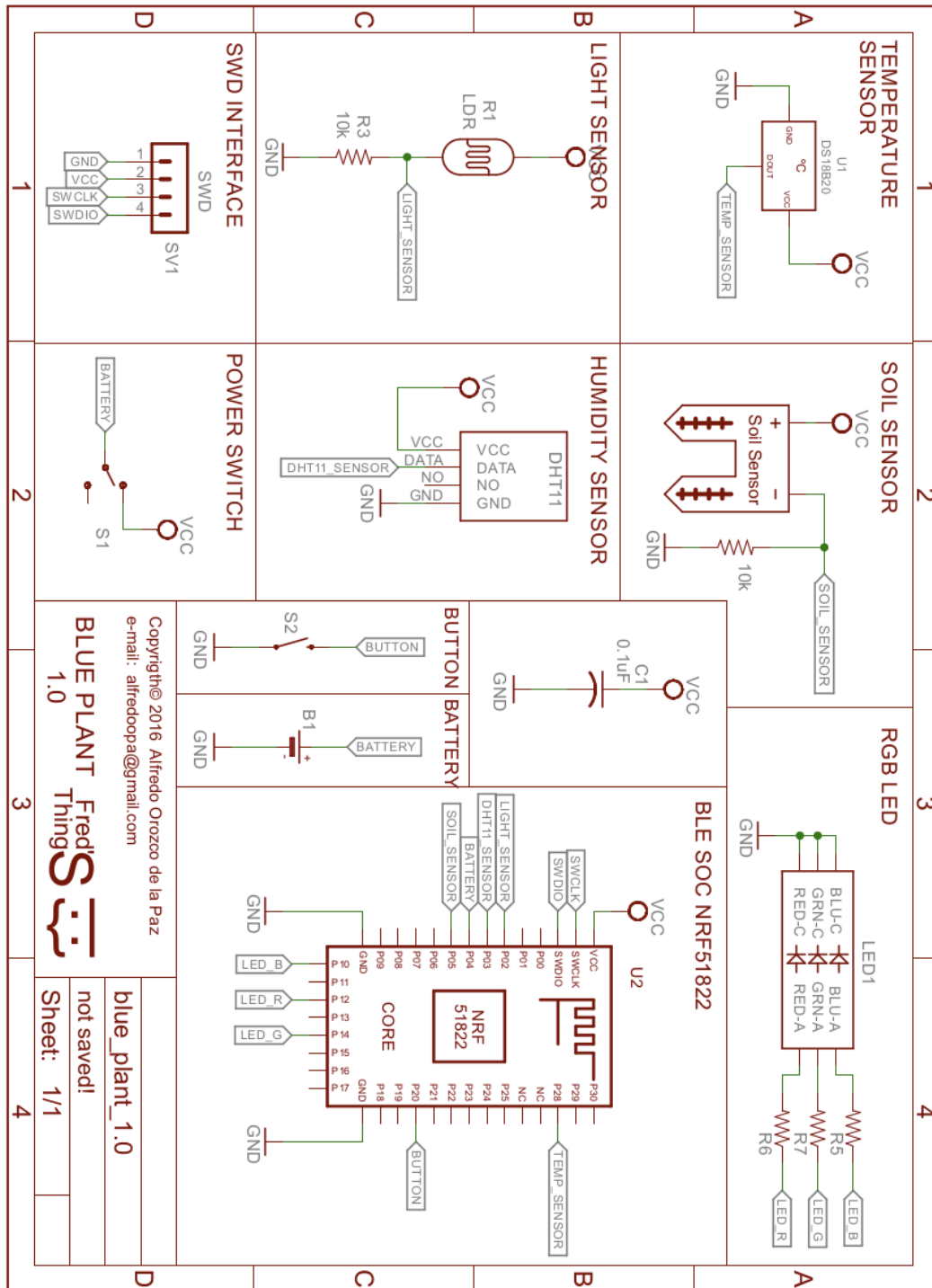
Como se muestra en este documento, el desarrollo del proyecto es la creación de un prototipo para el monitoreo de plantas en ambientes controlados, y aún se encuentra en fase de pruebas. Hemos detectado algunos problemas e inconvenientes con el diseño y a la vez se han encontrado posibles nuevas aplicaciones que pueden ayudar a resolver problemas en el Hardware y ampliar la funcionalidad para ser aplicado en otras áreas.

El trabajo a futuro propuesto para resolver los problemas e inconvenientes presentados y ampliar la funcionalidad es el siguiente:

- Diseñar una carcasa protectora impermeable para el hardware capaz de medir las variables externas. Esto debido a que el prototipo actual tiene todos sus componentes electrónicos expuestos a la intemperie, y esto es un gran problema ya que el dispositivo de hardware debe trabajar en ambientes con alto grado de humedad y con esta humedad los circuitos y componentes se pueden dañar.
- Crear una base de datos extensa de las plantas para darle a la aplicación móvil posibilidad de configurar una variedad más amplia de plantas
- Agregar la opción al hardware de conectarse a internet directamente mediante WiFi. Con esta nueva opción se abre un mundo de posibilidades, ya que el dispositivo se podría conectar a alguna red WiFi y enviar sus lecturas a través de internet o crear una red de dispositivos que sensen un área mayor de terreno.
- Programar el dispositivo para que sea configurable y que sea él quien lea los sensores y mande mensajes a la red de acuerdo a las necesidades de la planta, haciéndose más independiente de la aplicación móvil.
- Hacer que el dispositivo de hardware se recargue con energía solar. Sabiendo que el dispositivo consume muy poca energía, se puede agregar una batería recargable y un modulo de carga basado en celdas solares, con lo que se reduciría la contaminación causada por baterías y le daría gran autonomía al dispositivo sin gastar energía.

## 7 Anexos

### 7.1 Diagrama de esquemático del hardware



## 8 Referencias

### Enlaces:

*Android Developers*. (2016). *Developer.android.com*. Recuperado 1 June 2016, a partir de <http://developer.android.com/index.html>

*Develop Bluetooth Products and Apps | Bluetooth Technology Website*. (2016). *Bluetooth.com*. Recuperado 1 June 2016, a partir de <https://www.bluetooth.com/develop-with-bluetooth>

Systems, e. (2016). *Home - Ultra Low Power Wireless Solutions from NORDIC SEMICONDUCTOR*. *Nordicsemi.com*. Recuperado 1 June 2016, a partir de <http://nordicsemi.com>

### Bibliografía:

Galeano, G. *Programación de Sistemas Embebidos en C*. Alfaomega.

Townsend, K., Cufi, C., & Davidson, R. (2014). *Getting Started with Bluetooth Low Energy*. O'Reilly.

You, J. *The definitive Guide to ARM® Cortex®-M0 and Cortex-M0+ Processors* (2nd ed.). Nenes.

## 9 Glosario de términos

**Bluetooth:** Estándar de comunicación inalámbrica.

**BLE:** Abreviatura de Bluetooth Low Energy, también llamado Bluetooth Smart.

**Advertising:** Modo publicidad. Es el modo en que un dispositivo inalámbrico es visible para otros dispositivos.

**PCB:** Del ingles Printed Circuit Board, tarjeta de circuitos impresos.

**Firmware:** Programa informático que se graba internamente en la memoria de algún microcontrolador. Tambien llamado programa embebido.

**App:** Abreviatura usada para nombrar a los programas para dispositivos móviles.

**ARM:** Abreviatura de Advanced Risc Machine, arquitectura de procesadores diseñada por la empresa ARM.

**RGB:** Modelo de representación de colores en Rojo, Verde y Azul (Red, Green, Blue).

**SOC:** Abreviatura de System On Chip, es un chip que implementa todo un sistema dentro de él.

**Open Source:** Término usado para decir que algún programa, archivo, o algo es de libre distribución y no tiene costo hacer uso de ese algo.

**Sistema Embebido:** Sistema computacional diseñado para realizar pocas tareas dedicadas de propósito específico.

