

**ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**

**FACULTAD DE INGENIERIA EN ELECTRICIDAD Y COMPUTACIÓN**

**ADMINISTRACIÓN DE SERVICIOS DE RED BAJO LINUX (TICG1002)**

**Proyecto de ingeniería usando tecnologías basadas en Linux:**

Prototipo de un solmáforo para la prevención de enfermedades cutáneas producidas por altos índices de radiación ultravioleta

**Grupo #1**

**Integrantes:**

* Juan Pablo Cadena Aguilar
* Angélica Victoria Fajardo Neira
* Jimmy René Gilces Vargas
* Josue Vicente Martínez Salazar

**Docente: Msig. Adriana Elisa Collaguazo Jaramillo**

**Fecha de inicio y fin del trabajo:** desde 4/12/2018 hasta 31/1/2019

**Objetivos de Aprendizaje:**

1. Experimentar servicios de red de forma integrada en el sistema operativo Linux para la automatización de tareas básicas de operación, mantenimiento y gestión de los servicios de red.

2. Utilizar la metodología de diseño e implementación usando plataformas de prototipo bajo Linux para el análisis de los indicadores de la red.

3. Aplicar los mecanismos de seguridad a nivel de paquetes para el aseguramiento de los servicios de red.

**Introducción:**

**Contexto y justificación del Trabajo.**

En la actualidad no es raro escuchar en las noticias o medios de información una creciente expectativa, o preocupación acerca de los efectos causantes de la radiación emitida por el sol. Radiación que puede ser medida o catalogada en varios niveles de peligrosidad según su incidencia, estando relacionada con diversos tipos de afectaciones o enfermedades a la piel según el tiempo y la exposición que tengamos con la luz del sol.

Existen pocos mecanismos bajo los cuales se pueda medir este nivel de radiación, almacenando y mostrando los resultados de manera explicita e implícita, pudiendo ser un factor determinante para el cuidado y conciencia a nivel personal y social.

**Objetivos del Trabajo.**

Nuestro objetivo es desarrollar un prototipo capaz de medir el nivel de radiación y almacenarlos junto con otros parámetros en un servidor de base de datos, para que puedan ser visualizados de manera posterior y al mismo tiempo se presente bajo que nivel de radiación nos encontramos al momento de hacer la toma de datos, haciendo uso de leds para la representación correspondiente.

**Idea General.**

La idea general es realizar un solmáforo, para esto es necesario un panel solar y haciendo uso de una Raspberry se levantará un servidor y se capturaran los valores UV que inciden en el panel. Por otro parte se implementará en un Arduino un modulo para sensar valores de temperatura y humedad, dichos valores son importantes para poder medir el resultado final, que se enviarán a una base de datos para ser almacenados y se verán presentado en los focos que conforman el solmáforo. Indicando así cuál es el nivel de radiación al que nos estamos exponiendo.

**Componentes utilizados.**

**Raspberry pi 3 b+**

El Raspberry es una SBC de bajo costo, que brinda a los usuarios una gran variedad de funciones u opciones para realizar un prototipado que se adapta perfectamente a casi cualquier tipo de necesidad. A través de este dispositivo podemos implementar nuestro servidor para el almacenamiento y distribución de datos obtenidos por los sensores y al mismo tiempo serán enviados en una base de datos.

En cuanto al modelo de Raspberry utilizado para nuestra implementación, este nuevo micrordenador ha supuesto un rediseño absoluto de la placa, manteniendo el mismo tamaño y la misma posición de los elementos que en el modelo Pi 3 pero ha cambiado el procesador por otro más potente que funciona a 1.4 GHz, y además elimina el cuello de botella de la conectividad incluyendo Bluetooth 4.2, BLE, Wi-Fi a doble banda 2.4 GHz y 5 GHz y, además, la tarjeta de red, Gigabit Ethernet, ya no está limitada a los 100 Mbps, sino que es capaz de alcanzar los 300 Mbps al funcionar sobre USB 2.0.

Sus características son:

* CPU + GPU: Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz
* RAM: 1GB LPDDR2 SDRAM
* Wi-Fi + Bluetooth: 2.4GHz y 5GHz IEEE 802.11.b/g/n/ac, Bluetooth 4.2, BLE
* Ethernet: Gigabit Ethernet sobre USB 2.0 (300 Mbps)
* GPIO de 40 pines
* HDMI
* 4 puertos USB 2.0
* Puerto CSI para conectar una cámara.
* Puerto DSI para conectar una pantalla táctil
* Salida de audio estéreo y vídeo compuesto
* Micro-SD
* Power-over-Ethernet (PoE)



**Arduino Uno Rev2**

El Microcontrolador Arduino Yún rev2 USB es un sistema basado en Linux que permite conexiones y aplicaciones de red avanzadas. La conexión a su WiFi o red cableada es simple gracias al Panel Web de Yún y al boceto dedicado de '' YunFirstConfig ''.

Este microcontrolador es el que nos va a permitir realizar las operaciones respectivas luego de censar los datos a través de los módulos correspondientes de humedad y temperatura, en interacción con un panel solar. También se realizará la conexión con la base de datos creada para el almacenamiento de la información y los parámetros obtenidos.

Sus especificaciones son:

Técnicas

* Microcontrolador: ATmega32U4
* Voltaje de Operación: 5V
* Voltaje de Entrada: 5V
* Pines de E/S Digitales: 20
* Salida PWM: 7
* Pines Analógicos de E/S: 12
* Corriente DC por Pin E/S: 40 mA en Pines E/S; 50 mA en 3,3 Pin
* Memoria Flash: 32 KB (de los cuales 4 KB son utilizados por el gestor de arranque)
* SRAM: 2,5 KB
* EEPROM: 1 KB
* Velocidad del Reloj: 16 MHz

Microprocesador Linux

* Procesador: Atheros AR9331
* Arquitectura: MIPS
* Voltaje de Operación: 3,3V
* Ethernet: 802.3 10/100Mbit/s
* WiFi: 802.11b/g/n 2.4 GHz
* Tipo de USB: host 2.0
* Lector de Tarjetas: Micro-SD
* RAM: 64 MB DDR2
* Memoria Flash: 16 MB
* Velocidad del Reloj: 400 MHz



**ESP8266 (modulo wifi)**

El ESP8266 es un microprocesador de bajo coste con Wifi integrado fabricado por Espressif. Podemos usar el ESP8266 para conectar nuestros proyectos de electrónica y robótica con Arduino. Se trata de un chip integrado con conexión WiFi y compatible con el protocolo TCP/IP. El objetivo principal es dar acceso a cualquier microcontrolador a una red.

A través de este módulo se establece la conexión con la base de datos y el servidor implementado, controlando así el flujo de la información enviada y recibida desde el Arduino.

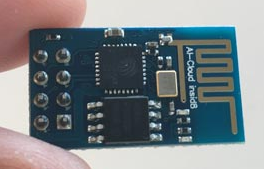
Entre sus especificaciones más importantes tenemos:

Hardware

* Utiliza una CPU Tensilica L106 32-bit
* Voltaje de operación entre 3V y 3,6V
* Corriente de operación 80 mA
* Temperatura de operación -40ºC y 125ºC

Conectividad

* Soporta IPv4 y los protocolos TCP/UDP/HTTP/FTP
* No soporta HTTPS en un principio. Si que lo hace mediante software tanto en cliente como servidor TLS1.2.

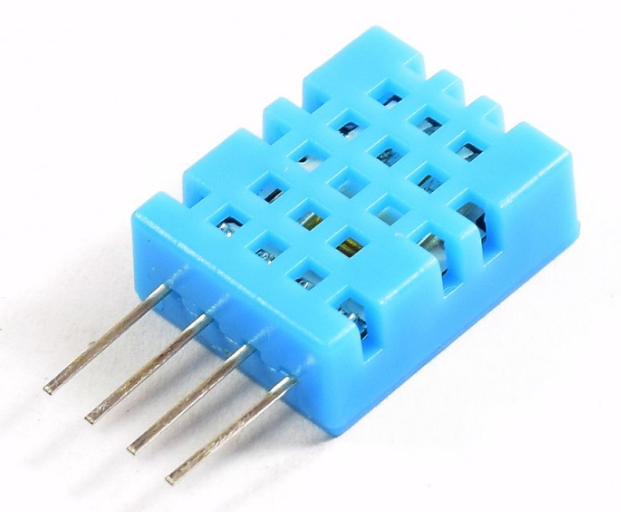


**DHT11 (sensor humedad y temperatura)**

El DHT11 es un sensor digital de temperatura y humedad relativa de bajo costo y fácil uso. Integra un sensor capacitivo de humedad y un termistor para medir el aire circundante, y muestra los datos mediante una señal digital en el pin de datos

Utilizar el sensor DHT11 con las plataformas Arduino/Raspberry Pi/Nodemcu es muy sencillo tanto a nivel de software como hardware. A nivel de software se dispone de librerías para Arduino con soporte para el protocolo "Single bus". En cuanto al hardware, solo es necesario conectar el pin VCC de alimentación a 3-5V, el pin GND a Tierra (0V) y el pin de datos a un pin digital en nuestro Arduino.

* Voltaje de Operación: 3V - 5V DC
* Rango de medición de temperatura: 0 a 50 °C
* Precisión de medición de temperatura: ±2.0 °C
* Resolución Temperatura: 0.1°C
* Rango de medición de humedad: 20% a 90% RH.
* Precisión de medición de humedad: 5% RH.
* Resolución Humedad: 1% RH
* Tiempo de sensado: 1 seg.
* Interface digital: Single-bus (bidireccional)
* Dimensiones: 16\*12\*5 mm
* Peso: 1 gr



**Panel Solar.**

Un panel solar o módulo solar es un dispositivo que capta la energía de la radiación solar para su aprovechamiento. Los paneles fotovoltaicos están formados por numerosas celdas que convierten la luz en electricidad. Las celdas a veces son llamadas células fotovoltaicas. Estas celdas dependen del efecto fotovoltaico por el que la energía lumínica produce cargas positiva y negativa en dos semiconductores próximos de diferente tipo, produciendo así un campo eléctrico capaz de generar una corriente.

A través de la corriente generada por nuestro panel y por el voltaje obtenido del mismo podemos calcular de manera implícita los valores necesarios para realizar los cálculos dentro de la programación correspondiente de nuestro prototipo.

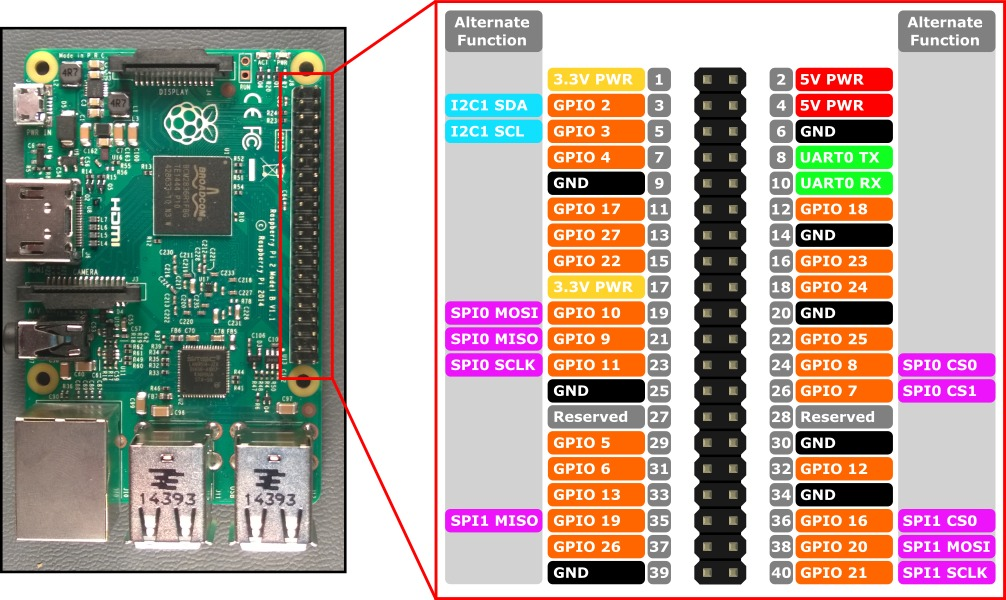


**Realización y Funcionamiento.**

El funcionamiento básico de nuestro solmáforo consiste en mostrar por medio de un foco led, el nivel o intervalo de peligrosidad de los rayos incidentes en tiempo real capturados por nuestro prototipo. Cada foco diferenciado por color está directamente relacionado con el un nivel de radiación como se indicará en la siguiente tabla.

|  |  |
| --- | --- |
| **Color** | **Nivel de peligrosidad** |
| Verde | Bajo |
| Amarillo | Moderado |
| Naranja | Alto |
| Rojo | Muy alto |
| Azul (morado) | Extremadamente alto |

A través del código implementado dentro de la Raspberry, se asignan los pines 17, 27, 22, 6 y 13 para el funcionamiento correspondiente dentro de la tabla gpio para nuestro prototipado.



Por medio del panel solar se obtiene el factor UV de radiación, haciendo uso de “web scraping” la cuál es una técnica para extraer información de una página web, donde se encuentran los valores y niveles relacionados. Esta información se actualiza en la última fila de base de datos.

El Arduino Uno, se encarga de sensar la temperatura y la humedad a través del **DHT11** y procede a enviarlo de igual manera a la base de datos. Por medio de la programación respectiva es la recopilación de los valores se muestra en consola, y también dentro de un pequeño circuito donde se implementan los leds con indicativo de color.

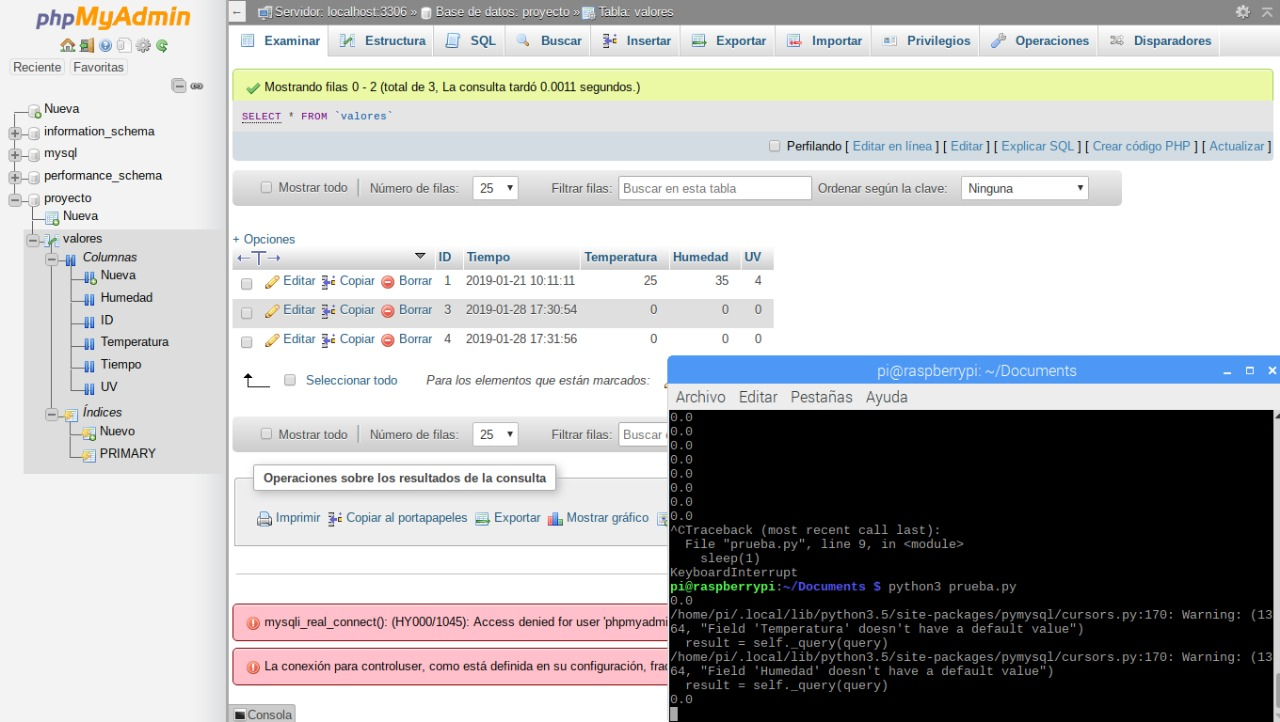


Ilustración 1 Base de Datos creada para almacenar la información recopilada

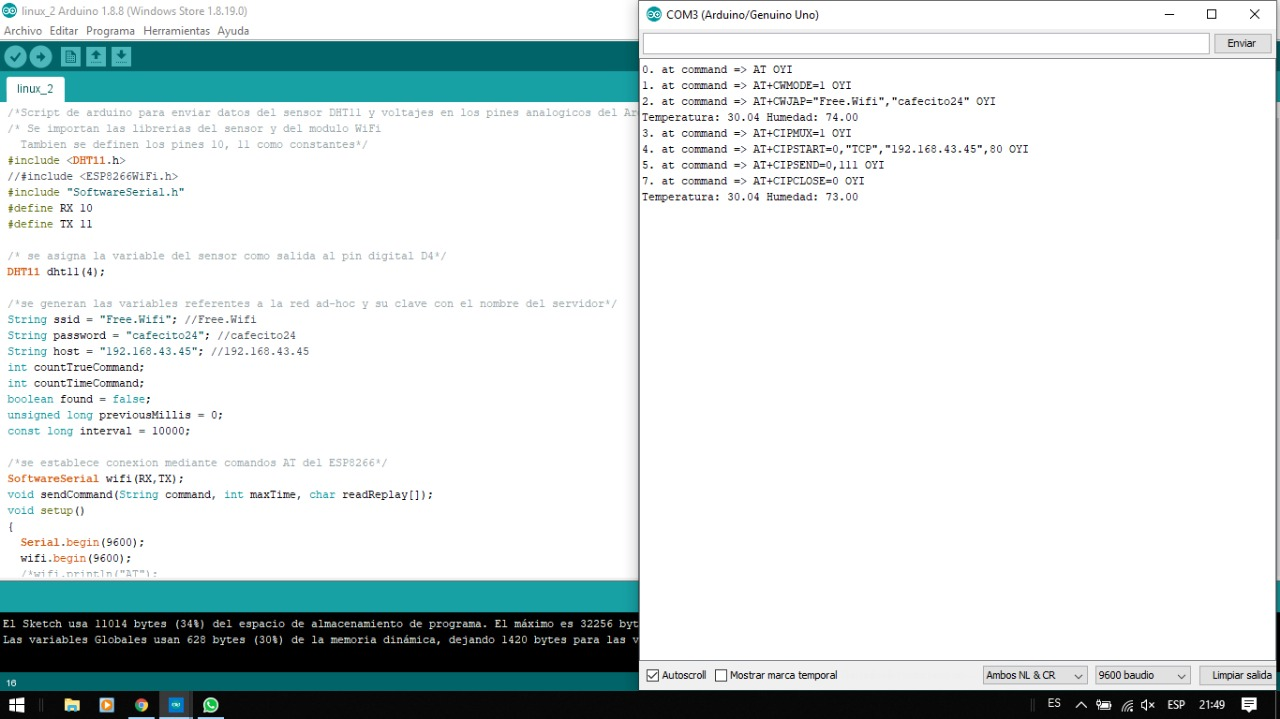


Ilustración 2. Presentación por consola de los datos obtenidos.

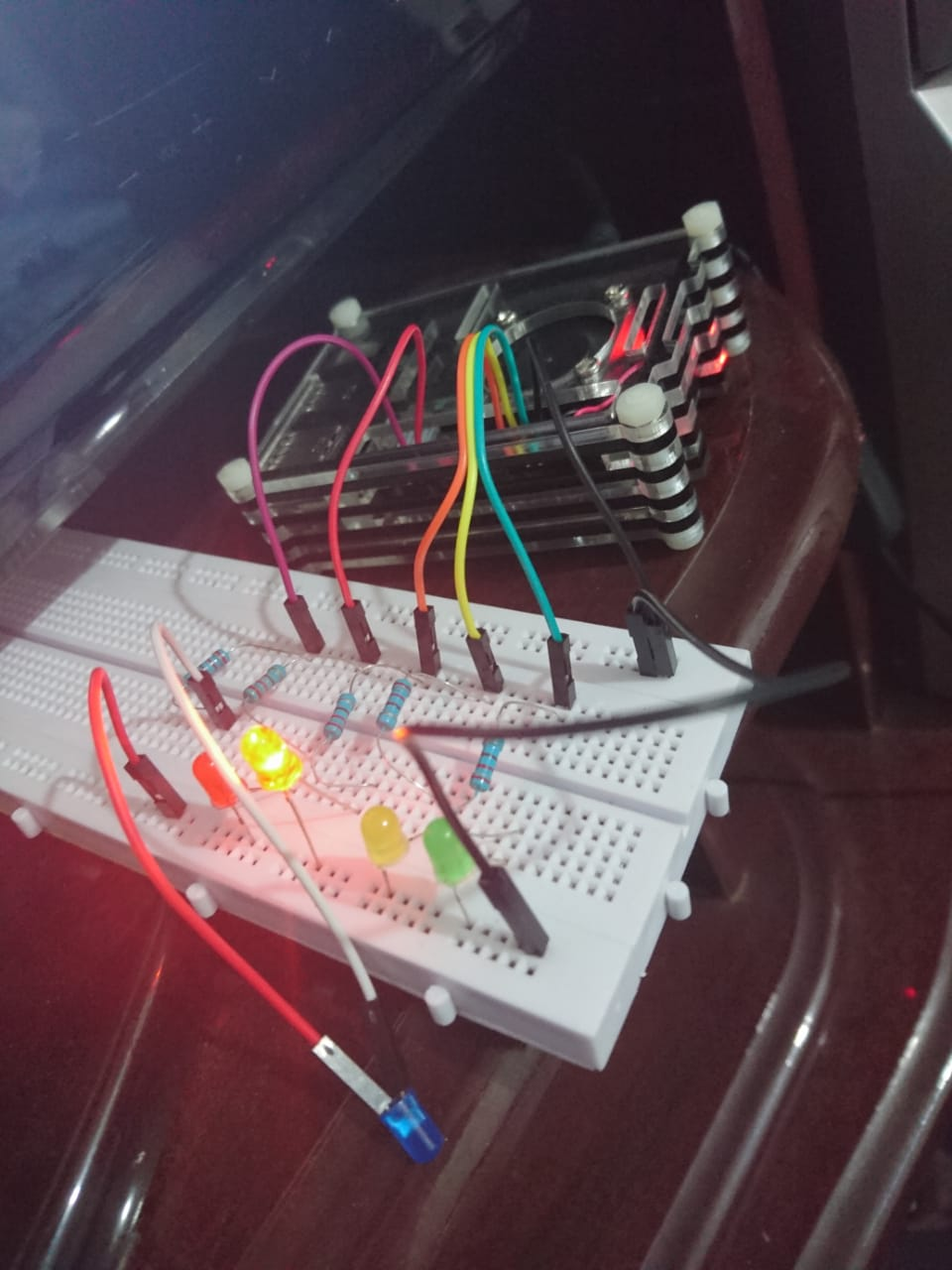


Ilustración 3. Funcionalidad del solmáforo con todos los componentes conectados.

**Conclusiones:**

* La realización de scripts es una gran estrategia para la automatización de procesos y configuración de servicios en servidores Linux.
* Se realizo un correcto uso y creación de una base de datos para poder almacenar la información.
* Se estableció un servidor a través de una Raspberry para poder acceder a la información recopilada por nuestro solmaforo.
* Se pudo establecer una conexión entre todos los dispositivos utilizados para este prototipo dando como resultado una buena implementación y solución del problema planteado inicialmente, el cual consistía en visualizar el nivel de radiación UV

**Recomendaciones:**

* Reconocer el correcto uso de comillas simples y dobles para no cometer errores dentro del script.
* Revisar la funcionalidad y asignación correcta de los pines de los diferentes módulos y plataformas de prototipado utilizados.
* Realizar las pruebas respectivas para evaluar el nivel de efectividad o aceptación del prototipo propuesto.
* Validar todas las funciones realizadas dentro de la programación.