**MANUEL DE USUARIO**

Proyecto: Vigilia

**CONTENIDO**

**Contenido**

**1 Presentacion................................................................................................................**

**2 objetivo…………..........................................................................................................**

**3**

**Requisitos del sistema………….................................................................................**

**3.1**

**Requerimientos de hardware.....................................................................................**

**3.2**

**Requerimientos de software......................................................................................**

**4**

**Herramientas utilizadas para el desarrollo…............................................................**

**5**

**Instalación de aplicaciones….....................................................................................**

**6 diagramas.....................................................................................................................**

**7**

**Base de datos...............................................................................................................**

**8**

**Componentes…………….............................................................................................**

**9**

**Secuencia del proyecto…………………………………………………………………….**

**1. Presentación**

El siguiente manual guiará a los usuarios que harán soporte al sistema, el cual les dará a conocer los requerimientos y la estructura para la construcción del sistema, en el desarrollo de programa de escritorio y aplicativo móvil conectados mediante una base de datos en la nube, el cual muestra las herramientas necesarias para la construcción y la funcionalidad del sistema.

**2. Objetivo**

Informar y especificar al usuario la estructura y conformación del sistema con el fin de que puedan hacer soporte y modificaciones o actualizaciones al sistema en general.

**3. Requisitos del sistema**

· **3.1** **Requerimientos de hardware**

PC, teclado, mouse, monitor, dispositivo móvil. Memoria RAM 2 GB (equipo y dispositivo móvil) y Procesador 1.4 GHz.

· **3.2 Requerimientos de software**

Sistema operativo (Windows 7 en adelante). Sistema operativo móvil (Android 5.0. en adelante o cualquiera de IOS) Conexión internet local y móvil.

**4. Herramientas utilizadas para el desarrollo**

**IDE Arduino**

El entorno de desarrollo integrado (IDE) de Arduino es una aplicación multiplataforma (para Windows, macOS, Linux) que está escrita en el lenguaje de programación Java. Se utiliza para escribir y cargar programas en placas compatibles con Arduino, pero también, con la ayuda de núcleos de terceros, se puede usar con placas de desarrollo de otros proveedores.

**ThingSpeak**

ThingSpeak es un software de código abierto escrito en Ruby que permite a los usuarios comunicarse con dispositivos habilitados para Internet. Facilita el acceso a los datos, la recuperación y el registro de datos al proporcionar una API tanto a los dispositivos como a los sitios web de redes sociales.

**Blynk**

Blynk es una plataforma que permite que cualquiera pueda controlar fácilmente su proyecto Arduino con un dispositivo con sistema iOS o Android. Los usuarios tendrán ahora la posibilidad de crear una interfaz gráfica de usuario de “arrastrar y soltar” para su proyecto en cuestión de minutos y sin ningún gasto extra.

**5. Instalación de aplicaciones**

**Requisitos generales preinstalación para el sistema de escritorio**

Para ejecutar el programa de escritorio se necesita de Arduino IDE 1.8.16 instalado con las siguientes características para la ejecución del programa de escritorio.

Soporte en procesador Intel 1.4.0 GHz entre otros.

Memoria RAM 2 GB.

Espacio en disco: 124 MB

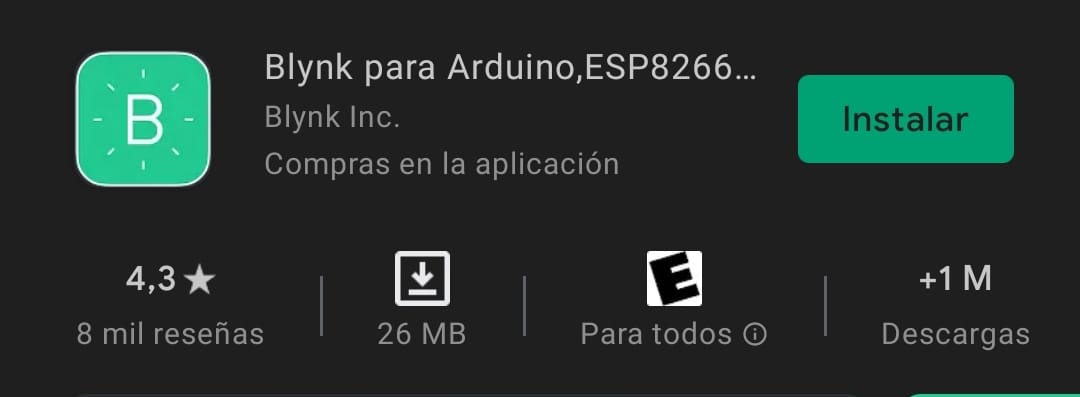
El programa se descarga del siguiente enlace en la página oficial de Arduino IDE: <https://www.arduino.cc/en/software>



Carga y ejecución del código. Luego de tener el IDE, descarga el archivo del link que se muestra a continuación: <https://github.com/Mar2SK/Vigilia>

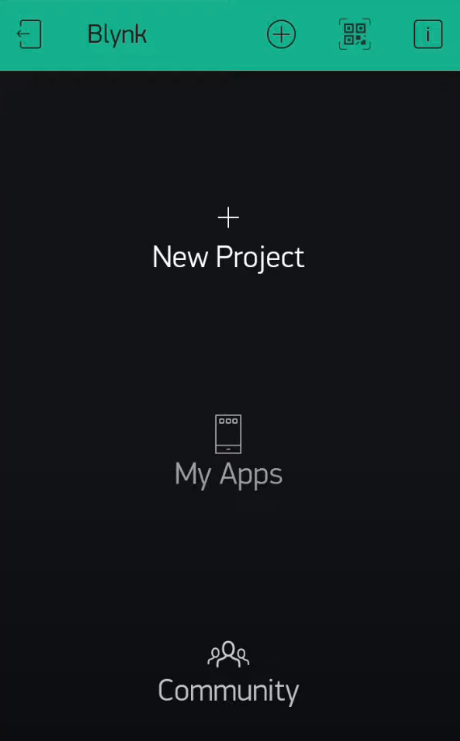
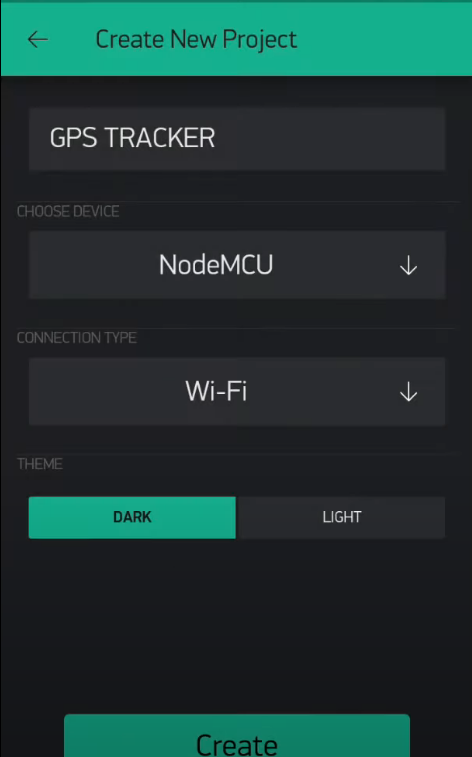
Una vez instalado copiar el código del archivo y pasar a hacer “*Copy Paste*” en la aplicación. En el link se especifican las librerías y la preferencia de la placa.

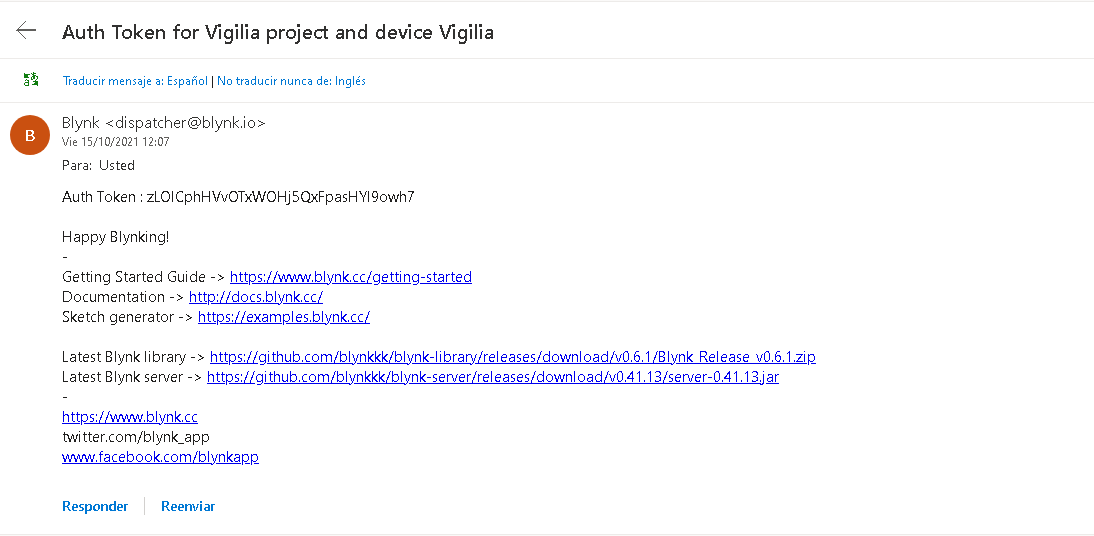
**Instalación de aplicativo móvil en un teléfono inteligente**

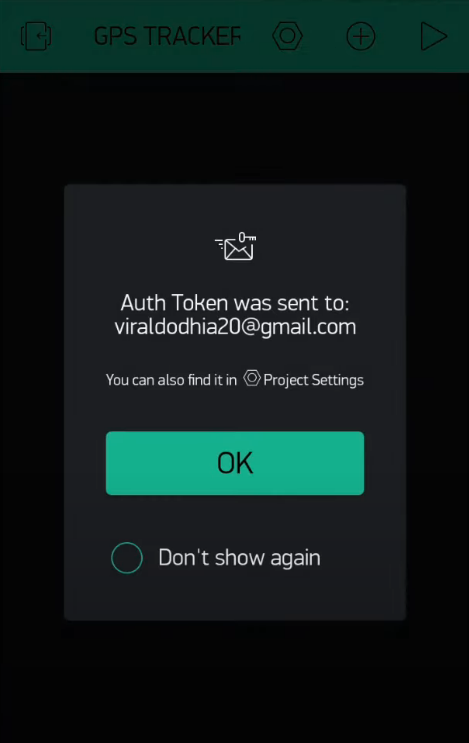


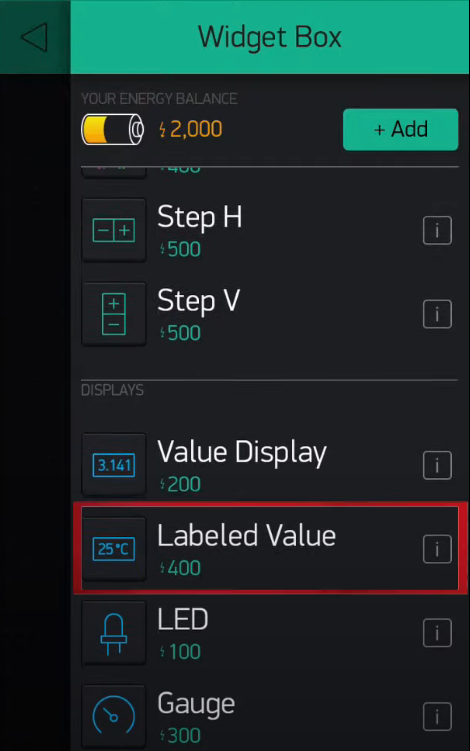
Para la instalación del aplicativo móvil en un teléfono se debe tener un sistema operativo móvil (Android 8.0. en adelante Y IOS) y conexión a internet móvil para el uso del aplicativo. Para este proceso vamos a descargarlo desde app store o play store

Una vez instalado, se procede a iniciar la aplicación para configurarla y arrancar a utilizarla. vamos a new Project y colocamos lo siguiente:

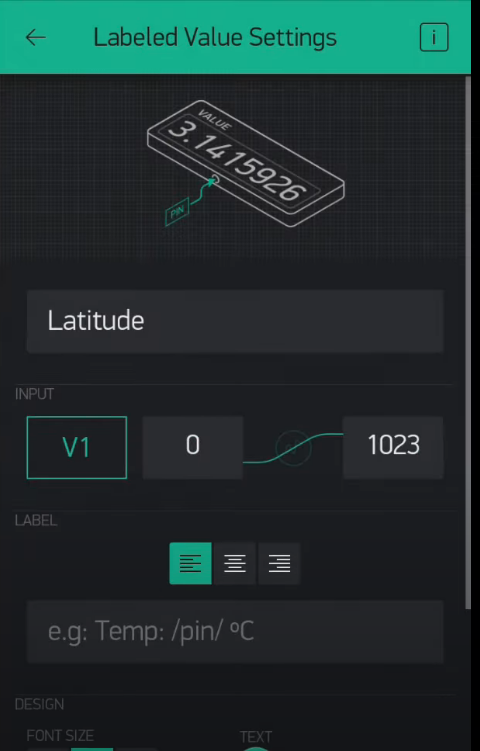


Lo siguiente después de crear el proyecto se arroja un token (código el cual necesitarás para conectar Blynk con IDE de Arduino) el cual se aloja en tu cuenta de Gmail.

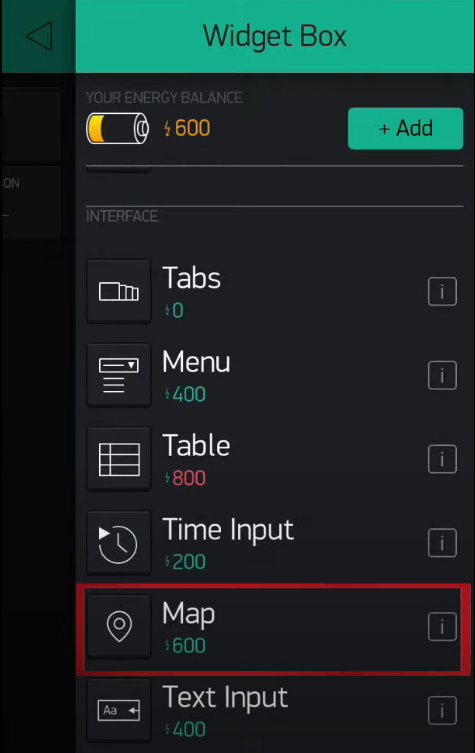
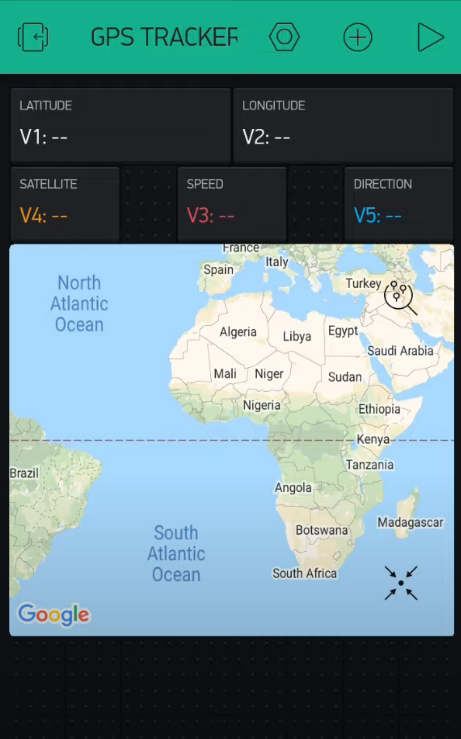
Ahora vamos a aplicar los widgets de información que son los siguientes



**Labeled Value vamos a usar 2 ya que van a ser la latitud y longitud**

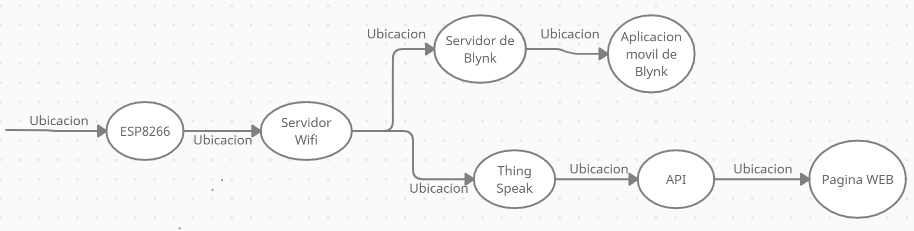


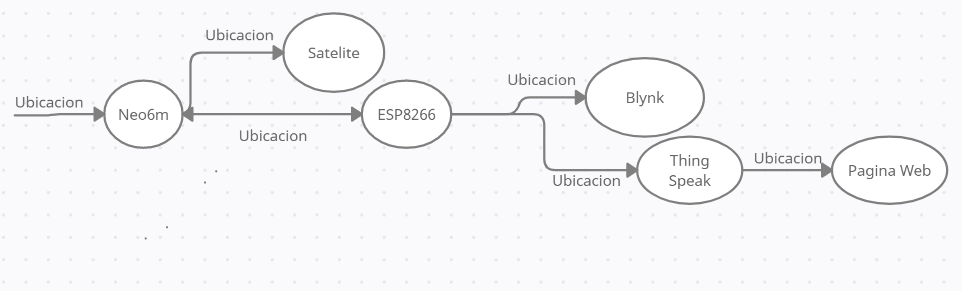
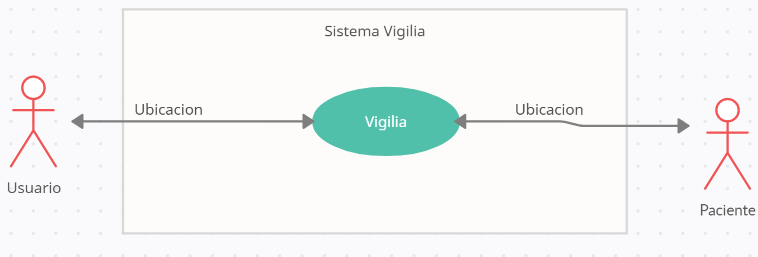
**Ahora lo más importante el mapa donde se verá reflejado el lugar donde estará la persona con el dispositivo en tiempo real**



Finalmente, para conectarla con el prototipo es necesario en las opciones de la izquierda copiar el token el cual se les enviará al Gmail. Para finalizar lo pegan en esta parte del código.

****

**6. Diagramas**



**7. Base de datos**

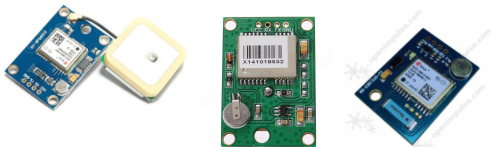
<https://thing-speak-1.xanstar.repl.co>

Mediante una programación en HTML, JavaScript y CSS realizamos una página web con el fin de que otros allegados fuera del que tenga la aplicación y quiere estar al tanto del asunto pueda mediante esta página ver la ubicación del enfermo.



Se tendrán que loguear y luego tendrán acceso a más opciones.

**8. Componentes**

1. Neo 6m 
2. ESP8266 ESP-12E module



**9. Secuencia del proyecto:**

El módulo GPS recibe datos de satélites; El módulo Wifi ESP8266 ESP-12 E inicia el servidor web y espera a que un cliente se conecte al servidor web. Una vez que el cliente está conectado al servidor web, el módulo Wifi ESP8266 ESP-12E envía los detalles de la ubicación al cliente conectado. Los detalles de la ubicación se muestran en una página web sencilla diseñada con HTML. Comprensión del módulo GPS NEO-6M GY-GPS6MV2 Comprensión del módulo WIFI ESP8266 ESP-12 E (LoLin NODEMCU V3)

.

**Señales y conexiones del módulo GPS NEO-6M GY-GPS6MV2**

El módulo GPS NEO6MV2 viene con 4 conexiones: RX, TX, VCC y GND, que es bastante fácil de incorporar con el uso de Software Serial en un Arduino Uno o una interfaz serial en un Arduino Mega. La fuente de alimentación del NEO 6 M debe ser de 3,6 V como máximo de acuerdo con la hoja de datos. Los tableros de conexiones típicos producidos en China contienen un regulador de voltaje para que entre 3-5 V VCC para que no dañe el tablero. Dado que los pines digitales también producen 5V, el divisor de voltaje se usa en el canal RX del receptor, ya que no está regulado.

RX (o RXD): pin de recepción. Conectado al pin TX de la placa Arduino.

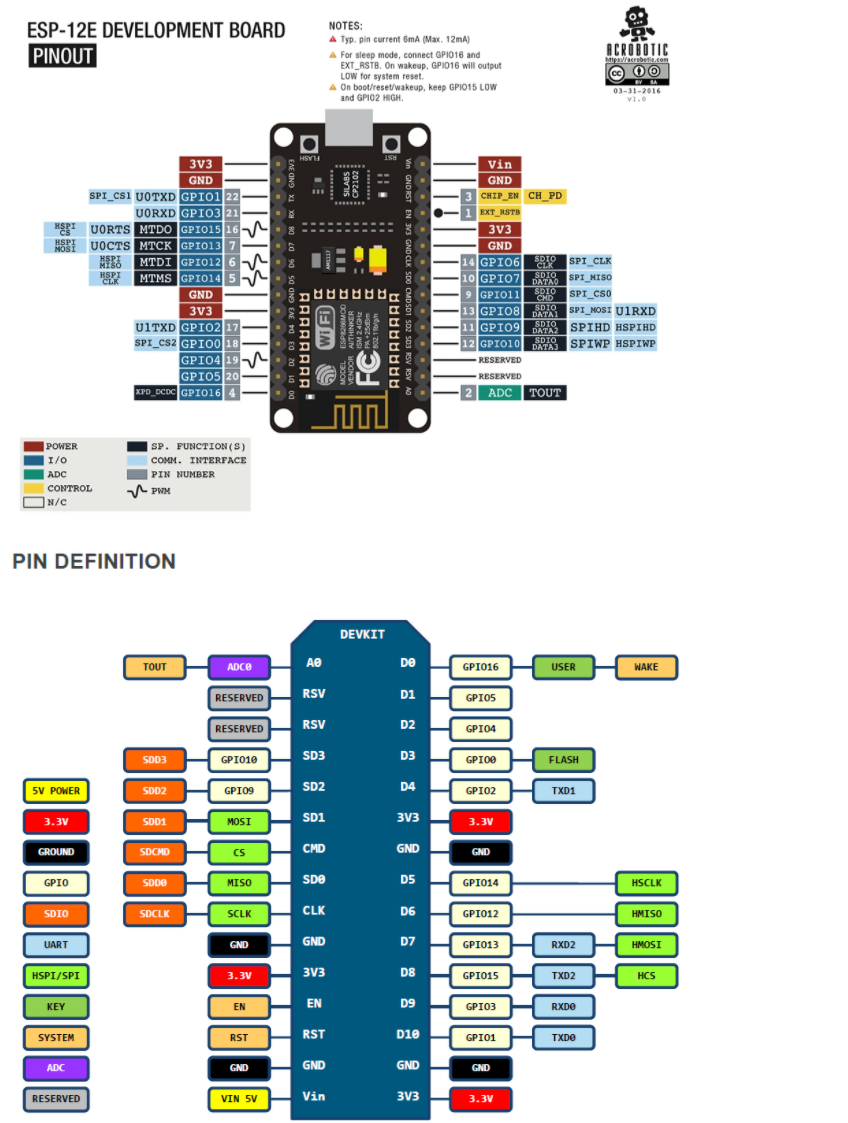
TX (o TXD) - pin de transmisión. Conectado al pin RX de la placa Arduino.

VCC - fuente de alimentación. Se puede conectar al pin + 5VDC o + 3.3VDC de la placa Arduino.

GND - tierra. Conectado al pin GND de la placa Arduino.

PPS: pulso por segundo. Este es un pin de salida en algunos módulos GPS. Generalmente, cuando este pin cambia, una vez por segundo, puede sincronizar el reloj de su sistema con el reloj del GPS.

Señales y conexiones del módulo ESP8266 ESP-12E WI FI (LoLin NODEMCU V3)



TX - pin de transmisión. Pin GPIO

RX - pin de recepción. Pin GPIO

3V3 (o 3V o 3.3V) - pin de fuente de alimentación (3-3.6V).

GND (o G) - pin de tierra.RST - pin de reinicio. Manténgalo en alto (3,3 V) para un funcionamiento normal. Ponlo en 0V para reiniciar el chip.

ES - Chip habilitado. Manténgalo en alto (3,3 V) para un funcionamiento normal.

Vin - Fuente de alimentación externa 5VDC.

D0-D8 - Pines GPIO (entrada y salida de propósito general)

D5-D8: interfaz SPI

Interfaz D1-D2– I²C / TWI

SC (o CMD) - (Selección de chip): el pin que el maestro puede usar para habilitar y deshabilitar dispositivos específicos. Pin GPIO

SO (o SDO) - Master In Slave Out (MISO) - Comunicación SP

I. La línea esclava para enviar datos al maestro. Pin GPIO

SK (o CLK) - SCK (Serial Clock) - Comunicación SPI. Pulsos de reloj que sincronizan la transmisión de datos generada por el maestro. Pin GPIO

S1 (o SD1) - Master Out / Slave In (MOSI). Comunicación SPI. La línea Master para enviar datos a los periféricos. Pin GPIO

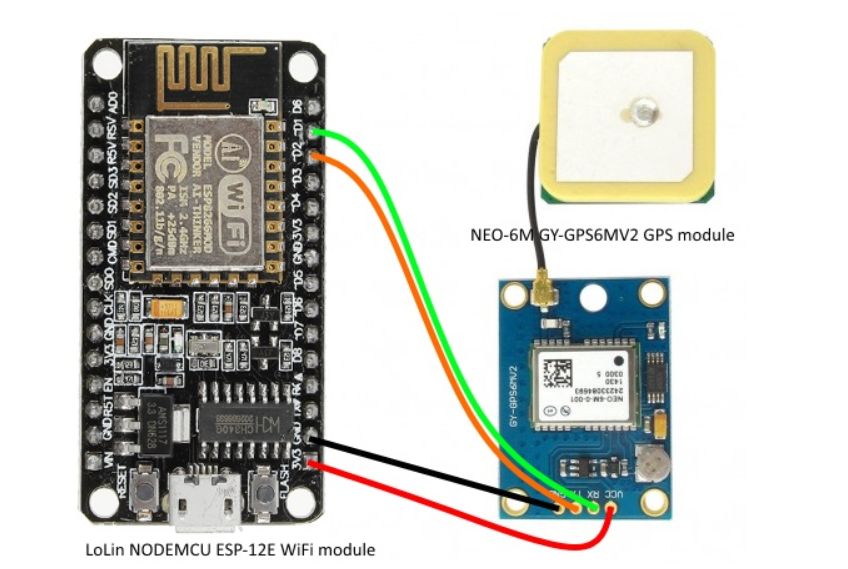
S2 (o SD2) - pin GPIO

S3 (o SD3) - pin GPIO

VU (o VUSB) - alimentación externa 5VDC.

A0 - Salida ADC.

RSV - reservado



**conexiones**

**3V3 VCC**

**GND GND**

**D1 (GPIO 5) RX**

**6D2 (GPIO4) TX**