Logotipo

Descripción generada automáticamente con confianza media

# TUTORIAL: GPS CON ESP32

Sergio Zubieta Mercado

Juan Diego Mamani Coarite

Ingeniería de Sistemas, Universidad privada del Valle

Ing. Rodmy Orellana Illanes

20 mayo 2024

[TUTORIAL: GPS CON ESP32 0](#_Toc167099840)

[1. MATERIALES 2](#_Toc167099841)

[2. SOFTWARE 2](#_Toc167099842)

[3. PROCEDIMIENTO GRAFICADO 2](#_Toc167099843)

[4. PROCEDIMIENTO 3](#_Toc167099844)

[I. Armar el circuito entre el ESP32 y GPS M8. 3](#_Toc167099845)

[II. Testeamos el gps para observar si manda las coordenadas 4](#_Toc167099846)

[III. Creamos la tabla dentro de la base de datos el cuál será una extensión de MySQL en Visual Studio Code 4](#_Toc167099847)

[IV. Generar un Token en Mapbox 5](#_Toc167099848)

[V. PHP #1: Recibe el HTTP post del Arduino code y conecta a la base de datos (anteriormente creada) 5](#_Toc167099849)

[VI. PHP #2: Sale un query SQL a la base de datos 6](#_Toc167099850)

[VII. HTML: Tiene el Token de mapbox y muestra el mapa 7](#_Toc167099851)

[VIII. Subir todo al 000webhost 7](#_Toc167099852)

[IX. HTTP POST 8](#_Toc167099853)

# MATERIALES

* ESP32: Lo necesitaras para mandar datos por wifi.
* GPS M8: Se conectará con un satélite.
* Protoboard: Para armar el circuito.
* Cables Jumper (macho macho): Realizar las conexiones.
* Computadora:
* Conexión a internet

# SOFTWARE

* Visual Studio Code: Para ver la base de datos
* 000WebHost: Servidor donde subiremos los códigos y veremos el MapBox
* 46: Contraseña y host en Arduino code

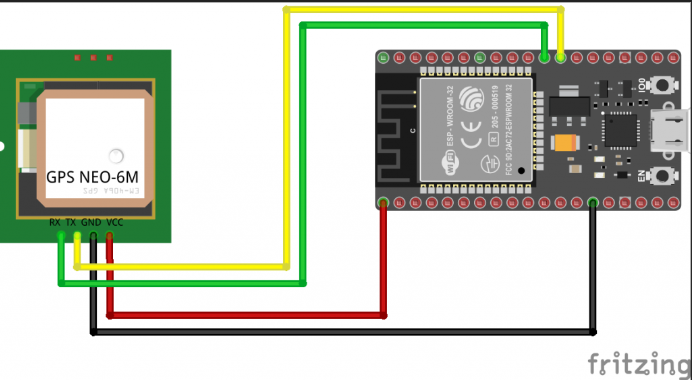
# PROCEDIMIENTO GRAFICADO

Diagrama

Descripción generada automáticamente

# PROCEDIMIENTO

## Armar el circuito entre el ESP32 y GPS M8.



Imágen 1: ESP32 + GPS M8

Imágen 2: ESP32 DATASHEET

Imagen que contiene circuito

Descripción generada automáticamente

## Testeamos el gps para observar si manda las coordenadas

Texto

Descripción generada automáticamente

## Creamos la tabla dentro de la base de datos el cuál será una extensión de MySQL en Visual Studio Code

Texto

Descripción generada automáticamente con confianza media

## Generar un Token en Mapbox

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

## PHP #1: Recibe el HTTP post del Arduino code y conecta a la base de datos (anteriormente creada)

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Verificación de parámetros GET:

Se verifica si los parámetros lat y lng están presentes en la solicitud GET.

Si están presentes, se asignan a las variables $latitude y $longitude.

Configuración de credenciales de la base de datos:

Define las variables para conectarse a la base de datos: servidor, nombre de usuario, contraseña y nombre de la base de datos.

Conexión a la base de datos:

Se crea una nueva instancia de mysqli para conectarse a la base de datos usando las credenciales definidas.

Verificación de la conexión:

Comprueba si la conexión fue exitosa. Si no lo fue, el script termina con un mensaje de error.

Consulta SQL para insertar las coordenadas:

Define una consulta SQL para insertar las coordenadas latitude y longitude en la tabla coordinates.

Ejecución de la consulta:

Ejecuta la consulta SQL. Si es exitosa, imprime un mensaje de éxito.

Si falla, imprime un mensaje de error incluyendo la consulta SQL y el error específico.

Cierre de la conexión:

Se cierra la conexión a la base de datos.

Mensaje de coordenadas no recibidas:

Si los parámetros lat y lng no están presentes en la solicitud GET, se imprime un mensaje indicando que no se recibieron coordenadas.

## PHP #2: Sale un query SQL a la base de datos

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Configuración de credenciales de la base de datos:

Define las variables para conectarse a la base de datos: servidor, nombre de usuario, contraseña y nombre de la base de datos.

Conexión a la base de datos:

Se crea una nueva instancia de mysqli para conectarse a la base de datos usando las credenciales definidas.

Verificación de la conexión:

Comprueba si la conexión fue exitosa. Si no lo fue, el script termina con un mensaje de error.

Consulta SQL para obtener las coordenadas:

Define una consulta SQL que selecciona las columnas latitude y longitude de la tabla coordinates.

Inicialización del array de coordenadas:

Se inicializa un array vacío para almacenar las coordenadas obtenidas.

Procesamiento de los resultados:

Si la consulta devuelve resultados, se recorren cada uno de ellos y se añaden al array coordinates como un array asociativo con las claves latitude y longitude.

Cierre de la conexión:

Se cierra la conexión a la base de datos.

Encabezado y salida JSON:

Se establece el encabezado de la respuesta HTTP a Content-Type: application/json para indicar que el contenido es JSON.

Se convierte el array de coordenadas a formato JSON y se imprime.

## HTML: Tiene el Token de mapbox y muestra el mapa

Captura de pantalla de computadora

Descripción generada automáticamente  
Texto

Descripción generada automáticamente

Inclusión de Mapbox GL JS:

Se incluyen los enlaces al script y la hoja de estilos de Mapbox GL JS en el <head> del documento HTML.

Estilos CSS:

Se define el estilo para que el cuerpo del documento y el contenedor del mapa no tengan margen ni relleno.

El contenedor del mapa (#map) ocupa todo el espacio disponible en la página.

Div contenedor del mapa:

Un elemento <div> con el ID map que servirá como contenedor del mapa.

Script de Mapbox GL JS:

Se define el token de acceso a Mapbox (mapboxgl.accessToken).

Se crea una nueva instancia de mapboxgl.Map con las opciones iniciales, como el contenedor del mapa, el estilo, la posición inicial (center) y el nivel de zoom (zoom).

Fetch de coordenadas y marcadores:

Se realiza una solicitud a fetch\_coordinates.php para obtener las coordenadas en formato JSON.

Cuando se recibe la respuesta, se procesa como JSON y se recorre el array de coordenadas.

Para cada coordenada, se crea un nuevo marcador (new mapboxgl.Marker()) y se posiciona en el mapa usando setLngLat con las coordenadas obtenidas. Luego, se agrega el marcador al mapa con addTo(map).

## Subir todo al 000webhost

Interfaz de usuario gráfica, Aplicación, Sitio web

Descripción generada automáticamente

## HTTP POST y conexión a WiFi del ESP32

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Texto

Descripción generada automáticamente

* No olvides cambiar las líneas 7 y 8 con tus datos de conexión hacia tu router.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente