

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

SISTEMA PROTOTIPO PARA EL MONITOREO INALÁMBRICO DE LA CANTIDAD DE DESECHOS DE UN CONTENEDOR DE BASURA PARA CIUDADES INTELIGENTES

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO
DE INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES**

EDISSON ALONSO ACOSTA CURIPALLO

edisson.acosta@epn.edu.ec

DIRECTOR: ING. RICARDO XAVIER LLUGSI CAÑAR, MSc.

ricardo.llugsi@epn.edu.ec

Quito, Enero 2020

AVAL

Certifico que el presente trabajo fue desarrollado por Edison Alonso Acosta Curipallo, bajo mi supervisión.

ING. RICARDO XAVIER LLUGSI CAÑAR, MSc.
DIRECTOR DEL TRABAJO DE TITULACIÓN

DECLARACIÓN DE AUTORÍA

Yo Edison Alonso Acosta Curipallo, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

EDISSON ALONSO ACOSTA CURIPALLO

DEDICATORIA

A mis padres y hermanos por el amor brindado, apoyo incondicional, consejos y paciencia durante toda la carrera. Este logro que alcanzo es por y para ellos

Edisson A. Acosta C.

AGRADECIMIENTO

A Héctor mi padre quién es el pilar de mi vida por todo el cariño, apoyo y esfuerzo que hizo para brindarme todo lo necesario en cada paso que he dado para culminar esta etapa de mi vida.

A Carmen mi madre quién es el motor de mi vida, sin ella nada sería posible, gracias por ser la luz que guía mi camino.

A mis hermanos Carolina y Mauricio, quienes son mis mejores amigos, gracias por su cariño infinito y apoyo incondicional.

A mi grupo de amigos, que de una u otra manera se hicieron presentes y ayudaron a superar cualquier problema e hicieron más grato el curso de la universidad.

Al Msc. Ricardo Llugsí, por su ayuda, paciencia y consejos, gracias por confiar en mí para la elaboración de este proyecto.

Edisson A. Acosta C.

ÍNDICE DE CONTENIDO

| | |
|--|------|
| AVAL..... | i |
| DECLARACIÓN DE AUTORÍA | ii |
| DEDICATORIA | iii |
| AGRADECIMIENTO | iv |
| ÍNDICE DE CONTENIDO..... | V |
| RESUMEN | VIII |
| ABSTRACT..... | IX |
| 1. INTRODUCCIÓN..... | 1 |
| 1.1 Objetivos | 1 |
| 1.2 Alcance | 2 |
| 1.3 Marco Teórico | 3 |
| 1.3.1 Residuos sólidos | 3 |
| 1.3.1.1 Impacto Ambiental. | 4 |
| 1.3.1.2 Impacto en la salud..... | 5 |
| 1.3.1.3 Contenedores | 5 |
| 1.3.2 Comunicaciones Inalámbricas | 6 |
| 1.3.3.1 El espectro radioeléctrico | 6 |
| 1.3.3.2 Sistemas Celulares | 7 |
| 1.3.2.2.1 Sistemas de primera generación | 8 |
| 1.3.2.2.2 Sistemas de segunda generación | 8 |
| 1.3.2.2.3 Sistemas de generación Intermedia | 9 |
| 1.3.2.2.4 Sistemas de tercera generación | 9 |
| 1.3.2.2.5 Sistemas de cuarta generación | 10 |
| 1.3.3 Sistema autónomo de energía fotovoltaica | 11 |
| 1.3.3.1 Módulos fotovoltaicos | 12 |
| 1.3.3.2 Baterías..... | 12 |
| 1.3.3.3 Controlador de carga solar | 12 |
| 1.3.4 Módulos GSM/GPRS..... | 12 |
| 1.3.4.1 SIM 800L..... | 12 |
| 1.3.4.2 SIM 900..... | 13 |
| 1.3.4.3 SIM 908..... | 13 |
| 1.3.4.4 Modem USB 3G..... | 14 |
| 1.3.5 Módulo GPS | 14 |

| | | |
|-----------|--|----|
| 1.3.5.1 | Ublox NEO 7M | 14 |
| 1.3.5.2 | Ublox NEO 6M | 15 |
| 1.3.6 | Sensores de distancia | 15 |
| 1.3.6.1 | VL53L0X | 15 |
| 1.3.6.2 | SC-HR04..... | 16 |
| 1.3.6.3 | JSN-SR04T | 17 |
| 1.3.7 | Tarjetas de procesamiento..... | 17 |
| 1.3.7.1 | Arduino..... | 17 |
| 1.3.7.1.1 | Arduino Uno | 18 |
| 1.3.7.1.2 | Arduino Mega..... | 18 |
| 1.3.7.2 | Raspberry Pi | 19 |
| 1.3.7.2.1 | Raspberry Pi Zero W v1.3 | 20 |
| 1.3.7.2.2 | Raspberry Pi 3 Modelo B+ | 20 |
| 1.3.8 | Conceptos para la creación de bases de datos en LAMP | 21 |
| 1.3.8.1 | MySQL | 21 |
| 1.3.8.2 | Apache | 21 |
| 1.3.8.3 | Phpmyadmin | 21 |
| 1.3.8.4 | HyperText Markup Language (HTML)..... | 21 |
| 1.3.8.5 | Hypertext Preprocessor (PHP) | 22 |
| 1.3.8.6 | Python | 22 |
| 1.3.9 | Conceptos de almacenamiento en la Nube | 22 |
| 2. | DESARROLLO DEL SISTEMA..... | 24 |
| 2.1. | Selección de elementos | 24 |
| 2.1.1. | Selección de la Tarjeta de Procesamiento | 24 |
| 2.1.2. | Selección del sensor de distancia..... | 25 |
| 2.1.3. | Selección de módulo GPS | 26 |
| 2.1.4. | Selección del módulo GSM/GPRS..... | 26 |
| 2.2. | Dimensionamiento del panel solar y batería..... | 27 |
| 2.2.1. | Consumo de potencia..... | 27 |
| 2.2.2. | Elección del panel solar | 28 |
| 2.2.3. | Elección de la batería | 29 |
| 2.2.4 | Elección del controlador de carga solar..... | 30 |
| 2.3. | Configuración del sistema | 30 |
| 2.3.1. | Raspbian..... | 30 |
| 2.3.1.1. | Conexión Remota por Windows | 35 |
| 2.3.1.2. | Conexión Remota por VNC | 37 |

| | | |
|----------|---|--------------------------------------|
| 2.3.2. | Configuración del sensor de Distancia | 39 |
| 2.3.2.1. | Prueba de funcionamiento del sensor de distancia..... | 40 |
| 2.3.3. | Configuración del módulo GPS..... | 42 |
| 2.3.3.1. | Prueba de funcionamiento del módulo GPS | 43 |
| 2.3.4. | Configuración del módulo GSM/GPRS..... | 46 |
| 2.3.4.1. | Prueba de funcionamiento del Modem 3G | 49 |
| 2.3.5. | Configuración de los servidores de la base de datos..... | 50 |
| 2.3.5.1. | Creación de un servidor web en la nube..... | 51 |
| 2.3.5.2. | Instalación del servidor Apache, MySQL y PHPmyAdmin | 55 |
| 2.3.5.3. | Creación de la base de datos | 58 |
| 2.3.5.4. | Prueba de conexión de la Raspberry a la base de datos | 60 |
| 2.3.6. | Creación del Sitio WEB | 61 |
| 2.3.7. | Conexión del Sistema Fotovoltaico..... | 68 |
| 2.3.8. | Conexión final..... | 69 |
| 2.3.9. | Configuraciones adicionales | 70 |
| 2.3.9.1. | Programa on/off..... | 70 |
| 2.3.9.2. | Envío de correos electrónicos..... | 71 |
| 2.3.9.3. | Automatización..... | 72 |
| 3. | PRUEBAS Y RESULTADOS | 73 |
| 3.1. | Etapa de construcción..... | 73 |
| 3.2. | Funcionamiento general..... | 75 |
| 3.3. | Visualización web..... | 79 |
| 3.4. | Presupuesto Referencial..... | 81 |
| 4. | CONCLUSIONES Y RECOMENDACIONES | 83 |
| 4.2. | Recomendaciones | 84 |
| 5. | REFERENCIAS BIBLIOGRÁFICAS..... | 87 |
| 6. | ANEXOS | ¡Error! Marcador no definido. |
| | ANEXO I. SCRIPT PROGRAMA DE DISTANCIA ... | ¡Error! Marcador no definido. |
| | ANEXO II. SCRIPT PROGRAMA DE UBICACION . | ¡Error! Marcador no definido. |
| | ANEXO III. SCRIPT PROGRAMA PRINCIPAL | ¡Error! Marcador no definido. |

RESUMEN

La contaminación ambiental, pérdidas de dinero, problemas de salubridad y problemas estéticos de una ciudad debido a la aglomeración y desbordamiento de basura de los contenedores impulsan a buscar soluciones inteligentes y de bajo costo.

Este proyecto busca adquirir y transmitir inalámbricamente datos de los niveles de desechos de un contenedor hacia un servidor y su posterior visualización en una página web.

Las pruebas se realizarán en un ambiente controlado, y la cantidad de desechos se medirán en tres niveles: vacío, medio y lleno, mediante un sensor ultrasónico. Los datos de la cantidad de desechos, así como su geolocalización serán procesados por una tarjeta de procesamiento, la cual enviará los datos adquiridos automáticamente a través de la red celular cada cierto tiempo.

El prototipo contará con una alimentación autónoma conformada por un panel solar, batería y controlador de carga solar. Se buscará proteger el prototipo contra elementos que sean dañinos para su vida útil como las altas temperaturas, agua, polvo, sustancias químicas, entre otras.

PALABRAS CLAVE: Comunicaciones Inalámbricas, Hardware Libre, Base de Datos, Sensores.

ABSTRACT

Environmental pollution, loss of money, health problems and aesthetic problems of a city due to the agglomeration and overflow of garbage from the containers impel to look for intelligent and low-cost solutions.

This project seeks to wirelessly acquire and transmit data of the waste levels of a container to a server and its subsequent visualization on a web page.

The tests will be carried out in a controlled environment and the amount of waste will be measured in three levels: empty medium and full using an ultrasonic sensor. The waste quantity data as well as its geolocation will be processed by a processing card which will send the data acquired automatically through the cellular network during a period.

The prototype will have an autonomous power supply consisting of a solar panel, battery and solar charge controller. It will seek to protect the prototype against elements that are harmful to its useful life such as high temperatures, water, dust, chemicals, among others.

KEYWORDS: Wireless Communications, Hardware, Database, Sensors.

1. INTRODUCCIÓN

Una buena estética y ornato de una ciudad, reducirá problemas ambientales y de salud para la sociedad, generará ganancias al atraer turismo a una ciudad limpia, reducirá costos de recolección y transporte.

La importancia de monitorear la cantidad de desechos en un contenedor permitirá realizar una mejor planificación de rutas para la recolección de los desechos y con ello un mejor manejo de la basura dentro de la ciudad trayendo beneficios ambientales, salubres y económicos.

El prototipo debe medir la cantidad de desechos al interior de un contenedor de basura, obteniendo información cuando los desechos superen cada uno de los niveles: vacío, medio y lleno; para emitir una señal de alerta cuando los desechos se encuentren en el nivel lleno. El proyecto estará alimentado automáticamente por paneles solares y baterías.

El prototipo está expuesto a posibles daños por las condiciones ambientales en las que se encuentra por la acción corrosiva del medio ambiente (temperatura, humedad, polvo, contaminación), por lo que se buscará proteger de manera óptima los componentes electrónicos, para evitar una grave degradación o fallo total del sistema.

1.1 Objetivos

El objetivo general de este Proyecto es: Monitorear inalámbricamente el nivel de los desechos de un contenedor de basura típico usado en la ciudad de Quito.

Los objetivos específicos de este Proyecto son:

- a) Detallar las tecnologías para transmisión inalámbrica de datos.
- b) Desarrollar el prototipo para la adquisición y transmisión de los datos geolocalizados de los niveles de desechos de un contenedor hacia una base de datos en un servidor y a una página web.
- c) Implementar un sistema fotovoltaico autónomo para la alimentación del prototipo.
- d) Utilizar un mecanismo de protección óptimo contra todas las posibles condiciones dañinas para los elementos utilizados con el fin de asegurar la vida útil y correcto funcionamiento del prototipo.

e) Probar el funcionamiento correcto del prototipo en una maqueta de dimensiones similares a los contenedores de basura utilizados en la ciudad de Quito ubicándolo en diferentes lugares.

1.2 Alcance

Con el presente proyecto se medirá la cantidad de desechos en el interior de un contenedor de basura en tres niveles (vacío, medio y lleno) con sensores digitales ultrasónicos.

Se usará un sensor ultrasónico para la adquisición de la cantidad de desechos, al ser digitales se obtendrá una buena precisión en cuanto a la distancia a los residuos dentro del contenedor. Estos sensores disponen de un rango de operación de 400 cm, consume un voltaje de 5 V con un ángulo de medida de 30°, precisión de 3 mm y funciona a una temperatura de -10°C a 70°C [1]. Estos sensores son muy confiables cuentan con una buena precisión, son de bajo costo y fáciles de implementar; puede ser resistente al agua.

Los datos de cantidad desechos recolectados por el sensor serán procesados por una tarjeta de procesamiento, la cual también se encargará de enviarlas a través de la red celular hacia un servidor para poder ser almacenada y posteriormente mostrada por una página web. La ubicación del contenedor se enviará solamente cuando exista una variación a la posición original, mientras que el envío de los datos de la cantidad de desechos se enviará cada que los desechos superen un nivel de cantidad (vacío, medio, lleno), una vez que la cantidad de desechos haya superado el nivel de lleno del contenedor de basura los datos se enviarán de manera periódica cada cierto tiempo.

El prototipo se alimentará mediante un sistema fotovoltaico con paneles solares y baterías que almacenen la energía. Este sistema debe contar con un adecuado mantenimiento cada cierto tiempo para asegurar su correcto funcionamiento.

Para la transmisión inalámbrica de datos la tarjeta de procesamiento se conectará a la red celular a través de un módulo GSM/GPRS que al estar ubicado en la zona urbana contará de una buena cobertura y se podría aumentar la tecnología.

La información será almacenada en una base de datos de un servidor para su posterior visualización a través de una página web, donde se mostrarán cuadros de valores de los niveles de desechos de un contenedor con su respectiva geolocalización, estampas de fecha y hora de una forma agradable al usuario.

Se buscará la protección del prototipo para que no sufra daños por el ambiente en el cual va a ser utilizado (temperaturas altas por el sol, agua, polvo, sustancias químicas, etc.).

Una parte del prototipo se encontrará al interior del contenedor (caja de procesamiento: sensores y circuito de comunicación) y otra parte fuera (paneles solares, baterías), se deberá aislar el ingreso de agua a la caja de procesamiento, para evitar la corrosión en los elementos electrónicos que serán utilizados.

Las pruebas del prototipo se realizarán en un ambiente controlado en una maqueta con tamaño y forma similar a un contenedor de basura típico de la ciudad de Quito, con el fin de constatar el buen funcionamiento de los elementos electrónicos, la adquisición y transmisión de datos a través de la red celular hacia un servidor y a una página web. Se verificará que no ingrese agua a la caja de procesamiento.

Finalmente, se busca que el proyecto sea de bajo costo, sin que el prototipo sea deficiente, que cuente con elementos que cumplan con las características técnicas adecuadas para brindar un servicio óptimo, y con los requerimientos previstos del proyecto. Asegurar su vida útil, con la protección y mantenimiento adecuado.

Aunque se probará en un ambiente controlado el funcionamiento del prototipo, se podrá implementar en los distintos contenedores de basura de las ciudades con el fin de ayudar con la planificación de recolección de desechos y con ello mejorar el ornato.

1.3 Marco Teórico

1.3.1 Residuos sólidos

El constante aumento poblacional en zonas urbanas implica un mayor consumo y con ello generación de residuos sólidos, es decir, cualquier material que ya ha cumplido con su vida útil y sea desechado por las personas se torna inútil y sin valor económico [2].

La mala gestión integral de este tipo de desechos produce impactos de contaminación ambiental para cualquier ciudad convirtiéndose en un serio problema, ya que los residuos sólidos causan problemas como:

- Deterioro en la salud pública.
- Pérdidas de económicas.
- Problemas estéticos de la ciudad debido a la aglomeración y desbordamiento de basura de los contenedores.
- Contaminación de aguas superficiales y subterráneas.
- Atracción de roedores e insectos los cuales son portadores de enfermedades [3].

1.3.1.1 Impacto Ambiental.

El impacto ambiental por residuos sólidos se produce debido a la acción directa del hombre en la generación de desechos, lo cual al no contar con un manejo correcto para la recolección de basura se convierte en un problema serio a nivel mundial que trae consigo diversos problemas para la sociedad.

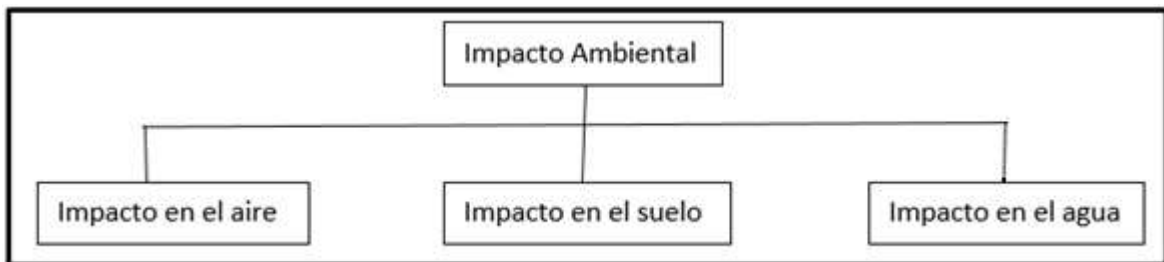


Figura 1.1 Impactos Ambientales

La contaminación ambiental debido a los residuos sólidos tiene impactos de diferentes maneras.

El impacto en el aire se debe a la expulsión de malos olores por parte de la basura en estado de descomposición y de elementos líquidos, esta contaminación es un resultado del desbordamiento de basura de los contenedores, trae consigo enfermedades respiratorias para las personas que absorben estos contaminantes y que se expanden de los pulmones hacia otras partes del cuerpo. Entre los gases presentes en el aire contaminado están:

- Dióxido de carbono.
- Óxido nitroso.
- Gas sulfhídrico que produce un olor a huevo podrido, causa irritaciones y puede llegar a provocar intoxicaciones.
- Gas metano, es el más peligroso ya que es el responsable de producir efecto invernadero y es explosivo.

El impacto en el agua se debe a la aglomeración de basura o quema de desechos sobre este líquido (agua lluvia, por ejemplo) que produce una contaminación por medio de lixiviados (sustancias tóxicas procedentes de la basura descompuesta), el cual es un líquido altamente contaminado compuesto principalmente de ácidos orgánicos los cuales pueden solubilizar metales. Puede haber contaminación de aguas subterráneas o en aguas

superficiales que afectan a todos los ecosistemas, ya sea que lleguen a ríos y afecten a peces y otros animales que pueden llegar a beber el agua contaminada.

El impacto en el suelo es provocado por el desbordamiento de basura de los contenedores, el arrojo de desechos en sitios no adecuados por lo que se convierte en un problema público y una molestia desagradable para la ciudad. Este tipo de contaminación trae consigo problemas turísticos ya que todos quieren vivir y visitar lugares frescos, limpios, saludables y no en una ciudad maloliente con saneamiento deficiente y basura por todas partes, además que la acumulación de basura atrae a insectos, roedores y perros callejeros los cuales portan enfermedades para el ser humano [4].

1.3.1.2 Impacto en la salud

Los impactos mencionados anteriormente afectan la salud del ser humano ya sea de manera directa (en contacto con los desechos) o de manera indirecta (por medio de animales, suelo, aire o agua contaminada). Este problema puede reducirse con el correcto manejo y la recolección a tiempo de los desechos para evitar la aglomeración en los alrededores de los contenedores.

Los residuos sólidos pueden causar más de 40 enfermedades causadas por la contaminación ambiental entre las cuales están: irritación de ojos, infecciones intestinales transmitidas por las moscas que se alimentan de los desechos, afecciones de la función de los pulmones al reducir la acción de los cilios disminuyendo así la limpieza normal de los pulmones provocado al respirar continuamente el aire contaminado por la descomposición de los residuos sólidos, infecciones en la piel y la sangre a través de heridas infectadas que se encuentren en contacto directo con los desechos, diversas enfermedades causadas por las mordeduras de animales que se alimentan de los desechos. Al existir aglomeración de basura las personas que se encargan de recoger los desechos están expuestos a desechos peligrosos como objetos afilados, además que este tipo de contaminación puede provocar cáncer debido a compuestos orgánicos volátiles y dioxinas [5][6][7].

1.3.1.3 Contenedores

El uso de los contenedores está diseñado para otorgar una mejor imagen de orden y limpieza en el espacio público. Este recipiente debe ser utilizado de manera correcta por la ciudadanía para evitar posibles daños al prototipo que será instalado, así como para los vehículos de recolección.

Se debe colocar los desechos en fundas grandes, bien cerradas dentro del contenedor y no en los alrededores, de esta manera se reducirá el impacto ambiental. Es obligación colocar únicamente residuos sólidos domésticos, no se deberán colocar escombros de construcción, llantas, electrodomésticos, césped o muebles [8].

Las dimensiones y forma del contenedor típico utilizado en la ciudad de Quito se presentan a continuación en la Figura 1.2.

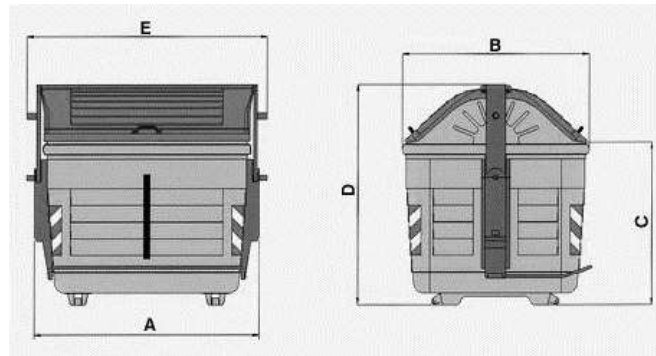


Figura 1.2 Vistas del contenedor de basura [51].

Tabla 1.1 Dimensiones del contenedor de basura.

| Medida | A | B | C | D | E |
|--------|-----|-----|-----|-----|-----|
| Cm | 160 | 130 | 113 | 158 | 162 |

1.3.2 Comunicaciones Inalámbricas

La tecnología para transmitir inalámbricamente los datos tiene casi un siglo de existencia y ha evolucionado aceleradamente en los últimos años, perfeccionando técnicas de transmisión que brinden confiabilidad, seguridad y velocidad de transmisión.

Para establecer la transmisión de datos de un punto a otro es necesario un equipo emisor, un mensaje, el canal o medio de transmisión (medios guiados y no guiados) por el cual se envían los datos por señales ópticas, eléctricas o electromagnéticas y el equipo receptor [9].

1.3.3.1 El espectro radioeléctrico

Es un subconjunto del espectro electromagnético que contiene el rango de frecuencias hábiles para las comunicaciones. Es un recurso escaso, ya que las radiofrecuencias son asignadas por un ente regulatorio (ARCOTEL en Ecuador) con el fin que se haga el mejor uso de ellas, además debido a las interferencias y a otros factores el rango de frecuencias empleadas para uso civil y militar va de los 9 KHz a los 50 GHz, y hasta 400 GHz para uso

experimental. Los sistemas de telefonía inalámbrica trabajan en la banda alta UHF (*Ultra High Frequency*, Frecuencia Ultra Alta) que va de 1 a 2 GHz para el uso de los sistemas de comunicaciones móviles celulares, inalámbricas como son UMTS (*Universal Mobile Telecommunications System*, Sistema Universal de Telecomunicaciones Móviles) y LTE (*Long Term Evolution*, Evolución a Largo Plazo) [10].

Esta asignación de frecuencias puede variar de unas regiones geográficas a otras, ya que el mundo se ha dividido en 3 (“Región 1 para Europa y África, Región 2 para América, Región 3 para Asia y Oceanía”) [10].

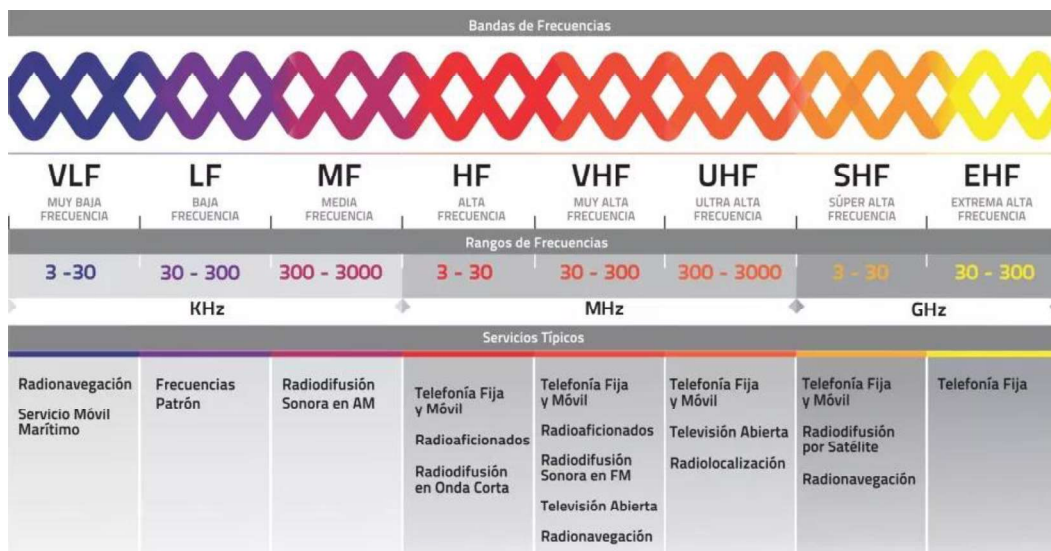


Figura 1.3 Bandas de frecuencias y aplicaciones [10].

1.3.3.2 Sistemas Celulares

La telefonía móvil moderna empezó en la década de los 80 con sistemas analógicos como AMPS (*Advanced Mobile Phone System*, Sistema Telefónico Móvil Avanzado) y que son conocidos como primera generación (1G), en la década de los 90 comienza la tecnología digital (2G, 3G y 4G), siendo GSM (*Global System for Mobile Communications*, Sistema Global para las Comunicaciones Móviles) el estándar representante de la segunda generación (2G), este estándar ha ido evolucionando a nuevos estándares para la tercera generación (3G) como lo es UMTS, y de cuarta generación (4G) como lo es la tecnología LTE, los sistemas digitales soportan comunicaciones de datos, como son las redes locales Wi-Fi (*Wireless Fidelity*, Fidelidad Inalámbrica), los sistemas WiMAX (*Worldwide Interoperability for Microwave Access*, Interoperabilidad Mundial para Acceso por Microondas), o la propia tecnología Bluetooth para comunicaciones a muy corta distancia [10].

1.3.2.2.1 Sistemas de primera generación

Estos sistemas utilizan tecnología analógica, estrictamente dedicados al servicio de voz que inició en 1979. Este sistema se basa en el uso de protocolos de traspaso de celdas entre varias estaciones base cercanas, con velocidades de 4800 bps. Algunos de los estándares son:

AMPS era la tecnología dominante de la primera generación, operaba en la banda de frecuencia de 800 MHz.

ETACS (*Extended Total Access Communications System*, Sistema de Comunicaciones de Acceso Total Extendido), operaba en Europa, en la banda de frecuencia de 900 MHz.

NMT (*Nordic Mobile Telephone*, Telefonía Móvil Nórdica), operaba en la banda de 900 MHz en los países escandinavos [9].

1.3.2.2.2 Sistemas de segunda generación

GSM es un sistema de tecnología digital que opera en las bandas de frecuencia de: 850, 900, 1800 y 1900 MHz, alcanzan una velocidad de 14.4 Kbps. Utiliza una técnica denominada de acceso múltiple, como son: "TDMA (*Time Division Multiple Access*, Acceso Múltiple por División de Tiempo) donde el sistema divide al canal de transmisión en ocho ranuras de tiempo y CDMA (*Code Division Multiple Access*, Acceso Múltiple por División de Código)" [10].

GSM provee mejoras a los sistemas analógicos como: mejoras en la calidad de voz, envío de datos a baja velocidad, facilita la formación de redes virtuales privadas, permite la identificación de un cliente suscrito, envío de SMS (*Short Message Service*, Servicio de Mensajes Cortos) con una longitud de 160 caracteres alfanuméricos, identificación, restricción y retención de llamadas, buzón de voz, multiconferencia, *roaming*.

En GSM los recursos son gestionados por la modalidad *resource reservation*, es decir, que los recursos están ocupados incluso el momento en que la petición de servicio no se ha establecido [10].

Características

- Rango de frecuencias de recepción: 925 a 960 MHz.
- Rango de frecuencias de emisión: 860 a 915 MHz.
- 8 canales por portadora en full dúplex y 16 half dúplex, de los cuales un canal es de control.

- El número total de portadoras es 124.
- Cada portadora tiene una separación de 200 KHz.
- El ancho de banda del canal de radio es 25 KHz.
- Capacidad de tráfico es 200 Erlangs por cada Km².
- Utiliza técnicas de transmisión como TDMA y FDD.
- Usa modulación GMSK y voz codificada a 13 Kbps.

1.3.2.2.3 Sistemas de generación Intermedia

GPRS (*General Packet Radio Service*, Servicio General de Conmutación de Paquetes sobre Radio), es una tecnología que junto con EDGE (*Enhanced Data Rates for GSM*. Tasas de Datos Mejoradas para la evolución de GSM) e IS-95 son conocidos como generación intermedia (2.5G). Permite enviar y recibir información aprovechando la subestructura de la red GSM para coexistir ambas y alcanzan mayor velocidad de datos máxima de 171.2 Kbps, aunque en la práctica es de 40 o 60 Kbps [9].

“GPRS introduce una interfaz para el cambio de paquetes y utiliza la técnica *del context reservation*, es decir, los recursos de radio se ocupan, solo si hay necesidad de transmitir o recibir datos. Los propios recursos de radio de una celda se agrupan esta manera entre todas las estaciones móviles (MS), de esta manera aumenta la eficacia del sistema” [10].

El sistema GPRS permite el acceso a la red de paquetes hacia los usuarios del dispositivo móvil por medio de protocolos TCP/IP (*Transmission Control Protocol/Internet Protocol*, Protocolo de Control de Transmisión/Protocolo de Internet), que omite la utilización de conexiones intermedias por conmutación del circuito.

GPRS ofrece un servicio direccionado para aplicaciones con las siguientes características:

- “Transmisión poco frecuente de pequeñas o grandes cantidades de datos (por ejemplo, aplicaciones interactivas).
- Transmisión intermitente de tráfico de datos, por ejemplo, aplicaciones las que el tiempo medio entre dos transacciones consecutivas es de duración superior a la duración media de una única transacción” [10].

1.3.2.2.4 Sistemas de tercera generación

UMTS es un sistema de tercera generación (3G), miembro de la familia global IMT-2000 (*International Mobile Telecommunication 2000*, Telecomunicaciones Móviles Internacionales 2000) de la UIT (*International Telecommunication Union*, Unión Internacional

de Telecomunicaciones), está estandarizado por el ETSI (*European Telecommunications Standards Institute*, Instituto Europeo de Estándares de Telecomunicación) junto con organismos como: TIA (*Telecommunications Industry Association*, Asociación de Industrias de Telecomunicación) en EE.UU. y la ARIB (*The Association of Radio Industries and Businesses*, Asociación de las Empresas de Radiodifusión) en Japón.

Esta red es conocida como WCDMA (*Wideband Code Division Multiple Access*, Acceso Múltiple por División de Código de Banda Ancha), representa una evolución de la red móvil GSM para servicios de datos inalámbricos de alta velocidad de datos, mayores capacidades de servicio. Utiliza una tecnología en base al protocolo IP la cual brinda soporte de voz y datos en paquetes, así con acceso rápido a internet, puede llegar a tener velocidades de datos hasta 2 Mbps, pero en la práctica se reducen al menos a 384 Kbps [10].

La tecnología UMTS evolucionó a HSDPA (*High Speed Downlink Packet Access*, Acceso a Descarga de Paquetes a Alta Velocidad) obteniendo una velocidad de 14.4 Mbps y HSUPA (*High Speed Uplink Packet Access*, Acceso Ascendente de Paquetes a Alta Velocidad) con una velocidad de 5.6 Mbps [9].

La tercera generación permite conectarse de manera continua a internet inalámbricamente con la convergencia de voz y datos, es decir, ofrece aplicaciones multimedia (por ejemplo, comunicaciones de datos con imagen, audio y video) y altas transmisiones de datos [9]

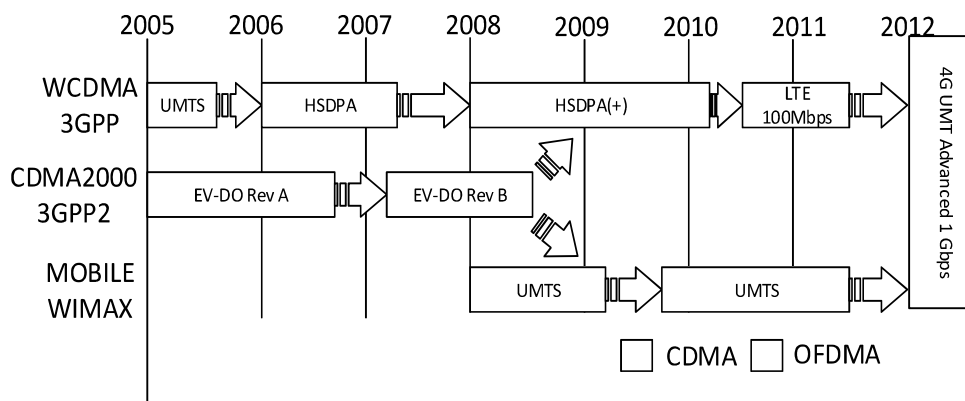


Figura 1.4 Evolución de la banda ancha móvil [9].

1.3.2.2.5 Sistemas de cuarta generación

UMTS ha evolucionado hacia LTE, es un sistema que puede brindar velocidades de 100 Mbps e incluso 1 Gbps en reposo. De la misma forma que UMTS, la tecnología LTE, posee gran eficiencia espectral y baja latencia.

LTE permite comunicaciones simultáneas de voz y datos incluso cuando existen altas densidades de usuarios, tiene soporte para las aplicaciones multimedia que requieren mayor capacidad de canal.

LTE presenta una infraestructura de la red que se basa en el protocolo IP por lo que es considerada como una red de redes, simplemente es una red completamente nueva, soporta tanto IPv4 como IPv6.

La evolución hacia LTE-A es considerada de cuarta generación (4G), es un sistema que brinda mayores velocidades de datos de bajada de hasta 3 Gbps y 1.5 Gbps de subida, aunque en la práctica se consiguen velocidades de 1.2 Gbps y 568 Mbps respectivamente, tiene mejor QoS (*Quality of Service*, Calidad de Servicio) y alta seguridad extremo a extremo; mayor eficiencia espectral, capacidad para suscriptores simultáneos con interoperabilidad entre sí mediante la tecnología MIMO (*Multiple Input Multiple Output*, Múltiple Entrada Múltiple Salida) [9].

1.3.3 Sistema autónomo de energía fotovoltaica

Es un sistema que se encuentra aislado de la red eléctrica, producen energía eléctrica únicamente cuando están expuestas a la luz del sol, por tanto, su fuente de energía es directamente proporcional a la radiación solar que llega al módulo fotovoltaico.

En general un sistema fotovoltaico autónomo está compuesto por paneles solares, baterías, reguladores de voltajes, inversores de voltajes y el consumidor.

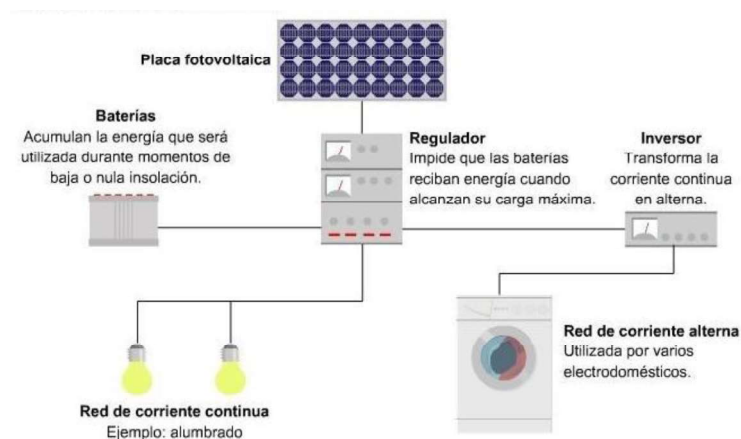


Figura 1.5 Sistema fotovoltaico [12].

Los beneficios de utilizar energía fotovoltaica son:

- Energía renovable.
- Energía no contaminante.
- Estructura de pequeñas dimensiones.

- Idónea para zonas de difícil acceso.
- Favorece al crecimiento sostenible.
- Bajo costo [11].

1.3.3.1 Módulos fotovoltaicos

Son elementos que se encargan de transformar la energía solar en energía eléctrica, está formado por células fotovoltaicas que conforman módulos fotovoltaicos que pueden tener arreglos en serie o paralelo para alcanzar un nivel de potencia que a su vez depende del número de células que posea [11].

Los paneles solares se asocian con baterías que acumulan la energía obtenida en horas de luz solar para brindar energía durante periodos de oscuridad, y, con el fin de proteger la batería de sobrecargas o sobre descargas se utiliza un regulador de voltaje.

1.3.3.2 Baterías

Son elementos que permiten almacenar energía, son utilizados en los sistemas fotovoltaicos autónomos para brindar disponibilidad de funcionamiento al proyecto que se va a alimentar especialmente en periodos de consumo en que la luz solar no incide sobre el panel solar.

Existen dos tipos de baterías usados comúnmente para estos sistemas, de plomo-ácido y de níquel-cadmio. Brindan salidas de voltaje entre 1.5 y 48 V y un amperaje entre 4 y 34 A [11].

1.3.3.3 Controlador de carga solar

Este dispositivo es el encargado de proteger la vida útil de la batería (limita la carga y descarga), con ello mejora el funcionamiento del sistema fotovoltaico, este aparato es de aspecto sencillo y de bajo coste [12].

1.3.4 Módulos GSM/GPRS

Son dispositivos que permiten comunicarse de manera similar a un teléfono móvil, utiliza las tecnologías GSM y El GPRS las cuales son las más utilizadas en teléfonos móviles. Permiten la transmisión de voz y datos (SMS, Internet), con velocidades de 9 Kbps hasta 171 Kbps [13].

1.3.4.1 SIM 800L

Es un módulo que ofrece datos GSM/GPRS, usa comunicación en serie y es fácil interactuar con la UART de Arduino y Raspberry pi [14].

Características

- Trabaja en las bandas 850/900/1800/1900 MHz.
- Admite llamadas de voz entrantes y salientes (necesita un auricular y un micrófono).
- Permite transmitir y recibir mensajes SMS.
- Consiente el envío y recepción de datos GPRS (TCP/IP, HTTP).
- Interfaz de instrucciones AT con descubrimiento de “transmisión automática”.
- Adicionalmente permite obtener emisiones de radio FM.
- Voltaje de operación 3.4 V - 4.4 V.
- Rango de temperatura -40°C ~ + 85°C.
- Consumo máximo 2 A.

1.3.4.2 SIM 900

Es un módulo que ofrece datos GSM/GPRS, controlados mediante comandos AT (GSM y SIMCOM), cuenta con conexión UART compatible con Arduino [15].

Características

- Trabaja en las bandas 850/900/1800 /1900 MHz.
- Envío y recepción de mensajes cortos.
- Puertos para micrófono y teléfono.
- Temperatura de funcionamiento: -40 °C ~ + 85°C.
- Voltaje de operación de 3.5 a 5.5 V, 5V recomendado.
- Consumo de corriente por pulsos 2000 mA.
- Consumo de corriente continua 500 mA.

1.3.4.3 SIM 908

Es un módulo GSM/GPRS cuatribanda que acopla la tecnología GPS satelital. Este dispositivo permite ahorrar tiempo y costos para desarrollar aplicaciones habilitadas para GPS rastreando cualquier ubicación cuando posee cobertura de señal [16].

Características

- Trabaja en las bandas 850/900/1800/1900 MHz.
- Control a través de comandos AT (GSM 07.07, 07.05 y SIMCOM comandos AT mejorados).
- Voltaje de operación de 3 a 4.8 V.
- Temperatura de funcionamiento: -40 °C ~ + 85°C.
- Consumo máximo de corriente 2A.

1.3.4.4 Modem USB 3G

Es un dispositivo que permite acceder a Internet mediante tecnología 3G, la cual está presentada la evolución de redes GSM (2G) y GPRS (3G), la conectividad se realiza a través de una de las redes de algún operador celular al cual se encuentre suscrito.

Permite conectarse a cualquier computador y sistema operativo mediante su puerto USB [17].

1.3.5 Módulo GPS

1.3.5.1 Ublox NEO 7M

Este módulo permite obtener la geolocalización de un dispositivo, utiliza chips GNSS con la serie NEO-7 que ofrece alta precisión y menores tiempos de adquisición manteniendo una baja potencia del sistema y fácil integración de radiofrecuencia [18].

Características

- Voltaje de operación de 3.3 a 5 V.
- Frecuencia de la señal de pulso de tiempo 0.25 Hz a 10 MHz.
- Temperatura de funcionamiento: -40 °C ~ + 85°C.
- Consumo de corriente de adquisición 35 mA.
- Sensibilidad de seguimiento -161 dBm.
- Sensibilidad de adquisición -160 dBm.
- Canales de adquisición 56.

1.3.5.2 Ublox NEO 6M

Es un módulo receptor GPS con el motor de posicionamiento u-blox 6 de alto rendimiento, son dispositivos flexibles y rentables que brindan una extraordinaria productividad de navegación incluso en los ambientes más difíciles [19].

Características

- Voltaje de operación de 3.3 a 5 V.
- Frecuencia de la señal de pulso de tiempo 0.25 Hz a 1 KHz.
- Temperatura de funcionamiento: -40 °C ~ + 85°C.
- Consumo de corriente máximo 40 mA.
- Sensibilidad de seguimiento -165 dBm.
- Sensibilidad de adquisición -148 dBm.
- Canales de adquisición 50.

1.3.6 Sensores de distancia

Los sensores de distancia son dispositivos que permiten obtener una medida de proximidad a un objeto mediante tecnología de infrarrojo o de ultrasónico.

1.3.6.1 VL53L0X

Es un sensor que tiene una fuente de láser que emite una luz muy fina con la cual permite calcular la distancia hacia los objetos por medio del cálculo de tiempo de vuelo, es decir, el tiempo que se demora la luz para regresar a la fuente de origen.

Tiene un margen de medición muy estrecho, pero con una gran precisión para la detección de objetos sin presentar problemas de linealidad o doble imagen, su desempeño depende de la luz ambiental y la distancia de los objetos al sensor, es decir, cuanto mejor esté iluminado mejor serán sus resultados, por lo que para el proyecto en mención no será muy favorable.

El sensor es pequeño y fácil de implementar la comunicación con una tarjeta de procesamiento se realiza sobre I2C [20].

Tabla 1.2 Especificación técnica del sensor VL53L0X

| Características | Descripción |
|------------------------|---|
| Dimensiones | 4.4x2.4x1 mm |
| Voltaje | 2.6-3.5 V |
| Consumo máximo | 18 mA |
| Rango de temperatura | -20 a 70 °C |
| Ángulo de medición | 25° |
| Rango de medición | Hasta 2 m, independiente de la reflectancia de un objeto. |
| Fuente infrarroja | 940 nm |
| I2C | Bus serial de hasta 400 KHz (modo rápido) Dirección 0x52 |

1.3.6.2 SC-HR04

Este sensor ultrasónico emite un sonar de la misma manera que lo hacen algunos animales con el fin de determinar la distancia a un objeto. Posee alta precisión de hasta 400 cm en la medición sin contacto, a diferencia de los sensores a láser la operación de este sensor no es afectada por la luz solar ni tampoco por materiales oscuros, pero los materiales blandos como telas son difíciles de detectar, cuenta con un transmisor ultrasónico y un módulo receptor [22]. Su precio es menor a los 5 dólares.

Tabla 1.3 Especificación técnica del sensor SC-HR04

| Características | Descripción |
|-------------------------------|--------------------|
| Fuente de alimentación | 5 VDC |
| Corriente de reposo | <2 mA |
| Corriente de trabajo | 15 mA |
| Ángulo efectivo | <15° |
| Rango de distancia | 2 a 400 cm |
| Resolución | 0,3 cm. |
| Ángulo de medición | 30° |
| Frecuencia de ultrasonido | 40 kHz |
| Dimensión | 45x20x15 mm |
| Temperatura de funcionamiento | -20°C a +70° |

1.3.6.3 JSN-SR04T

Este sensor es una mejora del sensor HCR-04 igualmente trabaja con ultrasonido para determinar la distancia de un objeto sin contacto en un rango de 20 a 600 cm, tiene una precisión de alcance de hasta 2 mm, cuenta con un transceptor y un circuito de control. Del mismo modo su trabajo no es perjudicado por la luz solar ni material negro pero los materiales blandos como tela o lana pueden ser difícil de detectar [1].

Los beneficios de este sensor incluyen su pequeño tamaño, bajo consumo energético, buena precisión y especialmente por su resistencia al agua, convirtiéndose en un sensor adecuado para aplicaciones donde estará expuesto a la intemperie. Su precio está en alrededor de 25 dólares.

Tabla 1.4 Especificación técnica del sensor JSN-SR04T

| Características | Descripción |
|-------------------------------|--------------------|
| Fuente de alimentación | 5 VDC |
| Corriente de trabajo | < 8 mA |
| Rango de detección | 20 a 600 cm |
| Precisión | +/- 1 cm |
| Frecuencia de ultrasonido | 40 KHz |
| Dimensión | 42x29x12 mm |
| Resolución | 2 mm |
| Ángulo de medición | 75° |
| Temperatura de funcionamiento | -20°C a +70° |

1.3.7 Tarjetas de procesamiento

Son placas electrónicas programables que pueden emular las funciones de un computador mediante órdenes que son programadas, permiten el desarrollo de diferentes aplicaciones siendo las más utilizadas Arduino y Raspberry Pi.

1.3.7.1 Arduino

Es una tarjeta de procesamiento electrónica de código libre. Ejecuta órdenes mediante un grupo de instrucciones enviadas al microcontrolador [23].

Características Generales:

- Costo bajo: las tarjetas de Arduino tienen un costo relativamente económico en relación con otras tarjetas de procesamiento.
- El software de Arduino (IDE) se puede ejecutar en sistemas de Windows, Macintosh OS X, y Linux, siendo un dispositivo de multiplataforma.

- La programación de Arduino es simple, sencilla y suficientemente flexible para usuarios principiantes y expertos.
- La parte de software es de código libre y su lenguaje se puede expandir mediante bibliotecas de C ++.
- El hardware de Arduino es flexible a modificaciones, los usuarios pueden mejorar o crear su propia versión del módulo de acuerdo a sus necesidades particulares.

1.3.7.1.1 Arduino Uno

Es una tarjeta de procesamiento que se basa en el microcontrolador ATmega328P de código abierto para el desarrollo de múltiples proyectos, de fácil uso y aprendizaje, posee un conector USB, otro para alimentación y un botón de reinicio [24].

Tabla 1.5 Especificación técnica del Arduino Uno.

| Características | Descripción |
|------------------------|---------------|
| Fuente de alimentación | 5 V |
| Pines digitales E/S | 14 |
| Pines PWM Digital E/S | 6 |
| Pines analógicos E | 6 |
| Memoria flash | 32 KB |
| Velocidad de reloj | 16 MHz |
| Tamaño | 68.6x53.4 mm |
| Peso | 25 gr |
| Consumo | 46 mA / 0.8 W |

1.3.7.1.2 Arduino Mega

Este dispositivo está basado en el ATmega 2560, es de código abierto y está diseñado para proyectos de mayor necesidad de recursos como pines E/S, memoria e interfaces. posee un conector USB, otro para alimentación y un botón de reinicio [25].

Esta tarjeta es compatible con las funciones diseñadas para el Arduino Uno.

Tabla 1.6 Especificación técnica del Arduino Mega

| Características | Descripción |
|------------------------|--------------------|
| Fuente de alimentación | 5 V |
| Pines digitales E/S | 54 |
| Pines PWM Digital E/S | 16 |
| Pines analógicos E | 16 |
| Memoria flash | 256 KB |
| Velocidad de reloj | 16 MHz |
| Tamaño | 101.52x53.3 mm |
| Peso | 37 gr |
| Consumo | 93 mA /1 W |

1.3.7.2 Raspberry Pi

Es un computador completo de código libre que tiene las dimensiones de una *credit card*, cuenta con un procesador que funciona a 700 Mhz, que pueden soportar plataformas como: Debian (Raspbian), Fedora (Pidora), Linux, entre otros que le permiten acelerar gráficos 3D, contiene un espacio para la tarjeta microSD que guarda el sistema operativo y los diferentes programas. Esta placa de procesamiento permite ejecutar programas básicos, utilizar el internet y también programar, además de realizar tareas de escritorio como ejecutar hojas de cálculo, editores de texto o juegos [26].

La adquisición del kit de funcionamiento se obtiene por separado ya que solamente incluye la placa. La carcasa, cables y tarjeta microSD se deben adquirir de manera individual.

Para que la Raspberry funcione correctamente en la tarjeta microSD se debe instalar un sistema operativo el cual es gratis y de código abierto [27].

Características Generales

- Microprocesador ARM con potencia que llega a 1GHz.
- RAM mínima de 512 MB.
- Un GPU (*Graphics Processing Unit*, Unidad de Procesamiento Gráfico) Video Core IV
- Puerto USB que permiten la conexión de teclado, ratón o receptor wifi.
- Puerto HDMI (*High-Definition Multimedia Interface*, Interfaz Multimedia de Alta Definición).

1.3.7.2.1 Raspberry Pi Zero W v1.3

Es una computadora completa de un tamaño muy pequeño y más económico en relación con todas las tarjetas Raspberry, pero que brinda las mismas facilidades y funcionalidades de siempre gracias a su procesador ARM soporta el rango completo de distribuciones GNU/Linux, incluyendo Snappy Ubuntu Core [28][26].

Tabla 1.7 Especificación técnica de la Raspberry Pi Zero W v1.3

| Características | Descripción |
|------------------------|--------------------|
| CPU | 1GHz |
| Fuente de alimentación | micro USB 5 V |
| RAM | 512 MB |
| Salida video | Mini HDMI |
| Puerto | micro USB OTG |
| Pines de conexión | 40 |
| Un conector | CSI para cámara |
| Una ranura | microSD |
| Tamaño | 65x30x5 mm |
| Consumo | 200 mA / 1W |

1.3.7.2.2 Raspberry Pi 3 Modelo B+

Es una computadora completa compuesta de un procesador ARMv8, esta tarjeta contiene puertos USB, HDMI y Ethernet. Funciona con distintas plataformas como Linux (Raspbian) y soporta herramientas de software libre como Python, así como Microsoft Windows 10 [29].

Tabla 1.8 Especificación técnica de la Raspberry Pi

| Características | Descripción |
|------------------------|----------------------------|
| CPU | 1.2GHz |
| Fuente de alimentación | micro USB 5 V |
| RAM | 1 GB compartido con la GPU |
| Puerto audio/video | HDMI y mini jack 3.5 mm |
| Cuatro Puertos | USB 2.0 |
| Puerto Ethernet | RJ45 10/100 |
| Interface | Para Display (DSI) |
| Interface | Para cámara (CSI) |
| Una ranura | para tarjeta Micro SD |
| Tamaño | 85.6 mm x 56.5 mm |
| Conector GPIO | 27 pines GPIO |
| Consumo | 350 mA / 2,5 W |

1.3.8 Conceptos para la creación de bases de datos en LAMP

1.3.8.1 MySQL

MySQL (*My Structured Query Language*, Lenguaje de Consulta Estructurado) es una base de datos popular entre desarrolladores web, compatible con cualquier sistema operativo (Linux, Windows, MacOS, etc), o con programas de programación (Apache, PHP, Phpmyadmin).

MySQL es amigable con el desarrollador, contiene un conjunto de funciones avanzadas para administración y soporte técnico, convirtiéndose en un sitio escalable, confiable y seguro, reduciendo el costo y la complejidad para su administración.

Entre sus características están: múltiples motores de almacenamiento que permite tener gran velocidad al momento de hacer operaciones, gran gestión de la capacidad de memoria puede controlar en una tabla millones de registros, contiene filtros de búsqueda compleja a base de patrones, monitoreo de usuarios, permite exportar o importar datos [30].

1.3.8.2 Apache

Es un servidor web libre utilizado por cerca del 50% de los sitios web del mundo, permite instaurar la unión de un servidor con los navegadores web como Google Chrome, entre otros.

Apache es un software multiplataforma, es decir opera correctamente en distintos sistemas operativos, garantiza una comunicación fluida y segura entre cliente-servidor, es personalizable, cuenta con módulos de seguridad, memoria caché y autenticación. [31]

1.3.8.3 Phpmyadmin

Es un conjunto de herramientas de software gratuito que está diseñado para administrar MySQL y gestionar base de datos, tablas, usuarios, etc, a través de la web en lenguaje PHP (*Hypertext Preprocessor*, Preprocesador de Hipertexto). Tiene una interfaz web instintiva, permite exportar los datos en varios formatos, administrar múltiples servidores, contiene búsqueda global de una base de datos y permite transformar el formato de los datos y mostrarlo como imagen o enlace de descarga [32].

1.3.8.4 HyperText Markup Language (HTML)

Es un lenguaje de formato de documentos para hipertexto muy simple y general de formato abierto, es muy utilizado en la creación de páginas estáticas de Internet.

Este tipo de lenguaje desarrolla la estructura y contenidos que aparecen como textos a partir de etiquetas o tags, permite interconectar diversos conceptos y formatos con diversos estilos de programación que permiten la inclusión de imágenes, videos, animaciones, etc. [33].

1.3.8.5 Hypertext Preprocessor (PHP)

Este lenguaje es de código libre, utilizado para la creación web dinámica, puede ser introducido en un código HTML.

La programación en PHP es sencilla enfocado en la creación de scripts en el servidor, puede operar correctamente en todos los sistemas operativos, cuenta con una característica muy útil como MySQL que permite soportar una gran cantidad de bases de datos. [34]

1.3.8.6 Python

Es un lenguaje abierto de programación interpretado, es decir, necesita compilar el código para ser ejecutado con gran rapidez, es compatible con cualquier sistema operativo, es dirigido a objetos, permite el desarrollo de múltiples programas, aplicaciones a servidores de red, páginas web, etc.

Cuenta con un amplio rango de librerías, que permiten obtener diferentes tipos de datos y funciones que ayudan a la programación y acoplamiento de distintos dispositivos conectados a la Raspberry pi [26].

1.3.9 Conceptos de almacenamiento en la Nube

1.3.9.1. Cloud Computing

Tecnología que ofrece servicios remotos a través de la conectividad, ofreciendo mayor flexibilidad en la capacidad de almacenamiento y procesamiento de datos a través de Internet.

1.3.9.2. Amazon EC2

Amazon Elastic Compute Cloud (Amazon EC2) permite la gestión remota de servidores virtuales como Ubuntu, permitiendo el almacenamiento de datos y creación de aplicaciones en la nube de *Amazon Web Services* (AWS).

1.3.9.3. IP Elástica

Son direcciones IP estáticas públicas que fueron diseñadas para el servicio informático *cloud computing* y que puede tener acceso desde internet [50].

2. DESARROLLO DEL SISTEMA

En este capítulo se destallarán los componentes que serán utilizados en la implementación del sistema, configuración de los dispositivos, desarrollo y gestión de la base de datos y página web.

2.1. Selección de elementos

2.1.1. Selección de la Tarjeta de Procesamiento

El sistema a desarrollarse utilizará la tarjeta de procesamiento Raspberry Pi ZERO w v1.3, el cual cuenta con igual número de puertos GPIO, por lo que presta las mismas facilidades que el modelo 3B+, cuenta con dos puertos mini USB y un puerto mini HDMI.

Raspberry Pi Zero w tiene la mayor cantidad de procesamiento que Arduino e igual a los modelos A+ y B+, cuenta con módulo de conectividad inalámbrica, contiene una ranura microSD que permite la expansión de la capacidad de almacenamiento, tiene dimensiones menores a los modelos de Arduino y demás Raspberrys, por lo tanto, es adecuado para el desarrollo del proyecto.

La tabla 2.1 muestra una comparación de las tarjetas de procesamiento.

Tabla 2.1 Comparación de tarjetas de procesamiento

| Tarjetas de procesamiento | Consumo energético | RAM | Costo USD |
|---------------------------|--------------------|--------|-----------|
| Arduino Uno | 46 mA / 0.8 W | 2 KB | 12 |
| Arduino Mega | 93 mA / 1 W | 8 KB | 60 |
| Raspberry Pi Zero w v1.3 | 200 mA / 1 W | 512 MB | 25 |
| Raspberry Pi model 3B+ | 350 mA / 2,5 W | 1 GB | 70 |

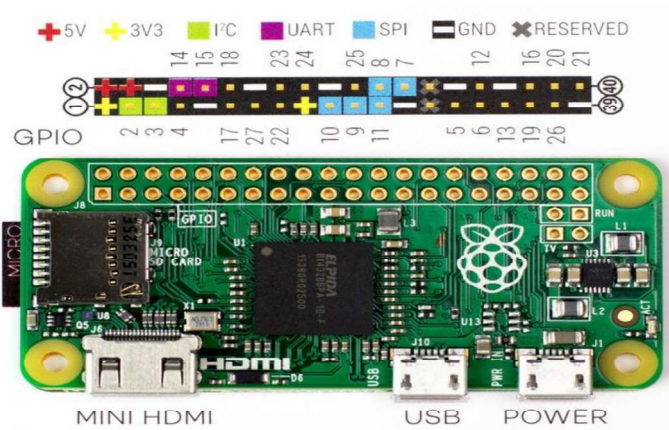


Figura 2.1 Pines Raspberry Pi Zero [52].

La tarjeta de procesamiento Raspberry pi zero cuenta con 40 pines que pueden ser configurados como BCM (numeración GPIO) y BOARD (numeración física).

2.1.2. Selección del sensor de distancia

El sensor JSN-SR04t se utilizará en el desarrollo del proyecto como dispositivo principal, ya que cuenta con mejores características que los dos sensores mencionados, no se ve afectado por lugares y objetos oscuros, soporta temperaturas hasta 70°C, cabe mencionar que su parte delantera puede resistir al agua que lo hace adecuado para el entorno en el cual va a trabajar y tiene un costo asequible.



Figura 2.2 Sensor JSN-SR04t [53].

Junto a este sensor también se utilizarán dos sensores ultrasónicos SC-HR04, con el objetivo de cubrir zonas no alcanzadas por el sensor principal.



Figura 2.3 Sensor HC-SR04 [22].

La tabla 2.2 muestra una comparación de los sensores de distancia.

Tabla 2.2 Comparación de sensores de distancia

| Sensor de distancia | Consumo energético mA | Ángulo de medición | Rango de medición cm |
|---------------------|-----------------------|--------------------|----------------------|
| VL53L0X | 18 | 25° | 2-200 |
| SC-HR04 | 15 | 30° | 2-400 |
| JSN-SR04t | 8 | 75° | 2-600 |

2.1.3. Selección de módulo GPS

Se utilizará el módulo Ublox NEO 7M, debido a que su funcionalidad es compatible con la tarjeta de procesamiento Raspberry pi zero w v1.3, a diferencia del módulo Ublox 6M posee mayor precisión en la geolocalización.



Figura 2.4 Módulo GPS Neo 7M [54].

Tabla 2.3 Comparación de módulos GPS

| Módulo GPS | Consumo energético mA | Sensibilidad de Adquisición dBm | Canales de Adquisición |
|--------------|-----------------------|---------------------------------|------------------------|
| Ublox NEO 7M | 35 | -160 | 56 |
| Ublox NEO 6M | 40 | -148 | 50 |

2.1.4. Selección del módulo GSM/GPRS

Se utilizará un modem USB Huawei E3531, que posee una tarjeta SIM 4G, la cual permitirá acceder a una red celular (GSM, GPRS, EDGE, UMTS, HSDPA y HSUPA) y conexión a Internet, para ello se debe contratar un plan de datos del operador celular al cual se encuentre suscrito.



Figura 2.5 Módem 3G USB Huawei E3531[55].

Características

- HSPA 21.6 Mbps, UMTS 5.76 Mbps.
- GSM, GPRS, EDGE a 850, 900, 1800, 1900 MHz.
- Ecuador y diversidad de recepción.
- Servicio de datos y SMS.
- Ranura para tarjeta microSD.
- Plug and play [55].

Tabla 2.4 Indicador Led Modem USB

| Luz led | Descripción |
|--------------------|---------------------------------------|
| Verde intermitente | Detección de red GPRS |
| Verde continuo | Conectividad a través de GPRS |
| Azul Intermitente | Detección de red 3G o 4G |
| Azul continuo | Conexión a la red a través de 3G o 4G |
| Celeste continuo | Conexión Broad Band 3G o 4G |

2.2. Dimensionamiento del panel solar y batería

El sistema autónomo fotovoltaico permitirá obtener energía eléctrica desde la energía solar (insolación), para alimentar a los diferentes dispositivos, por lo cual su capacidad debe ser abastecer el consumo de potencia total.

2.2.1. Consumo de potencia

La potencia total consumida se debe calcular por en base a las horas de funcionamiento diario.

Tabla 2.5 Comparación de consumos de Potencia.

| Equipo | Cantidad | mA | V | Consumo mW |
|----------------------------|----------|-----|-----|------------|
| Raspberry pi zero V1.3 | 1 | 200 | 5 | 1000 |
| Controlador de carga solar | 1 | 13 | 5 | 65 |
| Modulo GPS Neo 7M | 1 | 35 | 3.3 | 115.5 |
| SC-HR04 y JSN-SR04t | 3 | 45 | 3.3 | 148.5 |
| Modem 3G | 1 | 100 | 5 | 500 |
| Total | | 393 | | 1829 |

Para el realizar los cálculos se debe tener en cuenta la cantidad de insolación global sobre la superficie del panel solar expresado en Kwh/m²/día.

Tabla 2.6 Insolación solar durante un año.

| Mes | Ene | Feb | Mar | Abr | May | Jun | Jul | Ago | Sep | Oct | Nov | Dic |
|-------------------|------|------|------|------|------|------|------|------|------|------|------|------|
| Insolación global | 4.41 | 4.48 | 4.65 | 4.36 | 4.27 | 4.14 | 4.30 | 4.62 | 4.97 | 4.88 | 4.94 | 4.83 |

2.2.2. Elección del panel solar

Para dimensionar el panel solar se calcula de acuerdo a las horas de funcionamiento por día.

$$P_{diario} = 1.829 \text{ W} * 24 \frac{h}{día} = 43.896 \frac{Wh}{día} \quad (1)$$

Una vez obtenida la potencia diaria se calcula la intensidad de corriente total del sistema

$$I_T = \frac{P_{diario}}{V} = \frac{43.896 \text{ Wh/día}}{12 \text{ V}} = 3.658 \frac{Ah}{día} \quad (2)$$

Como en todo proyecto se debe emplear un factor de seguridad para evitar pérdidas

$$I_T = 3.658 * 1.2 = 4.3896 \frac{Ah}{día} \quad (3)$$

La corriente pico del sistema se calcula en base a un valor promedio del índice de radiación, el cual es igual a 4.4 Kwh/m²/día.

$$I_{pico} = \frac{4.3896}{4.4} = 0.99763 \text{ A} \quad (4)$$

Una vez definida la corriente pico se decidirá por un panel solar que cumpla con las características deseadas, en este caso será un panel solar de policristalino.

Características

- Potencia máxima 15 W.
- Voltaje de circuito abierto 21.37 V.
- Voltaje de punto máximo 17.85 V.
- Corriente de cortocircuito 0.96 A.
- Corriente de punto máximo 0.88 A.
- Dimensiones (LxWxH) 350x350x17 cm.

Si bien existe una pequeña diferencia en el amperaje pico calculado con el panel solar seleccionado, abastece de manera útil al proyecto considerando también las dimensiones del panel solar, ya que a mayor capacidad requiere un mayor tamaño por lo que se vuelve vulnerable a daños ya que se encontrará expuesto a la ciudadanía en un contenedor de basura [35]

2.2.3. Elección de la batería

Este elemento es muy importante para el sistema fotovoltaico, por ello se considera un funcionamiento libre de 2 días con un margen de descarga de 0.6.

$$\text{Capacidad nominal de la batería} = 4.3896 \frac{\text{Ah}}{\text{día}} * 2 \text{ días} = 8.7792 \text{Ah} \quad (5)$$

$$\text{Capacidad corregida} = \frac{8.7792}{0.6} = 14.632 \text{ Ah} \quad (6)$$

La batería de protección debe contar con salida de 12 V y 18 Ah, al ser un prototipo que busca elementos de bajo costo se trabaja con una batería con salida de 12 V y 5 Ah, consiguiendo resultados satisfactorios debido a que se utiliza un controlador de voltaje para la alimentación directa y que conmuta el voltaje proveniente del panel solar y la batería cuando existe ausencia de luz solar [35].

Con esta batería se calcula el tiempo de abastecimiento real al prototipo,

$$\text{Capacidad real de la batería} = 4.3896 \frac{\text{Ah}}{\text{día}} * \text{días} \quad (7)$$

$$5\text{Ah} = 4.3896 \frac{\text{Ah}}{\text{día}} * \text{días} \quad (8)$$

$$\frac{5\text{Ah}}{4.3896 \frac{\text{Ah}}{\text{día}}} = 1.139 \text{días} \quad (9)$$

La batería usada cuando existe un consumo máximo abastecerá aproximadamente 28h al prototipo, pero ya que el prototipo cuenta con tiempos muertos en la ejecución de los scripts el tiempo de abastecimiento puede ser superior.

2.2.4 Elección del controlador de carga solar.

Una vez que se han realizados los cálculos para el panel solar y la batería se define el controlador de carga solar de 10 A.



Figura 2.6 Controlador de carga solar [56].

Características:

- Incorporado microcontrolador industrial.
- Configuración de un solo botón, pantalla LCD, reconocimiento automático.
- Totalmente gestión de carga PWM de 3 etapas.
- Protecciones de: cortocircuito, de circuito abierto, de sobrecarga incorporadas.
- Protecciones de la corriente inversa del dual MOS, producciones del bajo calor.

Parámetros:

- Voltaje clasificado: 12V, 24V, Auto.
- Corriente: 10 A.
- Voltaje de la salida del USB: 5 V / 3 A.
- Temperatura de funcionamiento: $-35^{\circ}\sim+60^{\circ}$.
- Con el temporizador y el sensor ligero, podría trabajar para el sistema ligero solar.

2.3. Configuración del sistema

2.3.1. Raspbian

Es la plataforma operativa usada por Raspberry contiene lenguajes de programación como: Python, Scratch, Sonic Pi, Java, entre otros. Se puede instalar mediante NOOBS o descargar la imagen [37].



Figura 2.7 Imagen de Noobs e imagen Iso NOOBS de Raspbian [37].

Si desea instalar el sistema operativo Raspbian mediante la opción ISO, se debe seleccionar la opción descargar ZIP, en este caso se utilizará la opción de *Raspbian with desktop*, seguidamente se descomprime la carpeta ZIP para obtener la imagen ISO.

Para instalar el sistema operativo en la Raspberry es necesario tener tarjeta microSD con capacidad mínima de 4 GB.

Como primer paso se debe formatear la tarjeta microSD, mediante el programa SD Card Formatter que garantiza formatear la tarjeta de forma óptima.

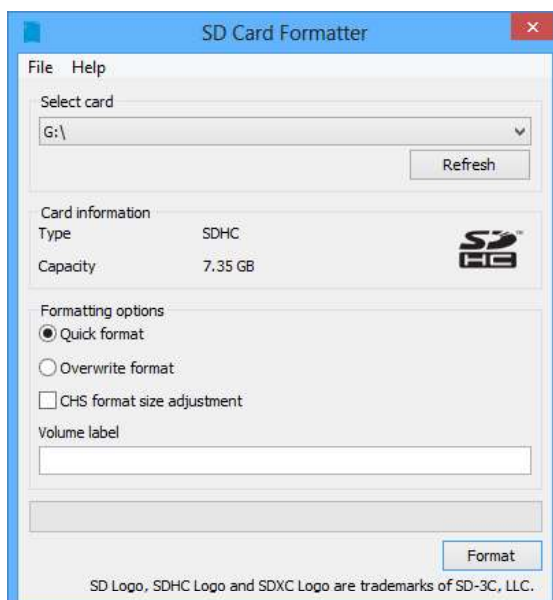


Figura 2.8 Interfaz del programa SD Card Formatter

El siguiente paso es instalar la imagen ISO en la tarjeta microSD, para ello se utiliza el programa Win32DiskImager que permite quemar la imagen en la tarjeta.

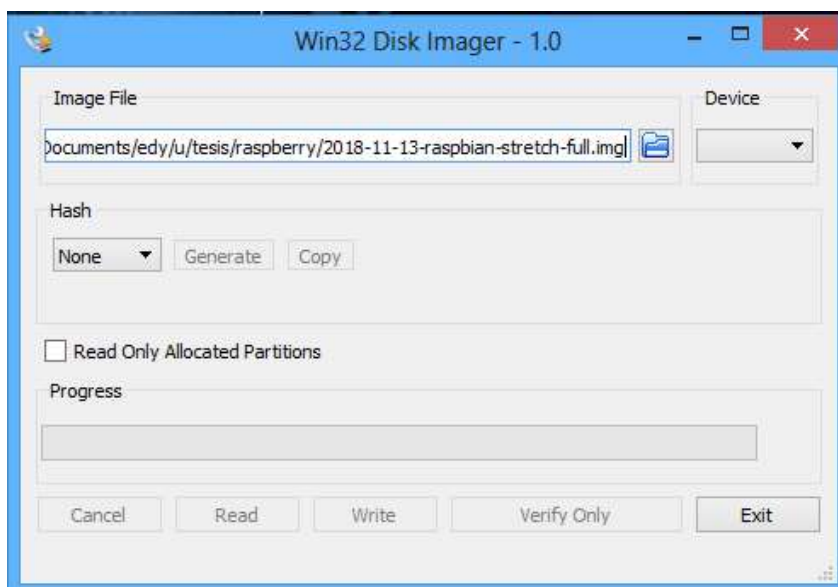


Figura 2.9 Interfaz del programa Win32DiskImager.

Se ubica el directorio de la imagen ISO en la tarjeta microSD y a continuación se selecciona la opción *write*, de esta manera empezará a instalarse el sistema operativo Raspbian en la tarjeta microSD de manera automática.

Una vez que termina la instalación de Raspbian se desea obtener conexión a internet de manera automática para ello se retira y se vuelve a ingresar la tarjeta microSD al computador y se debe crear dos archivos de texto [38].

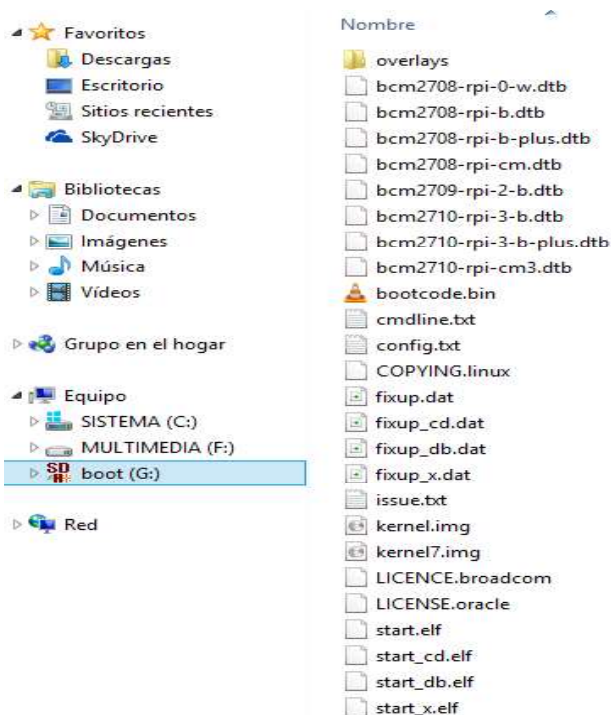


Figura 2.10 Interfaz de los archivos pertenecientes a la microSD.

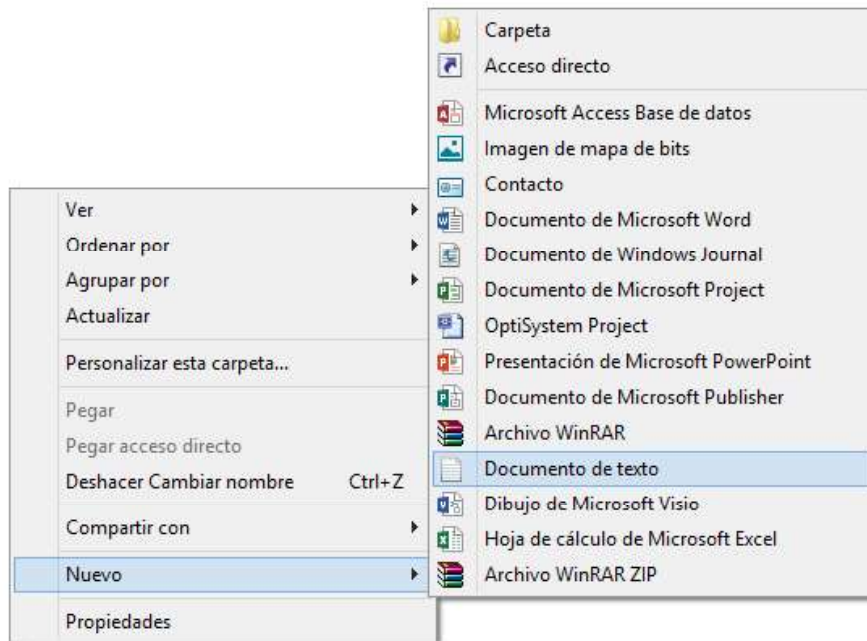


Figura 2.11 Interfaz, nuevo archivo de texto en la microSD.

Un archivo “.txt” se creará con el nombre ssh y otro con wpa_supplicant.conf dentro de este segundo archivo se debe ingresar el siguiente texto.

```
wpa_supplicant.conf.txt: Bloc de notas
Archivo Edición Formato Ver Ayuda
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
country=EC
network={
  ssid="Red H.Acosta"
  psk="hectoracosta"
  key_mgmt=WPA-PSK
}
```

Figura 2.12 Líneas de código de conexión wifi.

Dónde, se deben editar las variables:

- country, correspondiente al país de origen en este caso Ecuador (EC).
- ssid, que es la red local del usuario.
- psk, que es la contraseña a la red local del usuario.

Se guardan los cambios y al final se debe retirar el “.txt” de los archivos creados.

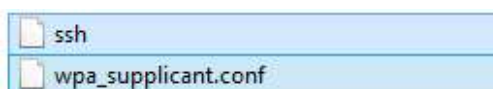


Figura 2.13 Archivos nuevos en la microSD.

Se retira de manera segura la tarjeta microSD y se coloca en la Raspberry.

Es recomendable iniciar la primera vez la Raspberry de manera directa con un monitor, un teclado y un ratón.

Al inicio saldrá una ventana en la cual se debe seleccionar el país, idioma, zona horaria, red wifi y tiene la opción de descargar las actualizaciones recientes.

Una vez dentro de Raspbian, se deben realizar las configuraciones básicas: Menú, Preferences, Raspberry Pi Configuration

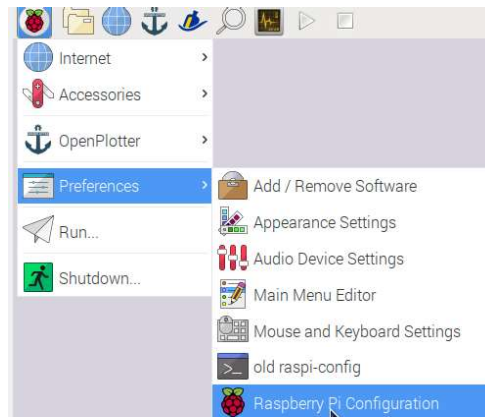


Figura 2.14 Interfaz menú Raspbian.

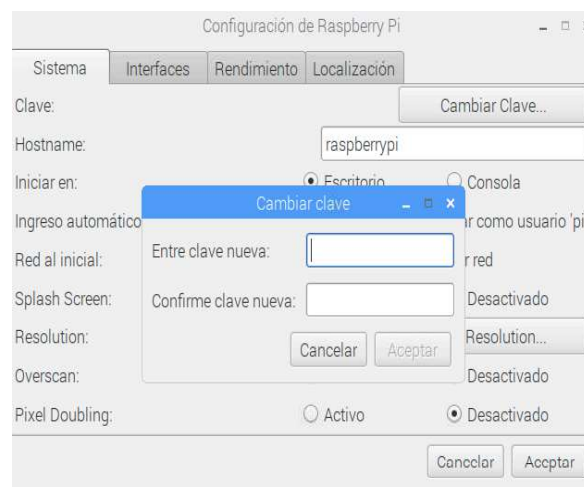


Figura 2.15 Interfaz configuración Raspbian, cambio de contraseña.

En la pestaña sistema se puede modificar el nombre de la Raspberry y cambiar la contraseña de ingreso, mientras que en la pestaña interfaces se deben habilitar las opciones ssh y vnc para una posterior conexión remota.

Como siguiente paso se debe abrir el terminal y digitar los siguientes comandos:

```
sudo apt-get upgrade
```

```
sudo apt-get update
```

Así se actualiza el sistema con las versiones más recientes de los paquetes del sistema operativo.

Se debe habilitar VNC desde la configuración del modo de arranque al cual se accede mediante el comando:

```
sudo raspi-config
```

Se abrirá la ventana de configuración y se selecciona habilitar VNC.

Se recomienda crear una dirección IP estática de tal manera que no se cambie su dirección cada vez que se conecte a una red wifi diferente [26].

Para configurar una IP estática en Raspbian, en el terminal se ingresa el comando:

```
sudo nano /etc/dhcpd.conf
```

Muestra un editor con las interfaces de red conectadas a la Raspberry, en la parte final se debe editar la interfaz wlan0, ya que la Raspberry está conectada de manera inalámbrica.

```
interface wlan0  
  
static ip_address=192.168.1.117/24  
  
static routers=192.168.1.1  
  
static domain_name_servers=192.168.1.1
```

Guardar y finalizar la edición de configuración IP estática.

Reiniciar la Raspberry.

```
sudo reboot
```

2.3.1.1. Conexión Remota por Windows

Windows brinda dos formas sencillas y seguras de conexión remota SSH con la Raspberry, para ello la Raspberry debe estar configurada correctamente como se mostró previamente, es decir, debe contar con conexión a la red de internet y contar con una dirección IP conocida (en este caso es mejor tener una IP estática) [39].

La manera más sencilla de una conexión remota es mediante el programa Putty, el cual se descarga desde su página oficial y acorde a los recursos del computador, una vez descargado el programa, se ejecuta.

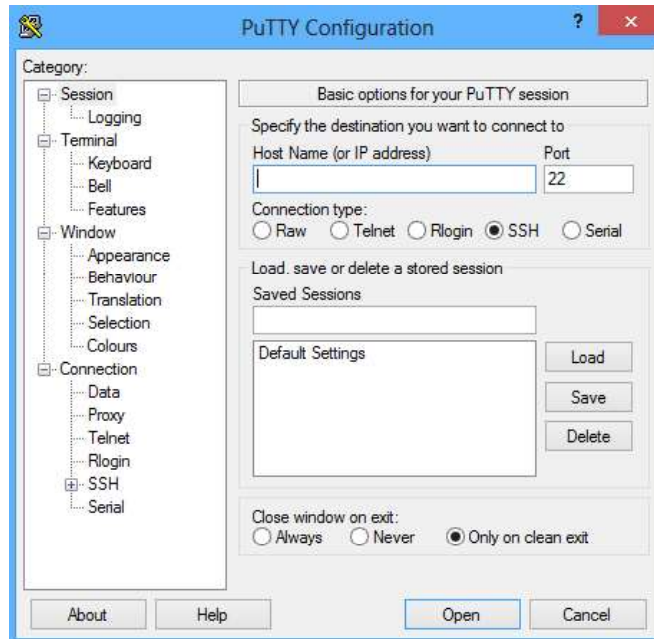


Figura 2.16 Interfaz del programa Putty.

En la casilla *Host Name* ingresar la dirección IP de la Raspberry o su nombre por defecto que es raspberrypi, de esta manera se accede a un terminal, se ingresa el *user* y *password* configuradas previamente, de esta manera se ingresa al terminal de comandos (modo consola) de la Raspberry pi.

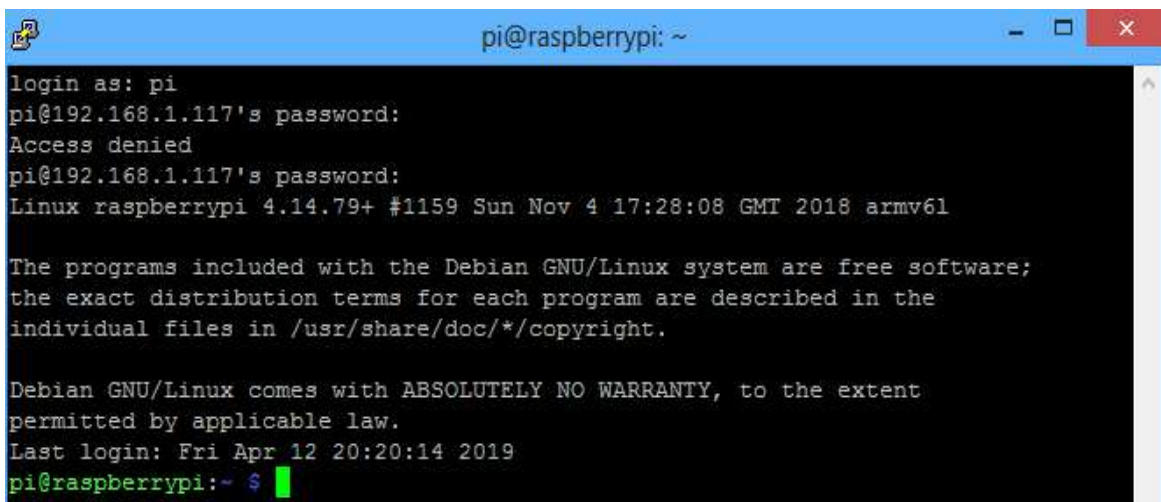


Figura 2.17 Modo consola de la Raspberry Pi mediante Putty.

Otra forma sencilla, pero de manera gráfica es la opción *Conexión a Escritorio Remoto*, antes que nada, se debe acceder al terminal de comandos de la Raspberry mediante Putty e instalar el servidor con el comando:

```
sudo apt-get install xrdp
```

Finalizado el proceso de instalación en Windows abrir el programa de conexión remota, se mostrará una ventana que solicitará una dirección IP a la cual se desea conectar, se introducirá la dirección de la Raspberry.



Figura 2.18 Interfaz conexión remota Windows.

Después solicitará la contraseña de arranque de la Raspberry, se ingresa y se obtiene una conexión con el sistema operativo de la Raspberry [39].

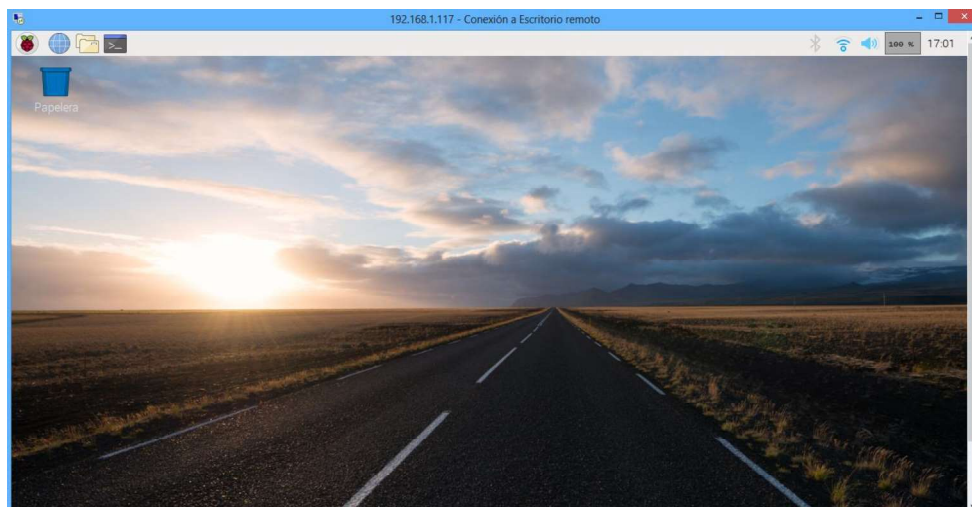


Figura 2.19 Interfaz Raspbian.

2.3.1.2. Conexión Remota por VNC

En el computador descargar el programa real vnc viewer desde la página oficial (registrarse de manera gratuita), posteriormente instalar y ejecutar el programa (iniciar sesión).

Abrir el terminal de comandos de la Raspberry mediante el programa Putty.

A continuación, se instala el servidor vnc y viewer vnc en la Raspberry, digitar los siguientes comandos:

```
sudo apt-get install realvnc-vnc-server
```

```
sudo apt-get install realvnc-vnc-viewer
```

Habilitar el VNC Server con el comando:

```
vncserver :1 -geometry 1280x800 -depth 16 -pixelformat rgb565
```

De esta manera también se define el tamaño de la ventana del programa VNC.

Para una conexión remota, abrir en la Pc el programa VNC Viewer, ubicar la dirección IP o nombre de la sesión creada, en este caso la cuenta se guardó con el nombre raspberrypi.

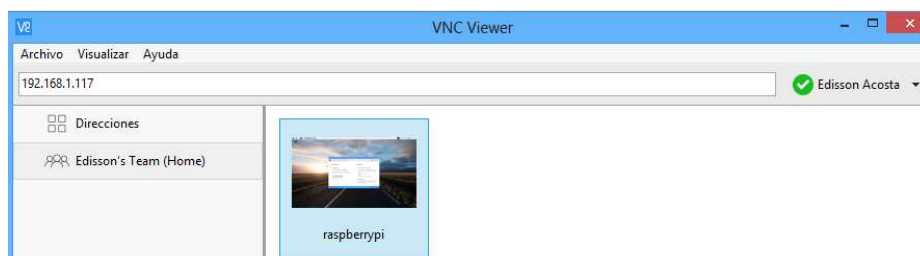


Figura 2.20 Interfaz del programa VNC Viewer.

Al seleccionar la interfaz de la Raspberry conectada, se abrirá una ventana solicitando usuario y contraseña, las cuales son las mismas de la configuración inicial de arranque del sistema operativo.

Después de un tiempo se abrirá una ventana perteneciente a la pantalla de la Raspberry [26].

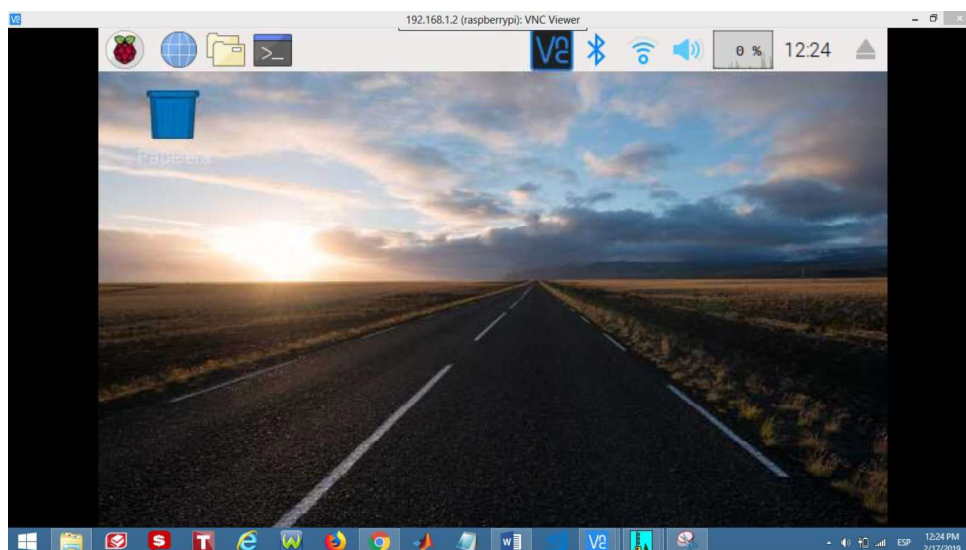


Figura 2.21 Interfaz de Raspbian mediante VNC.

2.3.2. Configuración del sensor de Distancia

A fin de tener un correcto funcionamiento del sensor de distancia (ultrasónico), desde la salida del pin ECHO se debe realizar un divisor de voltaje para hallar el valor de las resistencias y de esta manera obtener un voltaje en la salida de 3.3 V, esto con la finalidad de evitar daños en los pines GPIO [39].

Divisor de Voltaje

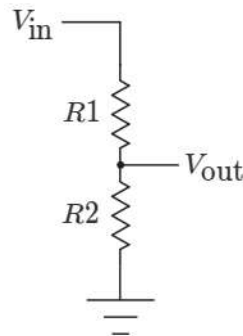


Figura 2.22 Divisor de Voltaje.

$$V_{out} = V_{IN} \frac{R_2}{R_1 + R_2} \quad (9)$$

Donde $V_{ECHO} = V_{IN} = 5 V$, $V_{out} = 3.3 V$:

$$3.3V = 5V \frac{R_2}{R_1 + R_2} \quad (10)$$

Se asume el valor de $R_1 = 1K\Omega$

$$3.3V = 5V \frac{R_2}{1K\Omega + R_2} \quad (11)$$

Se despeja R_2

$$R_2 K\Omega = \frac{3.300}{1.7} = 1.94117 \quad (12)$$

Se aproxima a un valor comercial $R_2 = 2K\Omega$

El funcionamiento del sensor inicia al disparar una señal TRIG en alto, que tarda alrededor de $10\mu S$. A partir de allí se emite un tren de ocho pulsos en la frecuencia de 40KHz. Cuando la señal choca contra un obstáculo, las ondas ultrasónicas se reflejan y la señal de salida ECHO será alta por un período de tiempo necesario para enviar y recibir señales ultrasónicas.

Conexiones:

- Fuente de alimentación 5 V.
- Trigger (TRIG) entrada del sensor.
- Echo (ECHO) salida del Sensor.
- GND (tierra).

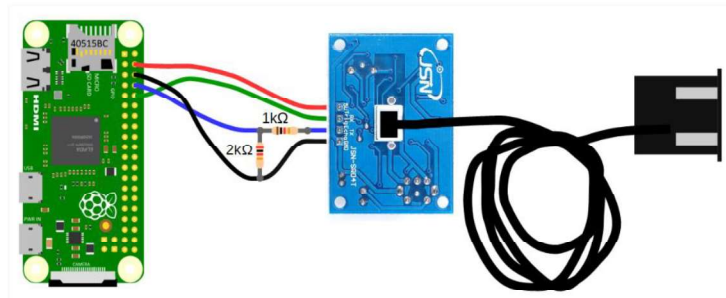


Figura 2.23 Conexión del sensor de distancia [39].

2.3.2.1. Prueba de funcionamiento del sensor de distancia

Con el fin probar el correcto funcionamiento del sensor ultrasónico con la Raspberry es necesario crear un pequeño programa en un archivo “.py”, como se muestra a continuación.

```
pi@raspberrypi: ~/Documents/tesis
GNU nano 2.7.4          Fichero: dist.py
import time             ## libreria de lectura del tiempo
import RPi.GPIO as gpio ## libreria para el control Rpi GPIOs

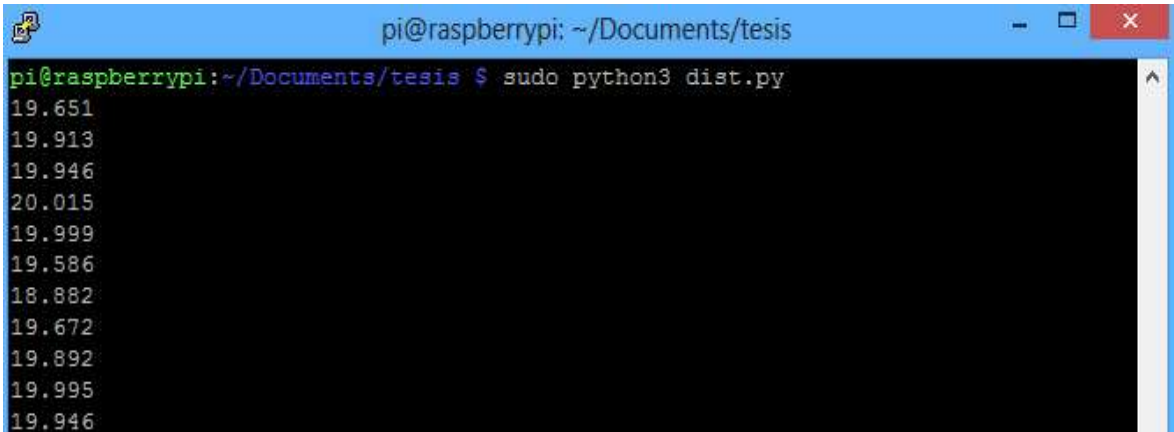
gpio.setwarnings(False) ## desactivamos avisos
gpio.setmode(gpio.BCM)  ## usaremos numeracion gpio modo BCM
trig= 23                 ## conexion a trigger pin 23 salida
echo= 24                 ## conexion a echo pin 24 entrada
gpio.setup(trig,gpio.OUT) ## configuracion TRIGGER como salida
gpio.setup(echo,gpio.IN)  ## configuracion ECHO como entrada
gpio.output(trig, False)  ## configuracion de trigger en bajo

while True:
    gpio.output(trig,True) ## se envia un pulso de ultrasonido
    time.sleep(0.0001)     ## retardo
    gpio.output(trig,False) ## se apaga el pulso
    tin= time.time()      ## se guarda el tiempo inicial
    while gpio.input(echo)==0: ## mientras el sensor no reciba señal
        tin=time.time()
    while gpio.input(echo)==1: ## mientras el sensor reciba señal
        tfin=time.time()
    tttotal= tfin-tin      ## se obtiene el tiempo total
    distancia= (34300*tttotal)/2 ## se obtiene la distancia
    print('%s.3f'%distancia)
```

Figura 2.24 Líneas de código del sensor de distancia.

Una vez conectado correctamente el sensor, se ejecuta el programa:

```
sudo python nombreprograma.py.
```



```
pi@raspberrypi: ~/Documents/tesis
pi@raspberrypi:~/Documents/tesis $ sudo python3 dist.py
19.651
19.913
19.946
20.015
19.999
19.586
18.882
19.672
19.892
19.995
19.946
```

Figura 2.25 Resultados de mediciones del sensor de distancia.

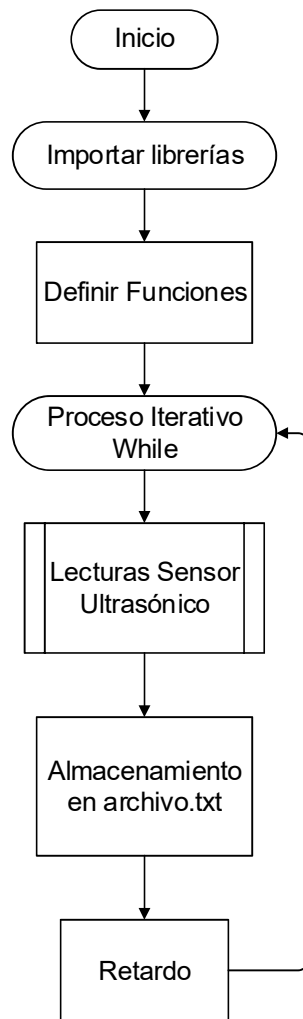


Figura 2.26 Diagrama de flujo del programa de distancia

2.3.3. Configuración del módulo GPS

El módulo GPS UBLOX NEO 7M trabaja con los pines UART de los puertos serie GPIO que deben ser habilitados. Primero se edita el archivo “config.txt”; en el terminal digitar:

```
sudo nano /boot/config.txt
```

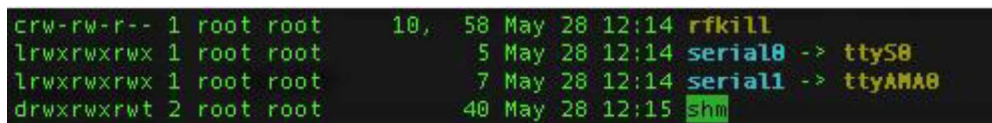
y en la parte inferior añadir la siguiente línea:

```
enable_uart=1
```

Los pines GPIO 14 y 15, son los puertos UART o puertos serie ttyAMA0 o ttyS0, dependiendo de la tarjeta Raspberry que se utilice, para conocer cual es el correcto se debe fijar el que lleve el nombre “serial 0”, mediante el comando:

```
ls -l /dev
```

al ingresar este comando se desplegará una lista de los puertos habilitados en la Raspberry en uso.



```
crw-rw-r-- 1 root root    10,  58 May 28 12:14 rfdkill
lrwxrwxrwx 1 root root    5 May 28 12:14 serial0 -> ttyS0
lrwxrwxrwx 1 root root    7 May 28 12:14 serial1 -> ttyAMA0
drwxrwxrwt 2 root root   40 May 28 12:15 shm
```

Figura 2.27 Puertos habilitados en la Raspberry Pi.

Como muestra la imagen el puerto serie en uso de la Raspberry pi zero W v1.3 es el ttyS0, dicho puerto debe ser habilitado de manera correcta para que el módulo GPS funcione.

Primero se debe desactivar la consola con los comandos:

```
sudo systemctl stop serial-getty@ttyS0.service
```

```
sudo systemctl disable serial-getty@ttyS0.service
```

Se debe editar el archivo “cmdline.txt” para desactivar la consola ahí también. Para ello digitar:

```
sudo nano /boot/cmdline.txt
```

se debe eliminar la línea console = serial0,115200, de tal manera que el archivo quede de la siguiente manera “dwc_otg.lpm_enable=0 console=tty1 root=/dev/mmcblk0p2 rootfstype=ext4 elevator=deadline fsck.repair=yes rootwait” [40].

Después guardar los cambios y reiniciar la tarjeta para que los cambios tengan efecto.

2.3.3.1. Prueba de funcionamiento del módulo GPS

Los pines GPIO 14 (Tx) y 15 (RX) de la Raspberry se conectan de manera cruzada con el módulo GPS.

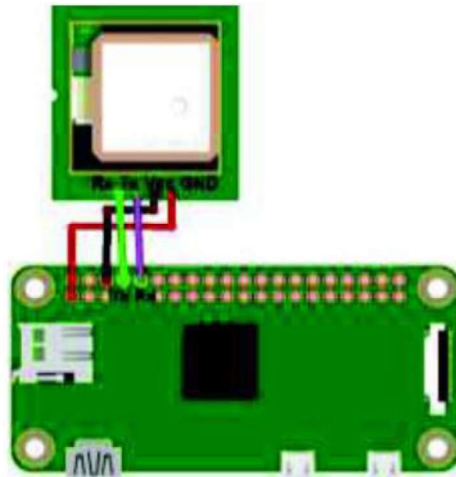


Figura 2.28 Conexión módulo GPS [41].

Se debe verificar que el puerto serie ttyS0 se encuentre bien configurado mediante el comando:

```
sudo cat /dev/ttyS0
```

Se desplegará un flujo de datos GPG que indicarán el correcto funcionamiento de los puertos serie obteniendo comunicación con el módulo GPS.

```
pi@raspberrypi: ~  
pi@raspberrypi:~$ sudo cat /dev/ttyS0  
$GPRMC,225150.00,A,0116.88535,S,07836.85733,W,0.116,,120419,,A*72  
  
$GPVTG,,T,,M,0.116,N,0.215,K,A*23  
  
$GPGGA,225150.00,0116.88535,S,07836.85733,W,1,10,0.75,2729.7,M,12.6,M,,*68  
  
$GPGSA,A,3,30,06,28,07,02,17,09,19,03,13,,1.48,0.75,1.27*05  
  
$GPGSV,4,1,16,02,31,250,12,03,12,050,22,04,21,156,19,05,05,212,17*70  
$GPGSV,4,2,16,06,55,291,26,07,25,160,32,09,17,139,29,13,05,250,26*7D  
$GPGSV,4,3,16,17,19,356,27,19,18,332,26,23,01,108,12,28,51,028,27*7E  
$GPGSV,4,4,16,30,54,189,29,46,32,271,30,48,28,271,,51,57,272,*70  
  
$GPGLL,0116.88535,S,07836.85733,W,225150.00,A,A*62  
  
$GPTXT,01,01,01,NMEA unknown msg*58  
$GPTXT,01,01,01,NMEA unknown msg*58
```

Figura 2.29 Información de puertos aptos para el sensor GPS.

Ahora se instalará algunas funciones de GPSD que permitirán interpretar los datos obtenidos por el módulo GPS, digitar:

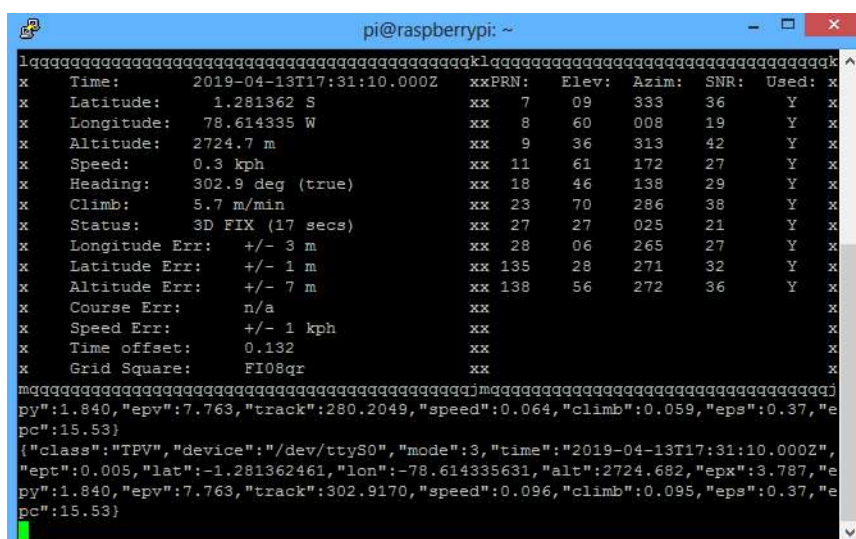
```
sudo apt-get install gpsd gpsd-clients python-gps
```

GPSD transmite los datos mediante un socket, que debe activarse cada vez que arranque el sistema de la Raspberry mediante:

```
sudo gpsd /dev/ttyS0 -F /var/run/gpsd.sock
```

En este momento el socket estará activo, lo que permite probar el funcionamiento del módulo GPS con el siguiente comando:

```
cgps -s
```



```
pi@raspberrypi: ~  
lqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqk  
x Time: 2019-04-13T17:31:10.000Z xxPRN: Elev: Azim: SNR: Used: x  
x Latitude: 1.281362 S xx 7 09 333 36 Y x  
x Longitude: 78.614335 W xx 8 60 008 19 Y x  
x Altitude: 2724.7 m xx 9 36 313 42 Y x  
x Speed: 0.3 kph xx 11 61 172 27 Y x  
x Heading: 302.9 deg (true) xx 18 46 138 29 Y x  
x Climb: 5.7 m/min xx 23 70 286 38 Y x  
x Status: 3D FIX (17 secs) xx 27 27 025 21 Y x  
x Longitude Err: +/- 3 m xx 28 06 265 27 Y x  
x Latitude Err: +/- 1 m xx 135 28 271 32 Y x  
x Altitude Err: +/- 7 m xx 138 56 272 36 Y x  
x Course Err: n/a xx x  
x Speed Err: +/- 1 kph xx x  
x Time offset: 0.132 xx x  
x Grid Square: FI08qr xx x  
mqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqj  
py":1.840,"epv":7.763,"track":280.2049,"speed":0.064,"climb":0.059,"eps":0.37,"e  
pc":15.53}  
{ "class": "TPV", "device": "/dev/ttyS0", "mode": 3, "time": "2019-04-13T17:31:10.000Z",  
"epc": 0.005, "lat": -1.281362461, "lon": -78.614335631, "alt": 2724.682, "epx": 3.787, "e  
py": 1.840, "epv": 7.763, "track": 302.9170, "speed": 0.096, "climb": 0.095, "eps": 0.37, "e  
pc": 15.53}
```

Figura 2.30 Prueba de funcionamiento del sensor GPS.

La imagen mostrada es el resultado si todo se encuentra correctamente configurado, ahora se puede hacer un script en Python para obtener los datos, mediante el comando:

```
sudo nano nombreachivo.py
```

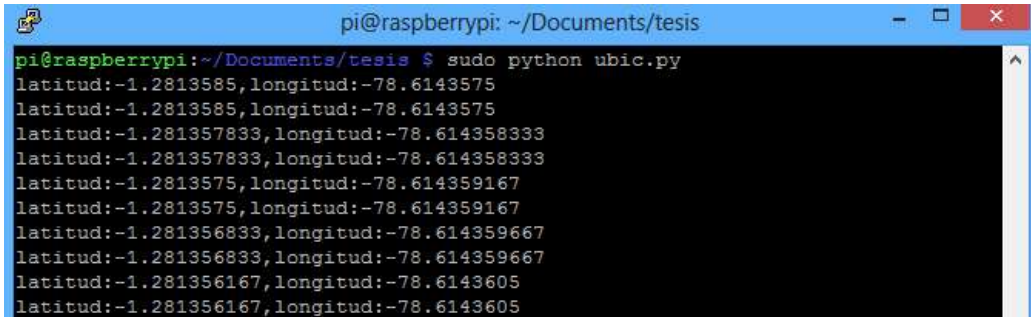
```
from sys import argv  
import gps  
import requests  
  
#escucho al puerto 2947 de gpsd  
session = gps.gps("localhost", "2947")  
session.stream(gps.WATCH_ENABLE | gps.WATCH_NEWSTYLE)  
  
while True :  
    rep = session.next()  
    if (rep["class"] == "TPV") :  
        print("latitud:" + str(rep.lat) + ",longitud:" + str(rep.lon))
```

Figura 2.31 Líneas de código del programa del sensor GPS.

Para ejecutar el programa, ingresar el comando:

```
sudo python nombrearchivo.py
```

Si no existe ningún error se visualizarán los datos de latitud y longitud tal como se muestra en la imagen.



```
pi@raspberrypi: ~/Documents/tesis
pi@raspberrypi:~/Documents/tesis $ sudo python ubic.py
latitud:-1.2813585, longitud:-78.6143575
latitud:-1.2813585, longitud:-78.6143575
latitud:-1.281357833, longitud:-78.614358333
latitud:-1.281357833, longitud:-78.614358333
latitud:-1.2813575, longitud:-78.614359167
latitud:-1.2813575, longitud:-78.614359167
latitud:-1.281356833, longitud:-78.614359667
latitud:-1.281356833, longitud:-78.614359667
latitud:-1.281356167, longitud:-78.6143605
latitud:-1.281356167, longitud:-78.6143605
```

Figura 2.32 Resultados del programa del sensor GPS.

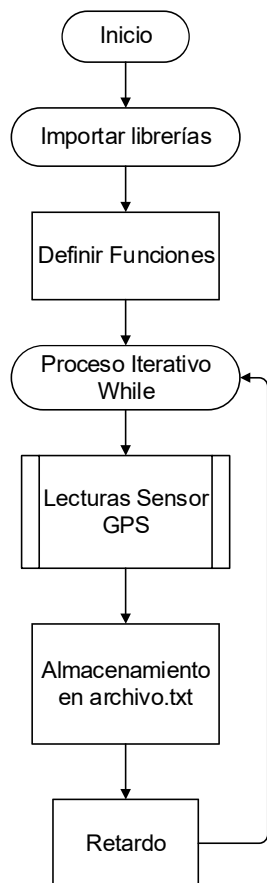


Figura 2.33 Diagrama de flujo del programa de ubicación

Es necesario automatizar el GPSD, para que en cada arranque se pueda obtener los valores del GPS sin necesidad de ingresar el comando del sock.

Se debe crear un archivo en /etc/rc.local, digitar:

```
sudo nano /etc/rc.local
```

Añadir en la parte final las siguientes líneas.

```
#!/bin/sh -e  
  
sudo gpsd /dev/ttyS0 -F /var/run/gpsd.sock  
  
exit 0
```

De esta manera se ejecutará la transmisión socket en cada arranque [41].

2.3.4. Configuración del módulo GSM/GPRS

El prototipo debe conectarse a internet con el propósito de enviar las mediciones del sensor de distancia y la ubicación obtenida del módulo GPS, además que debe enviar mensajes de alerta cuando el contenedor de basura se encuentre en nivel alto.

Para la configuración del modem USB 3G es necesario que se encuentre correctamente conectado a la Raspberry.

Al principio el dispositivo 3G no será reconocido como modem, ya que la Raspberry lo considera un dispositivo de almacenamiento, se puede observar al digitar el comando “lsusb” y mostrará la identificación actual del dispositivo en este caso “bus 001 Device 002: ID 12d1:15ca Huawei (mass storage mode)”, para solucionar este problema se debe desconectar el modem USB 3G e instalar la herramienta “usb-modeswitch” digitando:

```
sudo apt-get install usb-modeswitch
```

Este programa permite el cambio automático de la operación, una vez finalizada la instalación este modelo de modem necesita otra configuración.

Se reinicia la tarjeta y se vuelve a conectar el modem USB 3G a la Raspberry antes de iniciarla, a continuación, se ejecuta el comando “lsusb” para comprobar que se reconoce el modem y conocer el código de identificación correspondiente.

```
pi@raspberrypi:~$ lsusb  
Bus 001 Device 003: ID 12d1:1506 Huawei Technologies Co., Ltd. Modem/Networkcard
```

Figura 2.34 Información del estado del modem

Como se observa en la figura el dispositivo ya se encuentra reconocido como modem y con la identificación del producto cambiada a 1506.

Se puede comprobar los estados de operación digitando:


```
sudo tail -f /var/log/messages
```

Después se debe instalar el servicio pppd digitando:

```
sudo apt-get install ppp
```

Este programa permite acceder a Internet, utiliza el protocolo PPP (*Point to Point Protocol*, Protocolo Punto a Punto).

Por último, se instalan dos programas, “Sakis3g” que permite realizar la conexión 3G de manera interactiva con el entorno Raspbian y UMTSkeeper cuya función es comprobar constantemente la conexión y restaurarla cada vez que el servicio se interrumpa.

Primero se crea un directorio con el nombre “umtskeeper”; se accede al directorio donde se descargará, descomprimirá y se dará permisos a los archivos necesarios mediante los siguientes comandos:

```
sudo wget "http://mintakaconciencia.net/squares/umtskeeper/src/umtskeeper.tar.gz"
sudo tar -xzf umtskeeper.tar.gz
sudo chmod +x sakis3g umtskeeper
```

Finalmente se configure el programa “Sakis3g”, con el APN (*Access Point Name*, Nombre del Punto de Acceso), usuario y contraseña, (“ba.amx”, claro. claro) correspondientes a la operadora móvil a utilizar en este caso de “Claro”; se ejecuta:

```
sudo ./sakis3g --interactive
```

De esta manera aparecerá una ventana de configuración y se seleccionará las opciones correspondientes al modem en uso [42].

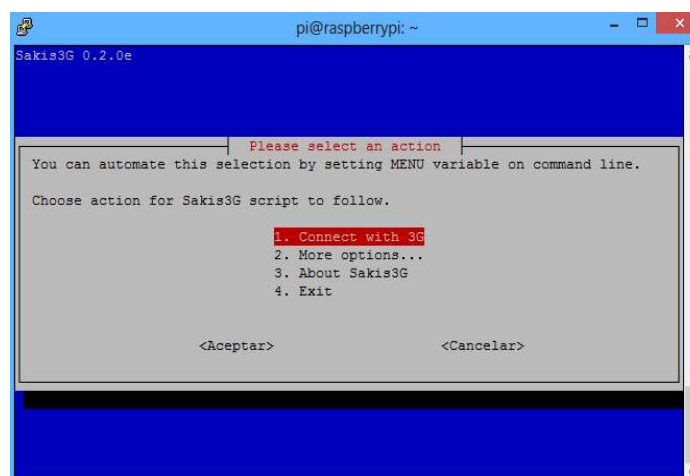


Figura 2.35 Interfaz del menú de Sakis3.

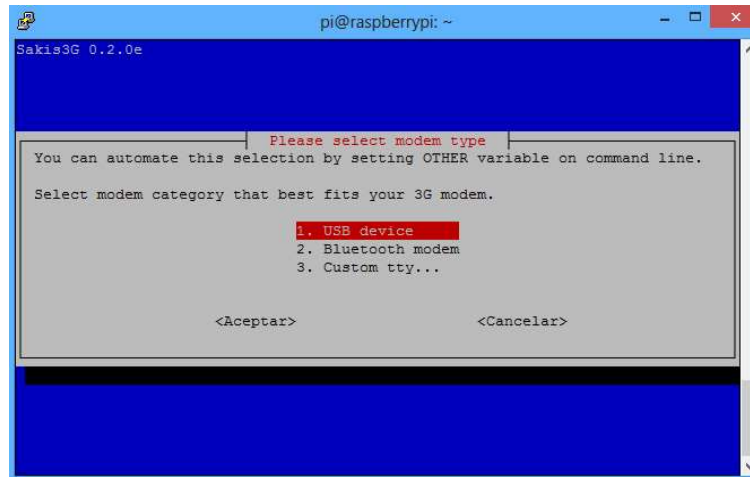


Figura 2.36 Opciones de conexión del módem.

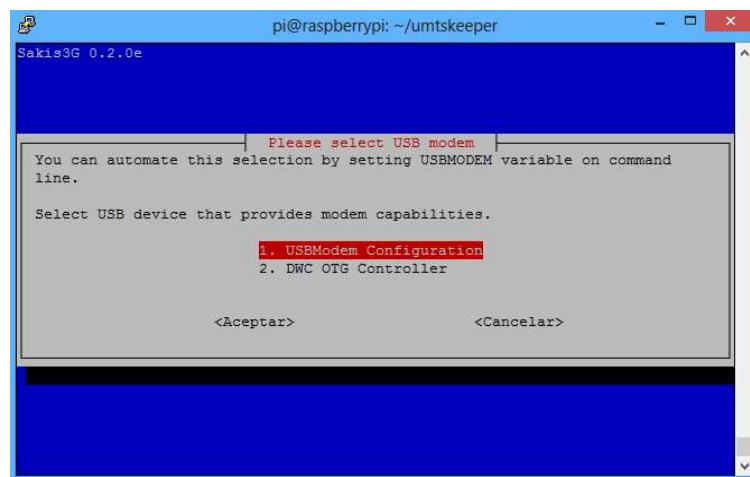


Figura 2.37 Opciones de elementos conectados.

Una vez llegado a este punto de la configuración, si el programa no reconoce automáticamente el APN correspondiente, se puede ingresar de forma manual en *Current APN*.

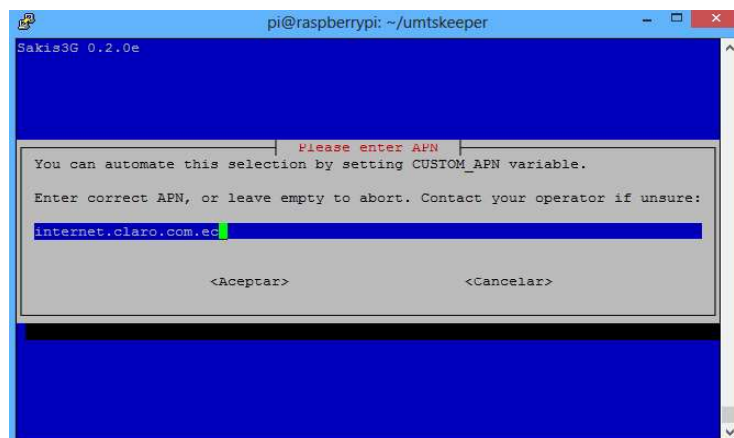


Figura 2.38 Ingreso de APN de la operadora.

A continuación, solicitará el usuario y la contraseña pertenecientes a la operadora móvil, se abrirá un aviso de conectividad si los datos ingresados son correctos.

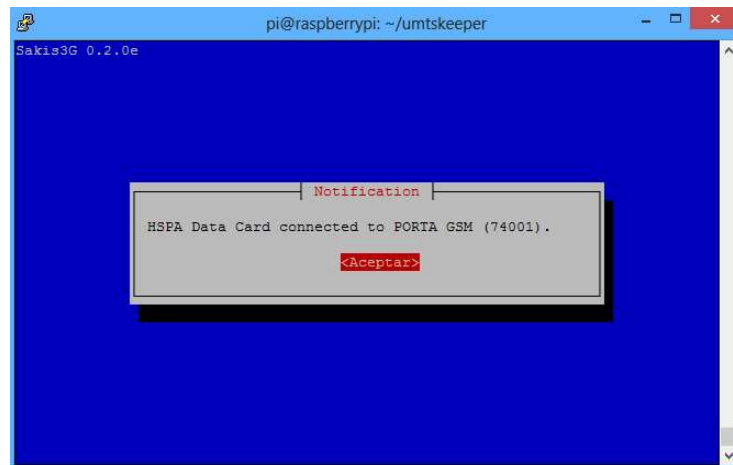


Figura 2.39 Conexión exitosa

2.3.4.1. Prueba de funcionamiento del Modem 3G

Una vez finalizado los pasos e ingresadas las opciones correctas, se despliega una ventana indicando que el modem USB 3G se encuentra conectado; luego se selecciona la opción *Connection Information* que mostrará los datos correspondientes a la conexión.

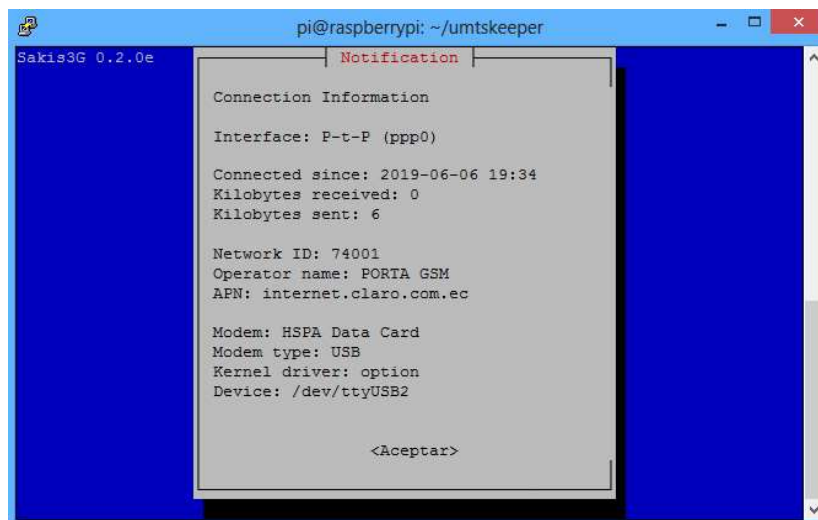


Figura 2.40 Información de conexión del módem.

En el terminal también se puede comprobar la conexión del modem por ppp0, mediante el comando "ifconfig".

```
ppp0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST> mtu 1500
    inet 10.65.167.163 netmask 255.255.255.255 destination 10.64.64.64
    ppp txqueuelen 3 (Point-to-Point Protocol)
    RX packets 8 bytes 158 (158.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 11 bytes 355 (355.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Figura 2.41 Información de conexión del modem mediante consola.

Existe otra manera de configurar sakis3g sin necesidad de ingresar al modo interactivo que es digitando “sudo nano /etc/sakis3g.conf” y editar el archivo con la información necesaria del modem y del operador.

```
GNU nano 2.7.4 Fichero: /etc/sakis3g.conf Modificado
OTHER="USBMODEM"
USBMODEM="12d1:1506"
USBINTERFACE="0"
APN="CUSTOM_APN"
CUSTOM_APN="ba.amx"
APN_USER="claro"
APN_PASS="claro"
```

Figura 2.42 Archivo Sakis3g

Si todo está correcto el modem se conectará y se puede verificarlo mediante el comando “sudo ./sakis3g connection status”

```
pi@raspberrypi:~/umtskeeper $ sudo ./sakis3g connection status
E3531 connected to Claro (74001).
```

Figura 2.43 Estado de Conexión del modem.

Para comprobar su funcionamiento se consulta el fichero /var/log/umtskeeper.log [35][42][48].

2.3.5. Configuración de los servidores de la base de datos

La creación de la base de datos se hará en MySQL, y será administrada mediante la herramienta phpMyAdmin que trabaja con lenguaje PHP y permite gestionar bases de datos de MySQL mediante páginas web.

El nombre de la base de datos será “administracion”, y la tabla de valores se llamará “contenedores”. La tabla tendrá 8 columnas: id, fecha y hora, distancia, cantidad, ubicación, vacío, medio y lleno.

2.3.5.1. Creación de un servidor web en la nube.

La creación de este servidor permitirá a los usuarios almacenar y visualizar los valores obtenidos de las pruebas realizadas en la Raspberry, desde cualquier red remota mediante una página web.

El servidor web se lo realizará en *Amazon Web Services (AWS)*, que contiene múltiples servicios convirtiéndose en una de las mejores plataformas de *cloud computing*. Se creará la instancia *Amazon Elastic Compute Cloud (Amazon EC2)*, diseñado para un simple uso de la nube de manera segura y capacidad modificable [50].

El servicio AWS permite crear una cuenta gratuita durante 12 meses, aunque es necesario contar con una tarjeta de crédito o débito como respaldo. Desde un navegador ingresar a <https://aws.amazon.com/es/free/> y crear las credenciales que permitan la autenticación a la cuenta, después de llenar toda la información que solicite la empresa y validar la cuenta, acceder a inicio de sesión en la consola.

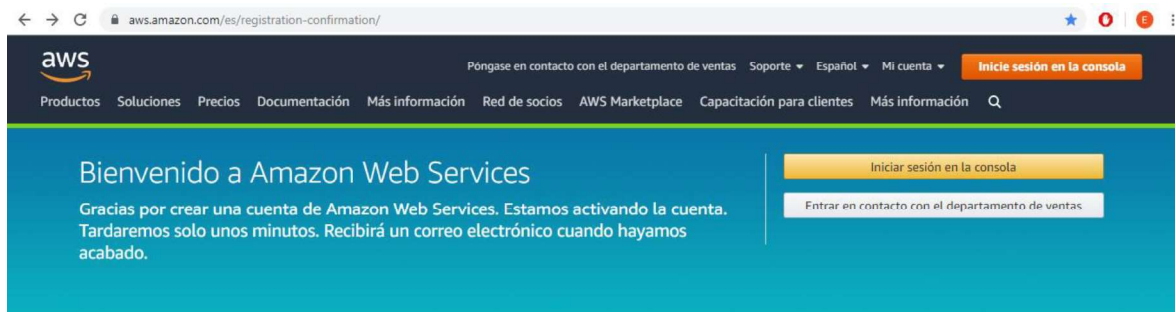


Figura 2.44 Inicio de sesión en AWS.

A continuación, se escoge una instancia en la cual se configurará la máquina virtual en el servidor, en el panel de opciones seleccionar *Instances* y después *Launch Instance*.

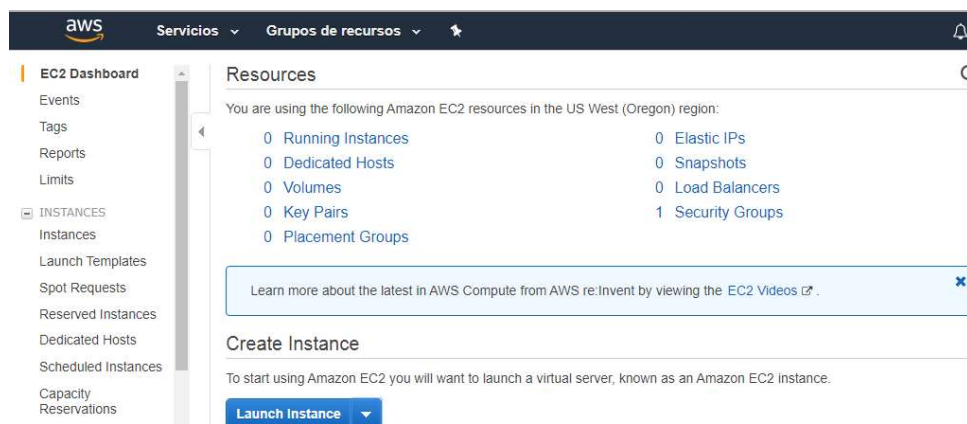


Figura 2.45 Creación de instancia Ec2

Escoger una imagen para la máquina virtual de Amazon, para este proyecto se escogió Ubuntu 16.04.

Step 1: Choose an Amazon Machine Image (AMI)

The screenshot shows the 'Choose an Amazon Machine Image (AMI)' step in the AWS console. It features a search bar and a list of AMIs. The selected AMI is 'Ubuntu Server 16.04 LTS (HVM), SSD Volume Type' with the ID 'ami-0b37e9efc396e4c38'. Other visible AMIs include 'Microsoft Windows Server 2019 Base' and 'Deep Learning AMI (Ubuntu) Version 24.1'. A 'Cancel and Exit' button is visible in the top right corner.

Figura 2.46 Máquinas virtuales de AWS

Escoger el tipo de instancia t2.micro que ofrece un servicio gratuito.

Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. [Learn more](#) about instance types and how they can meet your computing needs.

Filter by: **All Instance types** | **Current generation** | **Show/Hide Columns**

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

| | Family | Type | vCPUs | Memory (GiB) | Instance Storage (GB) | EBS-Optimized Available | Network Performance | IPv6 Support |
|-------------------------------------|-----------------|---|-------|--------------|-----------------------|-------------------------|---------------------|--------------|
| <input type="checkbox"/> | General purpose | t2.nano | 1 | 0.5 | EBS only | - | Low to Moderate | Yes |
| <input checked="" type="checkbox"/> | General purpose | t2.micro <small>Free tier eligible</small> | 1 | 1 | EBS only | - | Low to Moderate | Yes |
| <input type="checkbox"/> | General purpose | t2.small | 1 | 2 | EBS only | - | Low to Moderate | Yes |
| <input type="checkbox"/> | General purpose | t2.medium | 2 | 4 | EBS only | - | Low to Moderate | Yes |
| <input type="checkbox"/> | General purpose | t2.large | 2 | 8 | EBS only | - | Low to Moderate | Yes |

Figura 2.47 Tipos de instancias

En este paso no se debe modificar nada.

Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

The screenshot shows the 'Configure Instance Details' step. Key configuration options include:

- Number of instances:** 1 (with a 'Launch into Auto Scaling Group' link)
- Purchasing option:** Request Spot instances (checked)
- Network:** vpc-8542c5fd (default) (with 'Create new VPC' link)
- Subnet:** No preference (default subnet in any Availability Zone) (with 'Create new subnet' link)
- Auto-assign Public IP:** Use subnet setting (Enable)
- Placement group:** Add instance to placement group (checked)
- Capacity Reservation:** Open (with 'Create new Capacity Reservation' link)
- IAM role:** None (with 'Create new IAM role' link)
- Shutdown behavior:** Stop

 At the bottom, there are buttons for 'Cancel', 'Previous', 'Review and Launch', and 'Next: Add Storage'.

Figura 2.48 Detalles de la instancia

Como siguiente paso se debe escoger la capacidad de almacenamiento, el servicio gratuito permite tener entre 8 y 30 GB.

Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

| Volume Type | Device | Snapshot | Size (GiB) | Volume Type | IOPS | Throughput (MB/s) | Delete on Termination | Encryption |
|-------------|-----------|------------------------|------------|---------------------------|------------|-------------------|-------------------------------------|---------------|
| Root | /dev/sda1 | snap-0ea2cba30dfc1b01c | 8 | General Purpose SSD (gp2) | 100 / 3000 | N/A | <input checked="" type="checkbox"/> | Not Encrypted |

[Add New Volume](#)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

[Cancel](#)
[Previous](#)
[Review and Launch](#)
[Next: Add Tags](#)

Figura 2.49 Almacenamiento de la instancia.

El paso 5 es opcional, puede ser o no modificado, pero se recomienda dejarlo por *default*.

El paso 6 es el más importante, aquí se realiza la asignación de puertos para los diferentes servicios a los cuales se desea ingresar desde cualquier red exterior.

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: Create a new security group Select an existing security group

Security group name:

Description:

| Type | Protocol | Port Range | Source | Description |
|------------|----------|------------|-----------------------------------|----------------------------|
| SSH | TCP | 22 | Custom 0.0.0.0/0 | e.g. SSH for Admin Desktop |
| HTTP | TCP | 80 | Custom 0.0.0.0/0, :::0 | e.g. SSH for Admin Desktop |
| Custom TCP | TCP | 0 | Custom CIDR, IP or Security Group | e.g. SSH for Admin Desktop |

[Add Rule](#)

Warning

[Cancel](#)
[Previous](#)
[Review and Launch](#)

Figura 2.50 Configuración de grupos de seguridad AWS.

Una vez realizada la configuración el último paso es validar la configuración realizada en la instancia verificando que no se generen cobros.

A continuación, se debe crear una clave “.pem”, la cual debe ser cambiada a un formato “.ppk” mediante el programa “Putty Generator” que servirá para ingresar a la máquina virtual de Amazon de manera remota por el programa Putty. Guardar la clave en un lugar seguro.

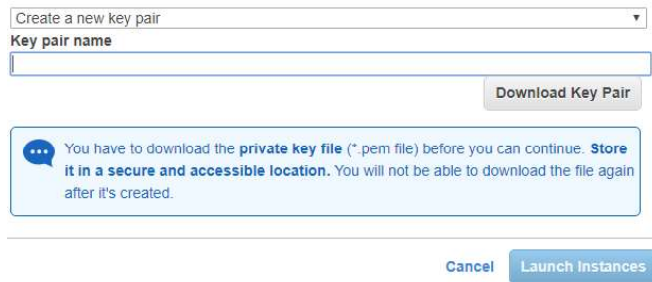


Figura 2.51 Creación de clave pem

Finalizado el proceso, se observará que la instancia creada se encuentra en un estado “running”, la cual indica que el servidor en la máquina se encuentra activa.

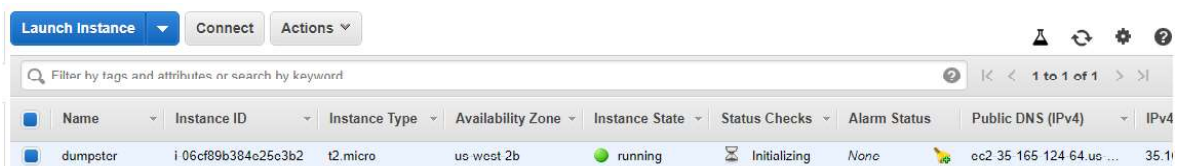


Figura 2.52 Instancia Ec2 creada

Si siguiendo con la configuración de la instancia se debe crear una dirección IP elástica, accediendo en el panel de opciones a *Elastic IPs* y posteriormente en *Allocate new address*. La IP asignada es la 52.24.120.1, la cual será utilizada para el desarrollo del servidor LAMP.

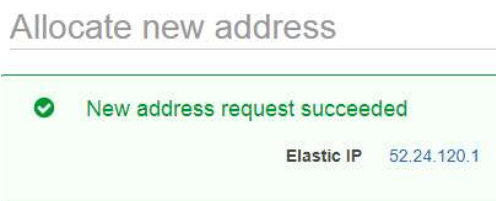


Figura 2.53 Asignación de IP elástica

El siguiente paso es asociar la dirección IP elástica a la instancia creada anteriormente. Para ello en la pestaña *Actions*, seleccionar la opción asociar a una instancia.



Figura 2.54 Asociación de IP elástica a una instancia.

Verificar en la instancia que la IP elástica se asoció, así como también se obtuvo una IP pública y una IP privada.

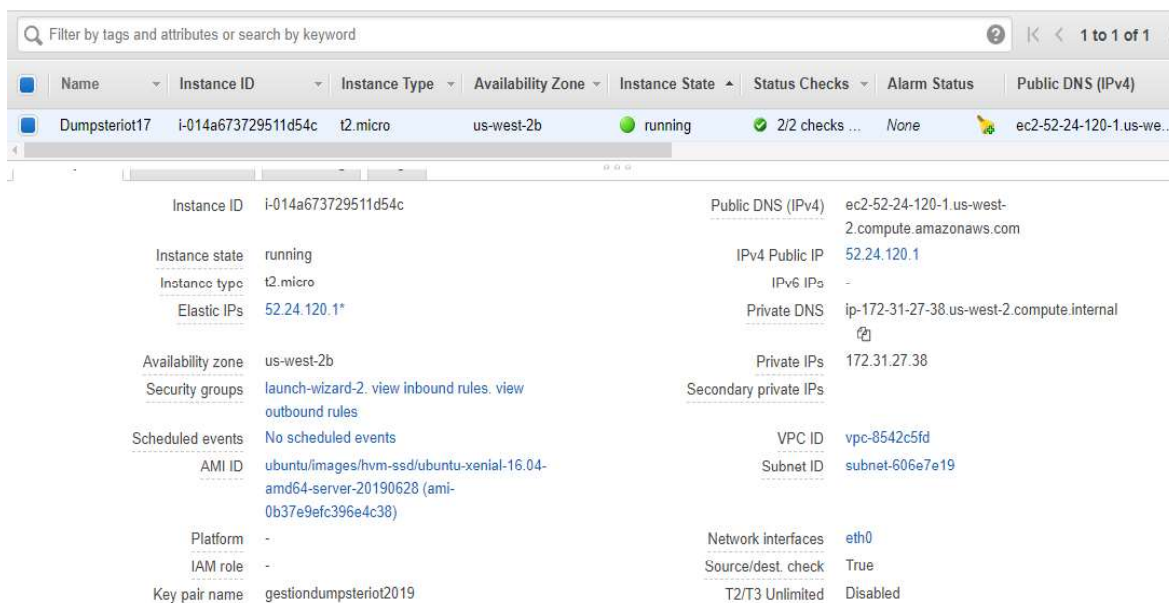


Figura 2.55 Detalles de la Instancia Ec2 creada.

2.3.5.2. Instalación del servidor Apache, MySQL y PHPmyAdmin

Ingresa desde Putty a la dirección IP elástica, configurando previamente la autenticación por SSH con la clave “.ppk” creada.

Primero se instala el servidor apache mediante:

```
Sudo apt-get install apache2
```

Una vez instalado el servidor apache, el directorio raíz de su sitio es “/var/www/html”, es decir, todas las páginas creadas en la Raspberry deberán ser almacenadas en este directorio.

A continuación, se otorgan derechos sobre el archivo apache, se ejecuta los siguientes comandos:

```
sudo chown -R pi: www-data / var / www / html /
sudo chmod -R 770 / var / www / html /
```

Finalizada la instalación, se comprueba que Apache funciona correctamente, para ello se digita la dirección IP de la máquina en cualquier navegador



Figura 2.56 Verificación de instalación de Apache.

A continuación, se instala MySQL con PHP en su última versión.

```
sudo apt-get install php7.0-mysql mysql-server mysql-client
```

Se comprueba que PHP está funcionando correctamente, para ello editar un archivo php, digitar el siguiente comando:

```
sudo nano /var/www/html/phpinfo.php
```

Escribir la siguiente línea `?> php phpinfo(); >?`, posteriormente, desde un navegador digitar la dirección IP de la máquina seguido de `/phpinfo.php` [43].

| PHP versión 7.0.33-0 + deb9u3 | |
|--|---|
| Sistema | Linux raspberrypi 4.14.79+ # 1159 dom. Nov 4 17:28:08 GMT 2018 armv6l |
| La fecha de construcción | 8 de marzo de 2019 10:01:24 |
| API del servidor | Manejador de Apache 2.0 |
| Soporte de directorio virtual | discapacitado |
| Ruta del archivo de configuración (php.ini) | /etc/php/7.0/apache2 |
| Archivo de configuración cargado | /etc/php/7.0/apache2/php.ini |
| Escanea este directorio para archivos adicionales .ini | /etc/php/7.0/apache2/conf.d |
| Archivos .ini adicionales analizados | /etc/php/7.0/apache2/conf.d/10-mysqlnd.ini, /etc/php/7.0/apache2/conf.d/10-opcache.ini, /etc/php/7.0/apache2/conf.d/10-pdo.ini, /etc/php/7.0/apache2/conf.d/15-xsl.ini, /etc/php/7.0/apache2/conf.d/20-bz2.ini, /etc / php / 7.0 / apache2 / conf.d / 20-calendar.ini, /etc/php/7.0/apache2/conf.d/20-ctype.ini, /etc/php/7.0/apache2/conf.d/20-curl.ini, /etc/php/7.0/apache2/conf.d/20-dom.ini, /etc/php/7.0/apache2/conf.d/20-exif.ini, /etc/php/7.0/apache2/conf.d/20-fileinfo.ini, /etc/php/7.0/apache2/conf.d/20-ftp.ini, /etc/php/7.0/apache2/conf.d/20-gd.ini, /etc / php / 7.0 / apache2 / conf.d / 20-gettext.ini, /etc/php/7.0/apache2/conf.d/20-gmconv.ini, /etc/php/7.0/apache2/conf.d/20-gmp.ini, /etc/php/7.0/apache2/conf.d/20-mbstring.ini, /etc/php/7.0/apache2/conf.d/20-mysql.ini, /etc/php/7.0/apache2 / conf.d / 20-pdo_mysql.ini, /etc/php/7.0/apache2/conf.d/20-phar.ini, / etc / php / 7.0 / apache2 / conf.d / 20-posix.ini, /etc/php/7.0/apache2/conf.d/20-readline.ini, /etc/php/7.0/apache2/conf.d/20-shmop.ini, /etc/php/7.0/apache2/conf.d/20-simplexml.ini, /etc/php/7.0/apache2/conf.d/20-sockets.ini, /etc/php/7.0/apache2/conf.d/20-sysvmsg.ini, /etc/php/7.0/apache2/conf.d/20-sysvsem.ini, /etc/php/7.0/apache2/conf.d/20-sysvshm.ini, / etc / php / 7.0 / apache2 / conf.d / 20-tokenizer.ini, /etc/php/7.0/apache2/conf.d/20-wddx.ini, /etc/php/7.0/apache2/conf.d/20-xmlreader.ini, /etc/php/7.0/apache2/conf.d/20-xmlwriter.ini, /etc/php/7.0/apache2/conf.d/20-xsl.ini, /etc/php/7.0/apache2/conf.d/20-zip.ini, / etc / php / 7.0 / apache2 / conf.d / 20-sysvsem.ini, /etc/php/7.0/apache2/conf.d/20-xmlwriter.ini, /etc/php/7.0/apache2/conf.d/20-tokenizer.ini, /etc/php/7.0/apache2/conf.d/20-xmlreader.ini, /etc/php/7.0/apache2/conf.d / 20-xmlwriter.ini, /etc/php/7.0/apache2/conf.d/20-xsl.ini, /etc/php/7.0/apache2/conf.d/20-zip.ini |
| API de PHP | 20151012 |
| Extensión de PHP | 20151012 |
| Extensión Zend | 20151012 |

Figura 2.57 Verificación de instalación de PHP.

Se reinicia el servicio de MySQL con los siguientes comandos:

```
sudo /etc/init.d/mysql restart
```

```
sudo service mysql start
```

Después se instala phpMyAdmin, mediante:

```
sudo apt-get install phpmyadmin
```

En el transcurso de la instalación, en primera instancia se debe escoger la opción apache, posteriormente se debe ingresar una contraseña para trabajar mediante el servidor MySQL.

Por último, se crea un usuario y contraseña (puede ser la misma contraseña ingresada en la instalación de phpmyadmin) para el ingreso mediante phpMyAdmin, para ello se ingresa al root de MySQL, con el siguiente comando:

```
sudo mariadb
```

De esta manera se ingresa a la raíz de MySQL en el cual se debe ingresar las siguientes líneas de comandos:

```
CREATE USER 'admin'@'localhost' IDENTIFIED BY 'controliot17';
```

```
GRANT ALL PRIVILEGES ON . TO 'admin'@'localhost';
```

```
FLUSH PRIVILEGES;
```

```
quit
```

Para obtener una buena conexión con el servidor y phpmyadmin, digitar:

```
sudo nano /etc/apache2/apache2.conf
```

Se muestra un editor del servidor apache en el cual se debe editar la línea final con:

```
Include /etc/phpmyadmin/apache.conf
```

A continuación, se reinicia el servidor [35][43].

```
sudo /etc/init.d/apache2 restart
```

Por último, modificar el archivo de configuración MySQL, para poder acceder de manera remota quien se autentifique con las credenciales de la instancia Ec2. Digitar el comando

```
sudo nano /etc/mysql/mysql.conf.d/mysqld.cnf
```

buscar "bind-address", allí colocar la dirección IP: 0.0.0.0, realizar lo mismo en:

```
sudo nano /etc/mysql/mariadb.conf.d/50-server.cnf
```

Como siguiente paso, acceder a la instancia Ec2, dirigirse a *Security Groups*, crear una regla *Inbound*, con el puerto MySQL/Aurora (3306) y con la dirección 0.0.0.0/0, de esta manera podrá escuchar peticiones.

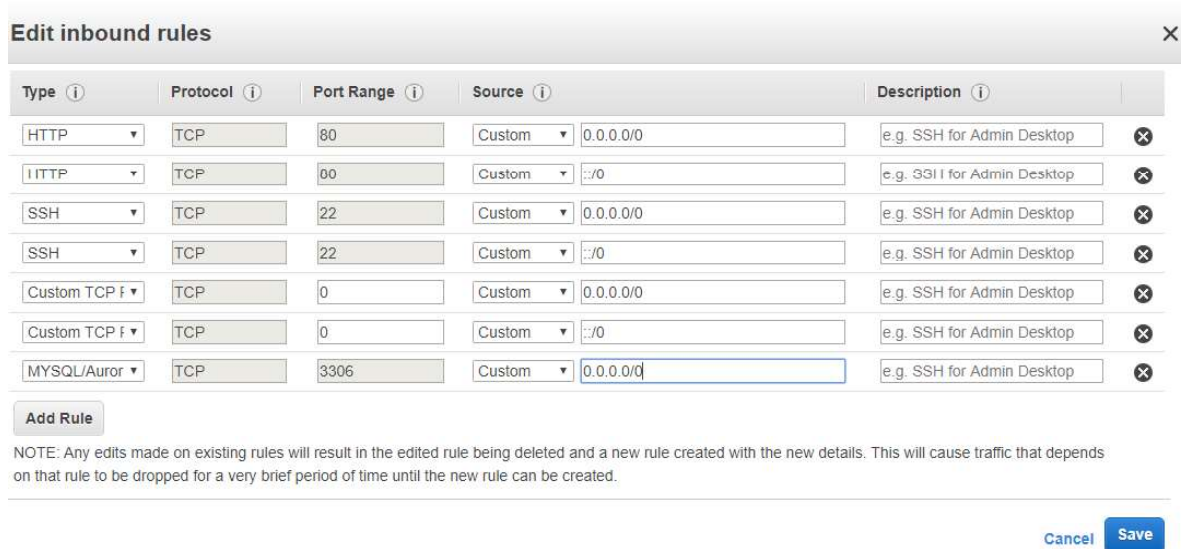


Figura 2.58 Asignación de puerto a MySQL AWS

2.3.5.3. Creación de la base de datos

En una nueva ventana del navegador se debe ingresar la dirección de la máquina seguido de /phpmyadmin (<http://52.24.120.1/phpmyadmin>). La Figura 2.59 muestra la interfaz de bienvenida de phpMyAdmin.



Figura 2.59 Página de bienvenida phpMyAdmin.

Al completar el proceso de autenticación, se ingresará a la página principal donde se procede a crear una nueva base de datos y tabla de valores como se ilustran en las Figuras 2.60 y 2.61 respectivamente.

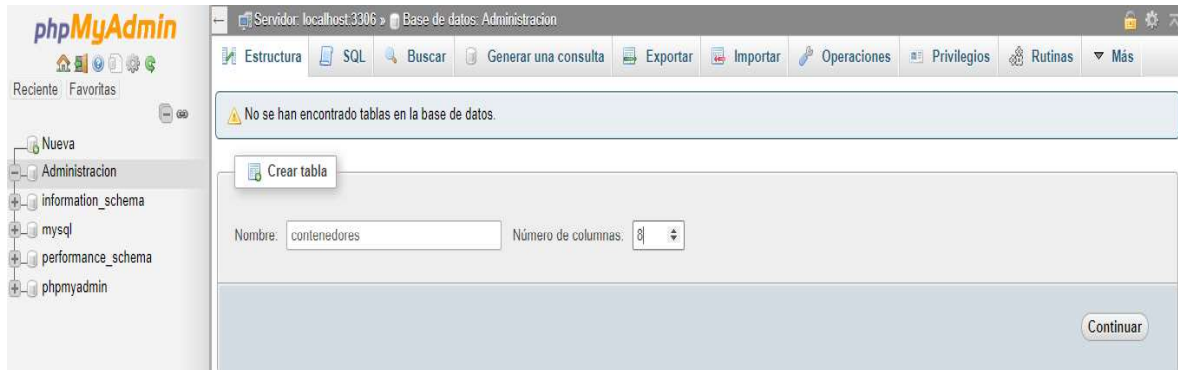


Figura 2.60 Creación de la base de datos.

| # | Nombre | Tipo | Cotejamiento | Atributos | Nulo | Predeterminado | Comentarios | Extra | Acción |
|---|--------------|--------------|--------------------|-----------|------|-------------------|-------------|-------|-------------------------------------|
| 1 | id | int(11) | | | No | Ninguna | | | Cambiar Eliminar Primaria Único Más |
| 2 | Fecha y Hora | timestamp | | | No | CURRENT_TIMESTAMP | | | Cambiar Eliminar Primaria Único Más |
| 3 | Ubicación | double | | | No | Ninguna | | | Cambiar Eliminar Primaria Único Más |
| 4 | Distancia | double | | | No | Ninguna | | | Cambiar Eliminar Primaria Único Más |
| 5 | Cantidad | double | | | No | Ninguna | | | Cambiar Eliminar Primaria Único Más |
| 6 | Vacio | varchar(255) | utf8mb4_general_ci | | No | Ninguna | | | Cambiar Eliminar Primaria Único Más |
| 7 | Medio | varchar(255) | utf8mb4_general_ci | | No | Ninguna | | | Cambiar Eliminar Primaria Único Más |
| 8 | Lleno | varchar(255) | utf8mb4_general_ci | | No | Ninguna | | | Cambiar Eliminar Primaria Único Más |

Figura 2.61 Tablas de la base de datos.

Como previamente se indicó base de datos “administración” contiene una tabla de valores conformada por 8 columnas, aquí se configurará los tipos de datos a almacenar en cada una de las columnas: El campo id será configurado como primario y con la opción de auto-incremento, servirá para identificación de cada valor subido a la base de datos, la Fecha y Hora es de tipo *TIMESTAMP* y *CURRENT_TIMESTAMP* como predeterminado para obtener la hora automáticamente cada vez que se actualice un valor, los campos Latitud, Longitud, Distancia son del tipo *FLOAT* ya que almacenarán valores con decimales, mientras que el campo lleno será *INT* ya que solamente indicará una señal de alerta [39].

2.3.5.4. Prueba de conexión de la Raspberry a la base de datos

Finalizada la base de datos con su correspondiente tabla y campos, se debe realizar un programa que permita obtener una conexión y envío de información con la base de datos.

Antes que nada, se debe instalar la librería que hará de intermediaria entre MySQL y Python, con el comando:

```
sudo apt-get install python-mysqldb
```

En el programa Python, se creará la función *query*, propia del lenguaje SQL que permitirá ejecutar de manera correcta la conexión a la base de datos. Por lo tanto, se debe seguir un orden:

Primero, abrir la conexión y crear un puntero.

`connect ()`: crea una conexión a la base de datos, utiliza cuatro argumentos: [servidor, usuario, contraseña, base de datos, tabla]

Segundo, ejecutar la consulta.

`cursor ()`: crea un cursor, capaz de ejecutar consultas SQL dentro de la base de datos.

`execute ()`: ejecuta consultas SQL en la base de datos. Toma una consulta SQL (como cadena) como un argumento.

Tercero, hacer efectiva la escritura.

`commit()`: en el caso que la base de datos al igual que las tablas admitan transacciones, esto confirma la transacción actual; De lo contrario, este método no hace nada con éxito.

Cuarto, cerrar el Puntero y la conexión.

`close ()`: cierra la conexión de la base de datos.

Quinto, envío de datos

Una vez finalizado el programa se ejecuta y en la base de datos pulsar en la opción estructura para ver la información obtenida [39].

| | | | id | Fecha y hora | Latitud | Longitud | Distancia |
|--|--|--|----|---------------------|----------|----------|-----------|
|  Editar |  Copiar |  Borrar | 1 | 2019-04-25 19:18:37 | -1.28137 | -78.6143 | 20.0314 |
|  Editar |  Copiar |  Borrar | 2 | 2019-04-25 19:18:57 | -1.28137 | -78.6143 | 19.4304 |
|  Editar |  Copiar |  Borrar | 3 | 2019-04-25 19:19:17 | -1.28137 | -78.6143 | 19.737 |
|  Editar |  Copiar |  Borrar | 4 | 2019-04-25 19:19:37 | -1.28137 | -78.6143 | 19.876 |
|  Editar |  Copiar |  Borrar | 5 | 2019-04-25 19:19:58 | -1.28137 | -78.6143 | 6.87749 |
|  Editar |  Copiar |  Borrar | 6 | 2019-04-25 19:29:08 | -1.28137 | -78.6143 | 33.5656 |
|  Editar |  Copiar |  Borrar | 7 | 2019-04-25 19:29:28 | -1.28137 | -78.6143 | 33.5452 |
|  Editar |  Copiar |  Borrar | 8 | 2019-04-25 19:29:48 | -1.28137 | -78.6143 | 33.9908 |
|  Editar |  Copiar |  Borrar | 9 | 2019-04-25 19:30:09 | -1.28137 | -78.6143 | 33.5124 |
|  Editar |  Copiar |  Borrar | 10 | 2019-04-25 19:30:29 | -1.28137 | -78.6143 | 34.0972 |
|  Editar |  Copiar |  Borrar | 11 | 2019-04-25 19:30:50 | -1.28137 | -78.6143 | 19.827 |
|  Editar |  Copiar |  Borrar | 12 | 2019-04-25 19:31:10 | -1.28137 | -78.6143 | 19.9987 |
|  Editar |  Copiar |  Borrar | 13 | 2019-04-25 19:31:31 | -1.28137 | -78.6143 | 19.7207 |
|  Editar |  Copiar |  Borrar | 14 | 2019-04-25 19:31:52 | -1.28137 | -78.6143 | 19.8597 |
|  Editar |  Copiar |  Borrar | 15 | 2019-04-25 19:32:12 | -1.28137 | -78.6143 | 19.7738 |

Figura 2.62 Registro de datos en las tablas.

2.3.6. Creación del Sitio WEB

Para la creación del sitio web se utilizarán los lenguajes de programación HTML (sitios estáticos), PHP (sitios dinámicos) y CSS para crear diferentes estilos visuales en la página web.

Un sitio estático se utiliza para mostrar una información permanente y sin cambio alguno en su interfaz, mientras que un sitio dinámico contiene aplicaciones dentro de la página web, siendo más interactivo con el navegante, mostrará información actualizada a medida que se actualicen los datos obtenidos, incluso permite ingresar parámetros tal sea el caso de encuestas, envío de e-mails, votaciones, entre otros [33].

El proyecto contará con tres páginas web, una estática llamada "index.html" que será la presentación y dos dinámicas, una página llamada "monitoreo.php" en la cual se mostrarán las tablas de valores de la cantidad de desechos del contenedor de basura y otra llamada

“ubicación.php” que mostrará la geolocalización de dicho contenedor, la información será obtenida desde la base de datos.

De manera complementaria se utiliza el programa PIXLR EDITOR, el cual permite la edición de imágenes de manera sencilla, si bien no está al nivel de otros programas como Photoshop, cuenta con características de uso amigable al usuario, que lo hace un programa excelente para aquellos que no cuentan con los conocimientos de diseño gráfico, hay que mencionar que es un programa gratuito y no necesita ser instalado pues se puede acceder de manera online en <https://pixlr.com/editor/> [44].

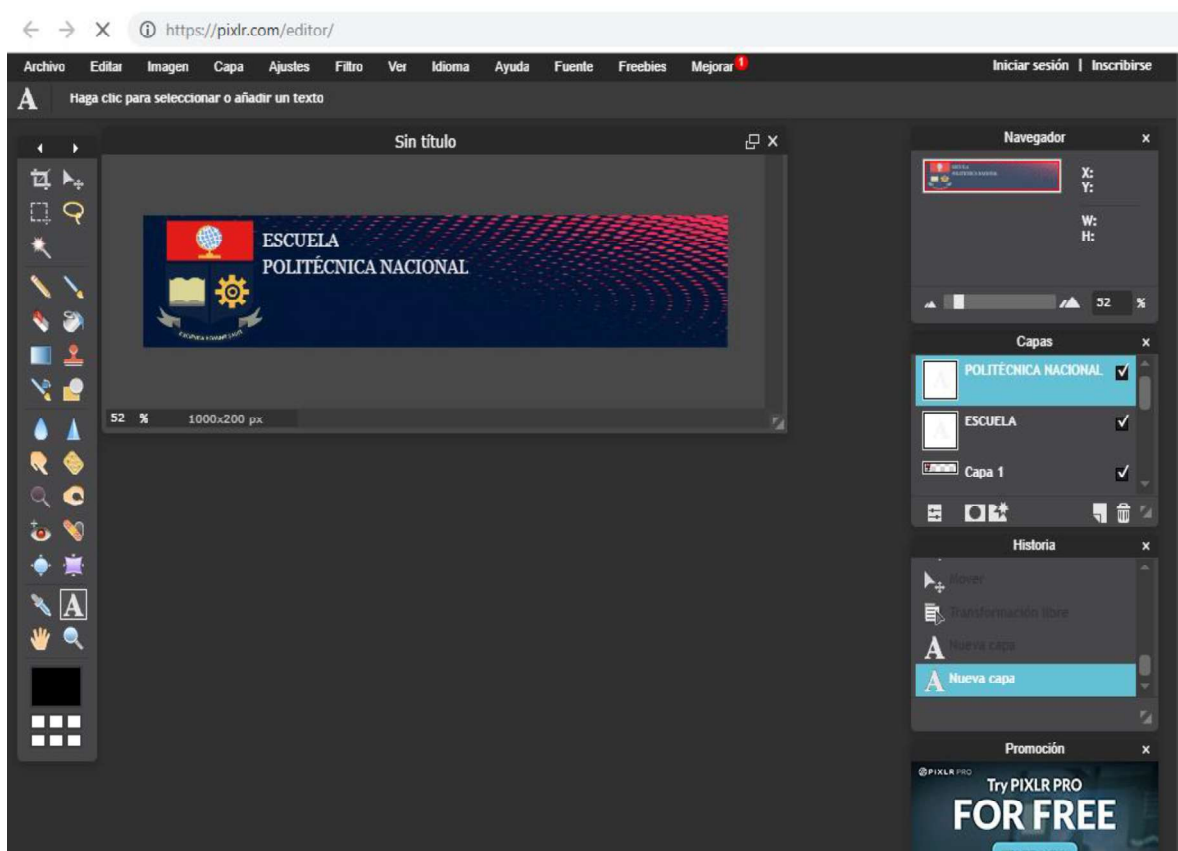


Figura 2.63 Interfaz del programa PIXLR.

Todas las imágenes elaboradas y que serán utilizadas en la creación de las páginas web también deben guardarse dentro del directorio del servidor apache.

También es necesario crear un archivo editable “.css”, en el cual se desarrollarán los estilos que se aplicarán en la página web como: tipo, tamaño y color de letra, creación de botones, animaciones, imágenes de fondo, color de fondo, dimensiones, etc.


```
index.html # estilos.css x
1 #header {
2   position: absolute;
3   width: 1366px; height: 200px;
4   z-index: 1;
5   left: 0px; top: 0px;
6   background-image: url(imagenes/header.jpg);
7 }
8 #main {
9   position: absolute;
10  width: 250px; height: 457px;
11  z-index: 1;
12  left: 0px; top: 200px;
13  background-color: #000033;
14 }
15 #body {
16  position: absolute;
17  width: 1116px; height: 457px;
18  z-index: 1;
19  left: 250px; top: 200px;
20 }
21 #img {
22  position: absolute;
23  top: 90px;
24  left: 50px;
25 }
26 .inicio {
27  display: block;
28  position: absolute;
29  width: 180px; height: 50px;
30  z-index: 2;
31  left: 35px; top: 40px;
32  background-color: #ff0000;
33  border-radius: .5em;
34  cursor: pointer;
```

Figura 2.64 Líneas de código css de estilos.

Para comenzar el desarrollo de la página web primero se debe acceder al directorio de Apache2 (/var/www/html), y crear un archivo editable “.html” que permitirá hacer el diseño de la página estática.

Dentro del archivo se crea la estructura básica de html, conformada por tres contenedores:

- <html>: Contiene toda la página web.
- <head>: Es el encabezado del documento, contiene información opcional como el título, estilos, tipo de texto, etc.
- <body>: Contiene todo aquello que aparece en la página web mostrada en el navegador.

Las tres páginas web se dividirán en tres secciones mediante la opción div, con los identificadores header (parte superior), main (lado izquierdo) y body (lado derecho).



Figura 2.65 Márgenes de las secciones de páginas web.

En la sección *header* se introducirá una imagen de fondo como carátula, esto se realiza en el archivo “.css” con la línea de comando *background-image* junto con el directorio de la imagen.

La sección *main* contendrá tres botones: inicio, monitoreo y ubicación, los cuales contarán con una pequeña animación y que al hacer clic sobre cualquiera de ellos, se enviará a la página web correspondiente, es decir, al hacer clic sobre el botón inicio se accederá a la página web estática index.html de presentación, con el botón monitoreo se redireccionará a la página web dinámica monitoreo.php que contendrá una tabla de valores de las mediciones del contenedor de basura obtenidos desde la base de datos y con el botón ubicación se mostrará la página web ubicación.php que contendrá la información de geolocalización (latitud y longitud) del contenedor de basura en un mapa global.

Esta sección también contará con una parte de contactos en el cual se mostrarán los correos electrónicos de las personas afines al proyecto, así como un enlace a una página de Facebook.

Para el desarrollo de las tres páginas web el funcionamiento de las dos secciones será completamente igual.

```

10 <div id="main">
11   <div class="inicio"><p> INICIO </p></div>
12   <div class="monitoreo"><p> MONITOREO </p></div>
13   <div class="ubicacion"><p> UBICACION </p></div>
14   <div class="contactos">CONTACTOS:</div>
15   <div class="correo"> <u>edisson.acosta@epn.edu.ec<br>ricardo.llugsi@epn.edu.ec </u></div>
16   <a target="blank" href="https://www.facebook.com/Grupo-de-investigaci%C3%B3n-Internet-de-todas-las-Cosas-187955935211483">
17     </a>

```

Figura 2.66 Líneas de código de la sección header.

La sección *body* de la página web *index.html* estará constituida por una imagen del esquema e información del proyecto en desarrollo.

Para introducir la imagen se utiliza la etiqueta `` un identificador `id` y el atributo `src` que contendrá el directorio de la imagen.

Para agregar texto se crea una división (`div`) tipo `class` que será el entorno dentro del cual entre las etiquetas `<p>` (inicia) `</p>` (termina) se introduce el texto deseado.

Dentro de estas etiquetas se recomienda el uso de la etiqueta `
` para el cambio de línea de texto y el comando `align="justify"` para tener el texto debidamente alineado.

```
19 <div id="body">
20 
22 <b>MONITOREO DE CONTENEDORES DE BASURA PARA CIUDADES INTELIGENTES</b><br><br><br>
23 La contaminación ambiental, pérdidas de dinero, problemas de salubridad y problemas estéticos de una ciudad debido a la aglomerac
24 Este proyecto busca adquirir y transmitir inalámbricamente datos de los niveles de desechos de un contenedor.<br><br>
25 Las pruebas se realizarán en un ambiente controlado, y la cantidad de desechos se medirán en tres niveles: vacío, medio y lleno m
26 Los datos de la cantidad de desechos así como su geolocalización serán procesados por una tarjeta de procesamiento,
27 la cual procesará y enviará automáticamente a través de la red celular cada cierto tiempo la información procesada para poder su
28 El prototipo contará con una alimentación autónoma conformada por paneles solares, baterías.<br><br>
29 Se buscará proteger el prototipo contra elementos que sean dañinos para su vida útil como las altas temperaturas, agua, polvo, su
30 </p>
31 </div>
32 </div>
```

Figura 2.67 Líneas de código de la sección *body*.

La sección *body* de la página web *monitoreo.php* estará constituida por una tabla de ocho columnas, para crear dicha tabla se utiliza un `div` tipo `class` junto con la etiqueta `<table>` `</table>`.

Dentro de estas etiquetas se colocan los nombres de las columnas de igual manera que se encuentran en la base de datos creada previamente, para sincronizar los valores que se vayan obteniendo.

Después entre etiquetas `php` se realiza la conexión con la base de datos, y dentro de un condicionado `"while"` se llamará a cada una de las columnas identificadas en la base de datos, deben ser llamadas en orden exacto que las columnas creadas anteriormente [45].

```

<div class="container">
  <h2> Tabla de valores del contenedor de basura IOT </h2>
  <table border="1">
    <tr>
      <td>id</td>
      <td>Fecha_Hora</td>
      <td>Latitud</td>
      <td>Longitud</td>
      <td>Distancia</td>
      <td>Vacio</td>
      <td>Medio</td>
      <td>LLeno</td>
    </tr>

<?php
$sql="SELECT * from contenedores";
$result=mysqli_query($conexion,$sql);

while($mostrar=mysqli_fetch_array($result)){
  >
  <tr>
    <td><?php echo $mostrar['id'] ?></td>
    <td><?php echo $mostrar['Fecha y Hora'] ?></td>
    <td><?php echo $mostrar['Latitud'] ?></td>
    <td><?php echo $mostrar['Longitud'] ?></td>
    <td><?php echo $mostrar['Distancia'] ?></td>
    <td><?php echo $mostrar['Vacio'] ?></td>
    <td><?php echo $mostrar['Medio'] ?></td>
    <td><?php echo $mostrar['LLeno'] ?></td>
  </tr>
<?php

```

Figura 2.68 Líneas de código, creación de tablas en la sección body.

La sección *body* de la página web ubicacion.php estará constituida por una tabla de dos columnas que mostrará únicamente la posición actual en longitud como latitud.

Además, se utilizará la aplicación de Google Maps para mostrar un mapa global, para ello en la sección header se introduce la siguiente línea [46].

```

<script async defer src="https://maps.googleapis.com/maps/api/js?key=AlzaSyCO-
_SfJnmmKtXB7ti2fWU0uMlbFe9902Y" type="text/javascript"></script>

```

```

<div id="body">
  <div class="container">
    <table class="table-elements">
      <tr>
        <td><input type="button" value="Obtener mi ubicacion - A" onclick="get_my_location();" class="btn"></td>
        <td><input type="text" placeholder="Latitud" id="my_lat" class="txt" readonly></td>
        <td><input type="text" placeholder="Longitud" id="my_lng" class="txt" readonly></td>
        <td><select class="txt" onchange="draw_rute(this.value)">
          <option value="0">Dibujar ruta con &#8595;</option>
          <?=$google->get_stores();?>
        </select>
        </td>
      </tr>
    </table>
    <div class="map" id="map"></div>
  </div>
  <script type="text/javascript">
    start_map();
  </script>
</div>

```

Figura 2.69 Líneas de código, tablas e interfaz del mapa de ubicación.

Para que el mapa funcione correctamente se debe obtener una clave API (introducida en la misma línea de comando en la parte "key"); primero en el navegador de búsqueda se ingresa a console.developers.google, después se selecciona la opción credenciales, nuevo proyecto, crear nueva clave (seleccionar clave API) y a continuación se muestra la clave API propia.

Para obtener dicha clave es necesario realizarla desde una cuenta de Google (Gmail), sin la clave API la aplicación de Google Maps no funcionará.

Por otra parte, esta aplicación ya no es gratuita desde el año 2016 y solo permite una visita por día, para ampliar esta limitación se debe crear una cuenta para facturación de Google, los precios van acorde a cuantas visitas por día requerirá la página web.

Al problema anterior se suma la incapacidad de acceder a la ubicación actual, esto debido a que Google restringió la obtención de ubicación en sitios no seguros (http), por lo que es necesario pagar un dominio, hosting y cambiar el sitio web de inseguro (http) a seguro (https).

Una vez creados los códigos fuente que contienen las páginas web, en el navegador digitar la dirección IP de la Raspberry seguido del nombre de las páginas, por ejemplo: 52.24.120.1/monitoreo.php.

ESCUELA POLITÉCNICA NACIONAL

MONITOREO INALÁMBRICO DE LA CANTIDAD DE DESECHOS DE UN CONTENEDOR DE BASURA IOT

INICIO

MONITOREO

UBICACION

CONTACTOS:
edison.acosta@epn.edu.ec
ricardo.llagui@epn.edu.ec

MONITOREO DE CONTENEDORES DE BASURA PARA CIUDADES INTELIGENTES

La contaminación ambiental, pérdidas de dinero, problemas de salubridad y problemas estéticos de una ciudad debido a la aglomeración y desbordamiento de basura de los contenedores impulsan a buscar soluciones inteligentes y de bajo costo.

Este proyecto busca adquirir y transmitir inalámbricamente datos de los niveles de desechos de un contenedor.

Las pruebas se realizarán en un ambiente controlado, y la cantidad de desechos se medirán en tres niveles: vacío, medio y lleno mediante un sensor ultrasónico.

Los datos de la cantidad de desechos así como su geolocalización serán procesados por una tarjeta de procesamiento, la cual procesará y enviará automáticamente a través de la red celular cada cierto tiempo la información procesada para poder su almacenamiento en un servidor y su posterior visualización en la página web.

El prototipo contará con una alimentación autónoma conformada por paneles solares, baterías.

Se buscará proteger el prototipo contra elementos que sean dañinos para su vida útil como las altas temperaturas, agua, polvo, sustancias químicas, entre otras.

Figura 2.70 Página web dumpsteriot.html

| id | Fecha_Hora | Latitud | Longitud | Distancia | Vacio | Medio | LLeno |
|----|------------|----------|----------|-----------|-------|-------|-------|
| 63 | | 0 | 0 | 0 | | | |
| 64 | | -1.28137 | -78.6143 | 23.6828 | 0 | 0 | |
| 65 | | -1.28137 | -78.6143 | 23.6133 | 0 | 0 | |
| 66 | | -1.28137 | -78.6143 | 23.5806 | 0 | 0 | |
| 67 | | 0 | 0 | 0 | | | |
| 68 | | 0 | 0 | 0 | | | |
| 69 | | 0 | 0 | 0 | | | |
| 70 | | 0 | 0 | 0 | | | |

Figura 2.71 Página web monitoreo.php

Figura 2.72 Página web ubicacion.php.

2.3.7. Conexión del Sistema Fotovoltaico

Una vez definidos los elementos con las características técnicas adecuadas para el proyecto se realiza la implementación del sistema fotovoltaico, tomando el controlador de carga solar como punto central para realizar las conexiones.

Instrucción de seguridad:

- Verificar que la batería tenga suficiente voltaje antes de la primera instalación.
- La conexión controlador-batería debe ser lo más cercana posible para minimizar las pérdidas.
- El controlador no es adecuado para baterías de hidruro de metal de níquel, litio.
- El controlador de carga sólo es adecuado para la regulación de paneles solares.
- Nunca conecte otra fuente de carga al regulador de carga.

Conexión

- Primero, conectar la batería teniendo en cuenta los terminales positivo y negativo.
- Segundo, conectar el panel solar a los pines correspondientes a este, tanto positivo como negativo.
- Tercero, conectar la Raspberry pi al conector USB.

2.3.8. Conexión final

El prototipo final para el proyecto desarrollado con las conexiones de cables adecuados para todos los elementos usados tanto para el sistema de alimentación fotovoltaica como para el circuito principal se muestran a continuación en los siguientes diagramas, siendo el color rojo, el terminal positivo y el color negro, el terminal negativo.

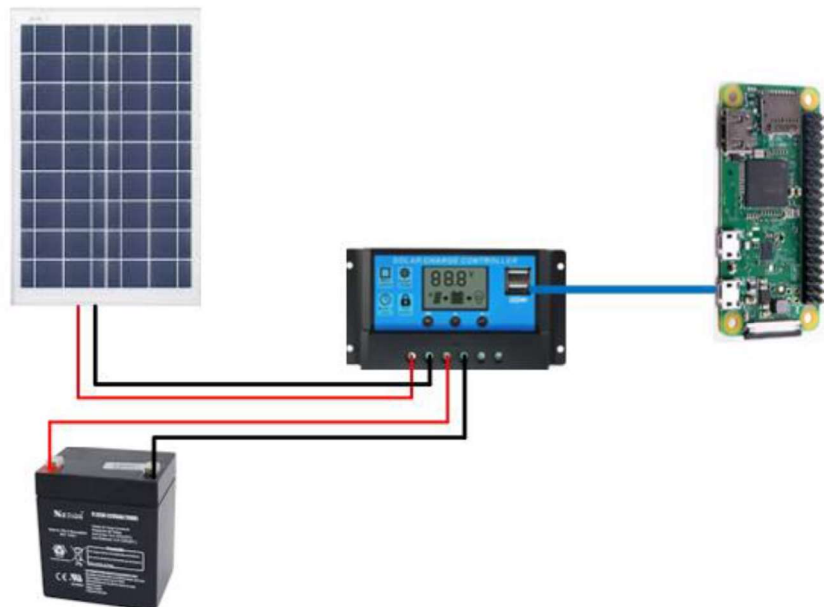


Figura 2.73 Sistema de alimentación autosustentable creada en Visio.

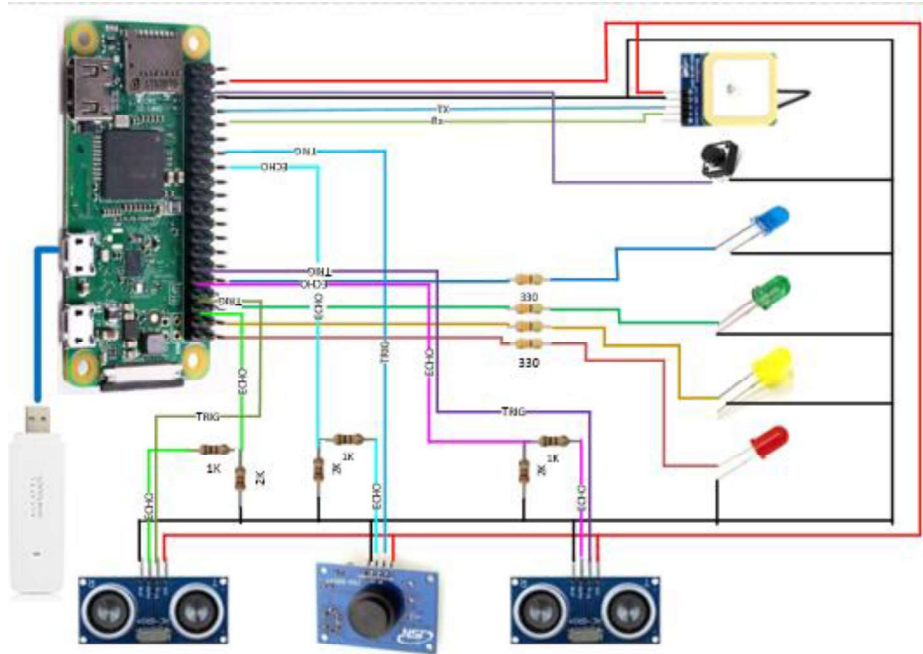


Figura 2.74 Circuito prototipo final creada en Visio.

2.3.9. Configuraciones adicionales

2.3.9.1. Programa on/off

Para la configuración de un pulsador como botón de encendido y apagado de la Raspberry se conecta a los pines GPIO cinco y seis, de esta sencilla manera y sin ninguna línea de código se puede poner en marcha la tarjeta de procesamiento.

Sin embargo, para el proyecto se necesita poder tanto encender como apagar la Raspberry, con la finalidad de reducir el consumo en situaciones que no se encuentre en funcionamiento.

Se utilizará el método *retropie*, primero se actualiza el sistema e instala los permisos de pines GPIO con los comandos:

```
“sudo apt-get update
sudo apt-get -y install python-rpi.gpio” [47].
```

A continuación, crear un script de Python en el directorio local.

```
sudo nano shutdown.py
```

dentro del archivo de texto creado se digitan las líneas de código respectivas.

```
“import RPi.GPIO as GPIO
import time
import os
GPIO.setmode(GPIO.BCM)
```



```

GPIO.setup(03, GPIO.IN, pull_up_down = GPIO.PUD_UP)
    while True:
        print GPIO.input(03)
        if(GPIO.input(03) == False):
            os.system("sudo shutdown -h now")
            break
        time.sleep(1)" [47].

```

Al finalizar guardar los cambios y salir (Ctrl+O y Ctrl+X, respectivamente).

Ejecutar el script de Python de manera manual, con el comando:

```
sudo python shutdown.py
```

Si el código es correcto no se mostrará ningún error en pantalla y al pulsar el botón la Raspberry se debe apagar y de igual forma el led, de esta manera al pulsar nuevamente el botón la Raspberry y el led encenderán [47].

2.3.9.2. Envío de correos electrónicos

El prototipo estará en la capacidad de enviar notificaciones de alerta mediante correo electrónico, para ello se trabajará con una cuenta de Gmail.

El programa utilizará el protocolo SMTP (*Simple Mail Transfer Protocol*, Protocolo para Transferencia Simple de Correo) que permite la transferencia de correos electrónicos en Internet.

Primero instalar las librerías ssmtp y mutt, mediante los siguientes comandos

```

"sudo apt-get install ssmtp
sudo apt-get install mutt" [42].

```

A continuación, modificar el archivo de configuración "ssmtp.conf".

```
sudo nano /etc/ssmtp/ssmtp.conf
```

Añadir su cuenta y contraseña de Gmail

```

mailhub=smtp.gmail.com:587
AuthUser=correo@gmail.com AuthPass=password
FromLineOverride=YES
UseTLS=YES
UseSTARTTLS=YES

```

Figura 2.75 configuración de credenciales de una cuenta Gmail

Asignar permisos al archivo de configuración:

```
sudo chmod 774 /etc/ssmtp/ssmtp.conf
```

Reiniciar la Raspberry.

Para finalizar se debe habilitar el Acceso de app menos seguras ingresando al enlace

<https://myaccount.google.com/lesssecureapps?pli=1>.

Ahora se realizará una prueba de envío, para verificar que no el proceso realizado ha sido correcto, mediante el siguiente comando:

```
sudo echo "Prueba envío de correos desde RPi" | mutt -s "Prueba 1" edy_acosta11@hotmail.com
```

Si todo está correcto, recibirá el correo electrónico al correo de destino.

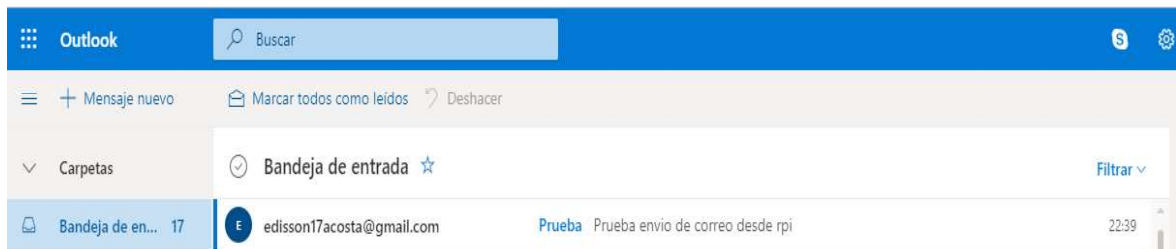


Figura 2.76 Bandeja de entrada del correo de destino.

2.3.9.3. Automatización

La Raspberry pi debe ejecutar los programas de Python realizados siempre cuando arranque el sistema, para ello se digita el siguiente comando:

```
sudo crontab -e
```

En caso de que aparezca una pregunta, seleccionar la opción 2 (nano).

Dentro del texto que aparece, hay que dirigirse al final y digitar:

```
@reboot sudo python /directorio/programa.py
```

Guardar los cambios y salir.

Por otro lado se debe automatizar la conexión del modem USB 3G cada vez que se reinicie la Raspberry, para ello se edita al archivo "rc.local" mediante "sudo nano /etc/rc.local" y se añade la siguiente línea de comando antes de "exit 0".

```
"sudo /home/pi/umtskeeper/umtskeeper—sakisoperators "USBINTERFACE='0' OTHER='USBMODEM' USBMODEM='12d1:1506' APN='ba.amx' APN_USER='0' APN_PASS='0'"—sakisswitches "—sudo—console"—devicename 'Huawei'—log—silent—monthstart 8 --nat 'no'—httpserver &>> /home/pi/umtskeeper/error.log&" [42].
```

Este comando me permitirá que se conecte de manera automática cada vez que se encienda o reinicie la tarjeta.

3. PRUEBAS Y RESULTADOS

Esta sección muestra las pruebas de funcionamiento y los resultados conseguidos en las diferentes etapas: envío de datos a la base MySQL, conexión con las páginas web, envío de mensaje de alarma y optimización de los recursos energéticos del sistema.

Las pruebas se realizarán en diferentes lugares, con diferente número de sensores.

3.1. Etapa de construcción

El prototipo desarrollado, se probará en una maqueta de tamaño y forma similar a un contenedor de basura de la ciudad de Quito.



Figura 3.1 Vista frontal del contenedor



Figura 3.2 Vista lateral del contenedor



Figura 3.3 Vista posterior del contenedor

Para la protección del circuito principal, así como su sistema de alimentación una caja metálica de la marca BeauCoup con la normativa INEN 2568 y grado de protección IP20.



Figura 3.4 Caja BeauCoup



Figura 3.5 Vista interior de la caja de protección.

Los sensores ultrasónicos se encuentran protegidos por cajas impresas en 3D de material acrílico.



Figura 3.6 Protección de sensores ultrasónicos

3.2. Funcionamiento general.

Una vez acoplado todo el sistema prototipo, se proceden a realizar pruebas de llenado en el contenedor con diversos materiales.

El funcionamiento se realizará mediante seis programas scripts que inician de manera automática.

Los tres programas de distancia (uno por cada sensor) tomarán datos cada 7 minutos (tiempo en que se tarde en arrancar el sistema) y enviarán su última medición hacia archivos de texto localizados en el mismo directorio y que posteriormente serán utilizados por el programa principal.

Dos programas de ubicación y se utilizarán para realizar mediciones con cinco minutos de diferencia, esto con la finalidad de comparar si el contenedor ha cambiado de localización cuando sea vaciado. De la misma manera enviarán sus datos hacia archivos de texto.

El programa principal lee los archivos de texto de los programas de distancia y ubicación para posteriormente realizar comparaciones, realiza la comunicación con MySQL y la base de datos creada, teniendo tres opciones de envío de información.

- Estado vacío: el dato de la distancia es superior a los 111 cm, se enciende la luz led amarilla, se envía valor de la distancia a la tabla de datos correspondiente, primera ubicación (latitud y longitud), y un valor de 1 a la casilla con el nombre vacío.
- Estado medio: el dato de la distancia está comprendido entre 75 a 111 cm, se enciende la luz led azul, se envía valor de la distancia a la tabla de datos correspondiente, última ubicación (latitud y longitud), y un valor de 1 a la casilla con el nombre medio.

- Estado lleno: el dato de la distancia es inferior a los 75 cm, se enciende la luz led roja, se envía valor de la distancia a la tabla de datos correspondiente, última ubicación (latitud y longitud), y un valor de 1 a la casilla con el nombre lleno.

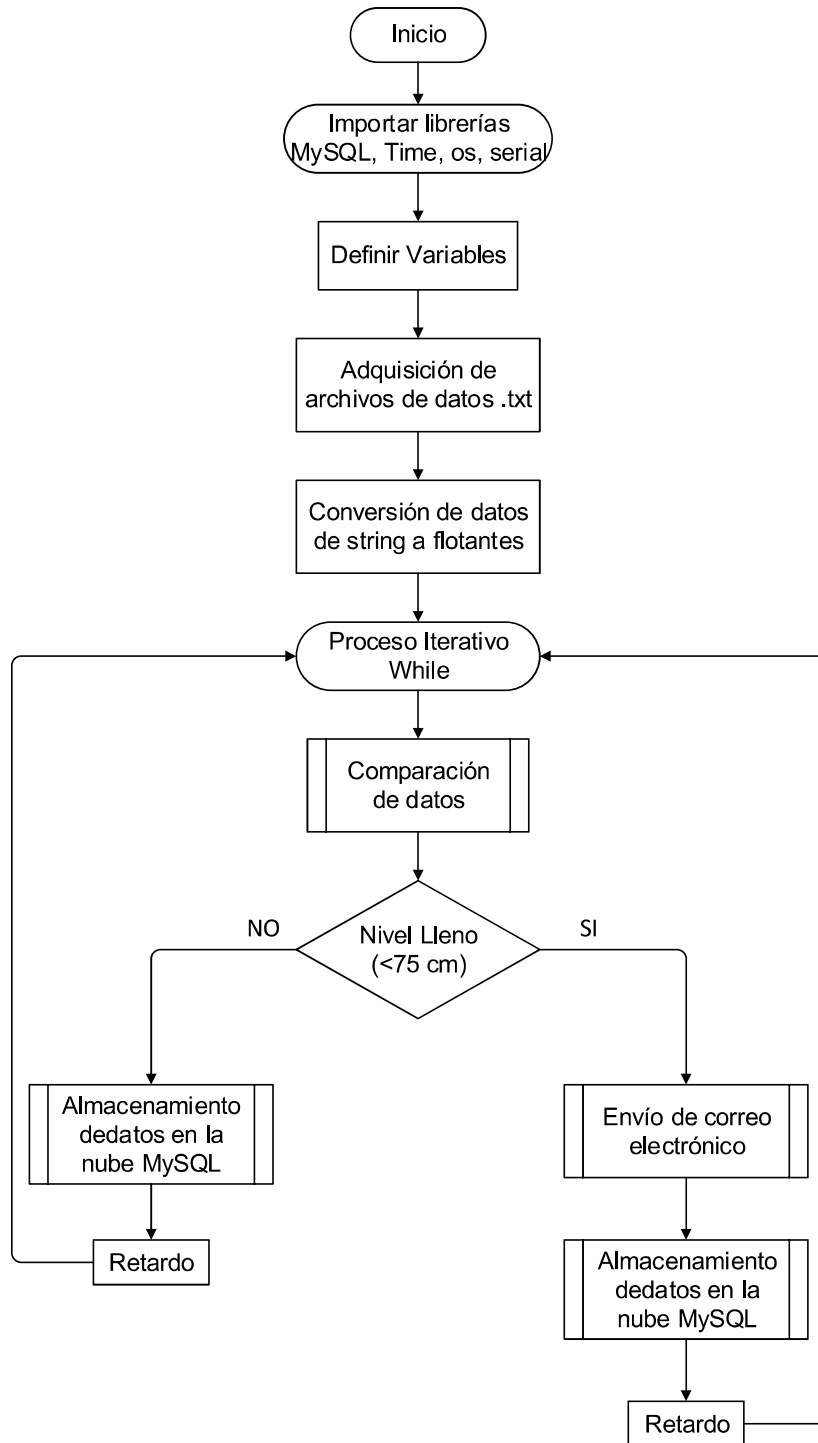


Figura 3.7 Diagrama de flujo del programa principal

En segundo plano, el programa “shutdown” se encuentra ejecutando normalmente hasta que se pulse el botón “on/off” para producir su funcionamiento. Así cuando el circuito se encuentra encendido, al momento de pulsar el botón alrededor de 0 a 5 seg, apagará el sistema y al pulsar nuevamente el botón encenderá una luz led verde que indicará que el sistema ha arrancado.

El sistema fotovoltaico se mantiene funcionando permanentemente, en días de sol el controlador de carga utilizará la energía obtenida desde el panel solar para alimentar el prototipo manteniendo intacta la carga almacenada en la batería.

En situaciones de días nublados o con presencia de lluvia, así como en las noches, el controlador de carga solar utilizará la carga almacenada en la batería para alimentar al prototipo ya que para estos casos la energía obtenida por el panel solar es mínima o nula, ya que este elemento debe recibir rayos solares de manera directa.

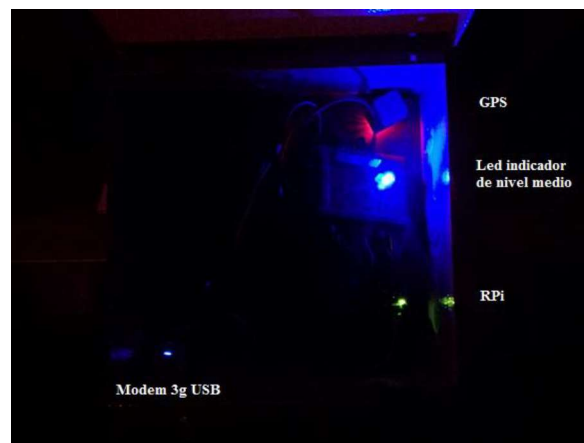


Figura 3.8 Prueba del sistema de alimentación en la noche.

En la figura anterior muestra la prueba visual del funcionamiento del prototipo en la noche, donde, la luz roja representa al módulo GPS, la luz azul indica el nivel medio de distancia, la luz celeste es el modem USB 3G y la luz verde a la Raspberry.

Al conectarse directamente a la Raspberry se puede observar el funcionamiento en el modo consola ejecutando el programa principal.

```
pi@raspberrypi:~/Documents/tesis $ sudo python tesisc.py
posicion nueva:
-1.28137 -78.61435
distancia:
119.767 cm
```

Figura 3.9 Prueba de funcionamiento en el modo consola.

Para tener una mejor apreciación del funcionamiento del prototipo se realiza un análisis de prueba y error de los sensores de distancia y del módulo GPS.

Tabla 3.1 Prueba del módulo GPS

| Ubicación Real (Google Maps) | Ubicación mediante módulo GPS | Error |
|--|--------------------------------------|---------------------------|
| Domicilio Ambato= -1.28111,- 78.61438 | -1.28136, -78.6143 | Lat:0.025% / Long: 0% |
| EPN Quito= -0.20929,-78.48926 | -0.20925, -78.4894 | Lat:0.004% / Long: 0.014% |

El módulo GPS tiene una alta precisión en la adquisición de datos de geolocalización, variará dependiendo de cuan obstruido se encuentre. Visite la pág. web 52.24.120.1/ubicación.php,

La siguiente tabla muestra una comparación del funcionamiento de los sensores de distancia.




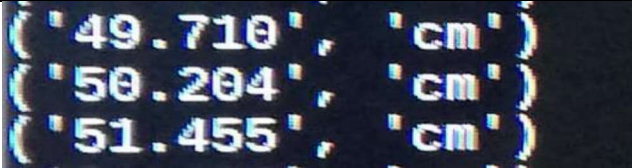
Tabla 3.2 Prueba de los sensores de distancia

| Distancia real en cm | Distancia en cm Sensor JSN-SR04T | Error | Distancia en cm Sensor SC-HR04T | Error |
|-----------------------------|---|--------------|--|--------------|
| 85 | 85.098 | 9.8% | 84.930 | 7% |
| | 85.151 | 15.1% | 84.824 | 17.6% |
| | 84.533. | 46.7% | 84.910 | 9% |
| 51 | 51.109 | 10.9% | 49.710 | 100% |
| | 51.386 | 38.6% | 50.204 | 79.6% |
| | 50.872 | 12.8% | 51.455 | 45.5% |

Las distancias fueron medidas en el tiempo de 7 minutos (como se encuentran en el programa principal). El sensor JSN-SR04T tiene mayor precisión que los sensores SC-HR04T.

Esto también se ve afectado por el dato de velocidad del sonido que se utilizó en el cálculo de la distancia.

Tabla 3.3 Comparación de resultados de sensores de distancia

| Distancia en cm Sensor JSN-SR04T | Distancia en cm Sensor SC-HR04T |
|---|--|
|  |  |
|  |  |

3.3. Visualización web.

El programa principal se enlazará con la base de datos y enviará valores cada 10 minutos, tiempo que permite un menor consumo de energía y una variación significativa en los valores de distancia.

| | id | Fecha_Hora | Latitud | Longitud | Distancia | Vacio | Medio | Lleno |
|--------------------------|-----|---------------------|----------|----------|-----------|-------|-------|-------|
| <input type="checkbox"/> | 152 | 2019-05-03 15:06:34 | -1.28137 | -78.6143 | 113.515 | 0 | 0 | 0 |
| <input type="checkbox"/> | 153 | 2019-05-03 15:16:34 | -1.28137 | -78.6143 | 28.5404 | 0 | 0 | 0 |
| <input type="checkbox"/> | 154 | 2019-06-06 20:35:12 | -1.28137 | -78.6143 | 71.6535 | 0 | 0 | 0 |
| <input type="checkbox"/> | 155 | 2019-07-30 22:15:22 | -1.28137 | -78.6143 | 59.3644 | 0 | 0 | 0 |
| <input type="checkbox"/> | 156 | 2019-07-30 22:27:18 | -1.28137 | -78.6143 | 43.381 | 0 | 0 | 0 |
| <input type="checkbox"/> | 157 | 2019-07-30 22:32:01 | -1.28137 | -78.6143 | 43.381 | 0 | 0 | 0 |
| <input type="checkbox"/> | 158 | 2019-07-30 22:53:39 | -1.28137 | -78.6143 | 43.381 | 0 | 1 | 0 |
| <input type="checkbox"/> | 159 | 2019-07-31 19:51:59 | -1.28137 | -78.6143 | 119.767 | 1 | 0 | 0 |
| <input type="checkbox"/> | 160 | 2019-09-03 13:14:40 | -1.28136 | -78.6143 | 82.7657 | 1 | 0 | 0 |
| <input type="checkbox"/> | 161 | 2019-09-03 13:15:11 | -1.28136 | -78.6143 | 82.7657 | 1 | 0 | 0 |
| <input type="checkbox"/> | 162 | 2019-09-03 13:36:41 | -1.28136 | -78.6143 | 73.0083 | 0 | 1 | 0 |
| <input type="checkbox"/> | 163 | 2019-09-03 13:43:42 | -1.28136 | -78.6143 | 71.6072 | 0 | 1 | 0 |
| <input type="checkbox"/> | 164 | 2019-09-03 13:53:56 | -1.28136 | -78.6143 | 72.305 | 0 | 1 | 0 |
| <input type="checkbox"/> | 165 | 2019-09-03 14:03:57 | -1.28136 | -78.6143 | 72.4522 | 0 | 1 | 0 |
| <input type="checkbox"/> | 166 | 2019-09-03 14:13:58 | -1.28136 | -78.6143 | 71.1669 | 0 | 1 | 0 |
| <input type="checkbox"/> | 167 | 2019-09-03 14:23:58 | -1.28136 | -78.6143 | -2402010 | 0 | 0 | 1 |
| <input type="checkbox"/> | 168 | 2019-09-03 14:34:01 | -1.28136 | -78.6143 | -2402010 | 0 | 0 | 1 |
| <input type="checkbox"/> | 169 | 2019-09-03 14:37:36 | -1.28136 | -78.6143 | -2402010 | 0 | 0 | 1 |

Figura 3.10 Prueba de funcionamiento en la base de datos.

Las páginas web están desarrolladas en el lenguaje PHP, que permite realizar una conexión con MySQL para adquirir de manera automática los valores guardados en cada una de las tablas de datos y mostrarlos en tiempo real en la pantalla principal.

ESCUELA POLITÉCNICA NACIONAL

VALORES DEL MONITOREO DE UN CONTENEDOR DE BASURA IOT

Tabla de valores del contenedor de basura IOT

| id | Fecha_Hora | Latitud | Longitud | Distancia | Vacio | Medio | Lleno |
|-----|---------------------|----------|----------|-----------|-------|-------|-------|
| 152 | 2019-05-03 15:06:34 | -1.28137 | -78.6143 | 113.515 | 0 | 0 | 0 |
| 153 | 2019-05-03 15:16:34 | -1.28137 | -78.6143 | 28.5404 | 0 | 0 | 0 |
| 154 | 2019-06-06 20:35:12 | -1.28137 | -78.6143 | 71.6535 | 0 | 0 | 0 |
| 155 | 2019-07-30 22:15:22 | -1.28137 | -78.6143 | 59.3644 | 0 | 0 | 0 |
| 156 | 2019-07-30 22:27:18 | -1.28137 | -78.6143 | 43.381 | 0 | 0 | 0 |
| 157 | 2019-07-30 22:32:01 | -1.28137 | -78.6143 | 43.381 | 0 | 0 | 0 |
| 158 | 2019-07-30 22:53:39 | -1.28137 | -78.6143 | 43.381 | 0 | 1 | 0 |
| 159 | 2019-07-31 19:51:59 | -1.28137 | -78.6143 | 119.767 | 1 | 0 | 0 |
| 160 | 2019-09-03 13:14:40 | -1.28136 | -78.6143 | 82.7657 | 1 | 0 | 0 |
| 161 | 2019-09-03 13:15:11 | -1.28136 | -78.6143 | 82.7657 | 1 | 0 | 0 |
| 162 | 2019-09-03 13:36:41 | -1.28136 | -78.6143 | 73.0083 | 0 | 1 | 0 |
| 163 | 2019-09-03 13:43:42 | -1.28136 | -78.6143 | 71.6072 | 0 | 1 | 0 |

CONTACTOS:
 edisson.acosta@epn.edu.ec
 ricardo.llugsi@epn.edu.ec

Figura 3.11 Visualización de la página monitoreo.php

En la página de ubicación se observará un mapa con un menú de opciones desplegable que contiene todos los registros por fechas, latitud y longitud, al momento de seleccionar una opción en el mapa aparecerá el indicador respectivo de la geolocalización del contenedor.

ESCUELA POLITÉCNICA NACIONAL

UBICACIÓN DEL CONTENEDOR DE BASURA

Obtener mi ubicación - A

Latitud

Longitud

Dibujar ruta con ↓

Dibujar ruta con ↓

- 2019-05-03 15:06:34-1.28137 -78.6143
- 2019-05-03 15:16:34-1.28137 -78.6143
- 2019-06-06 20:35:12-1.28137 -78.6143
- 2019-07-30 22:15:22-1.28137 -78.6143
- 2019-07-30 22:27:18-1.28137 -78.6143
- 2019-07-30 22:32:01-1.28137 -78.6143
- 2019-07-30 22:53:39-1.28137 -78.6143
- 2019-07-31 19:51:59-1.28137 -78.6143

CONTACTOS:
 edisson.acosta@epn.edu.ec
 ricardo.llugsi@epn.edu.ec

Figura 3.12 Visualización de opciones en la página ubicacion.php

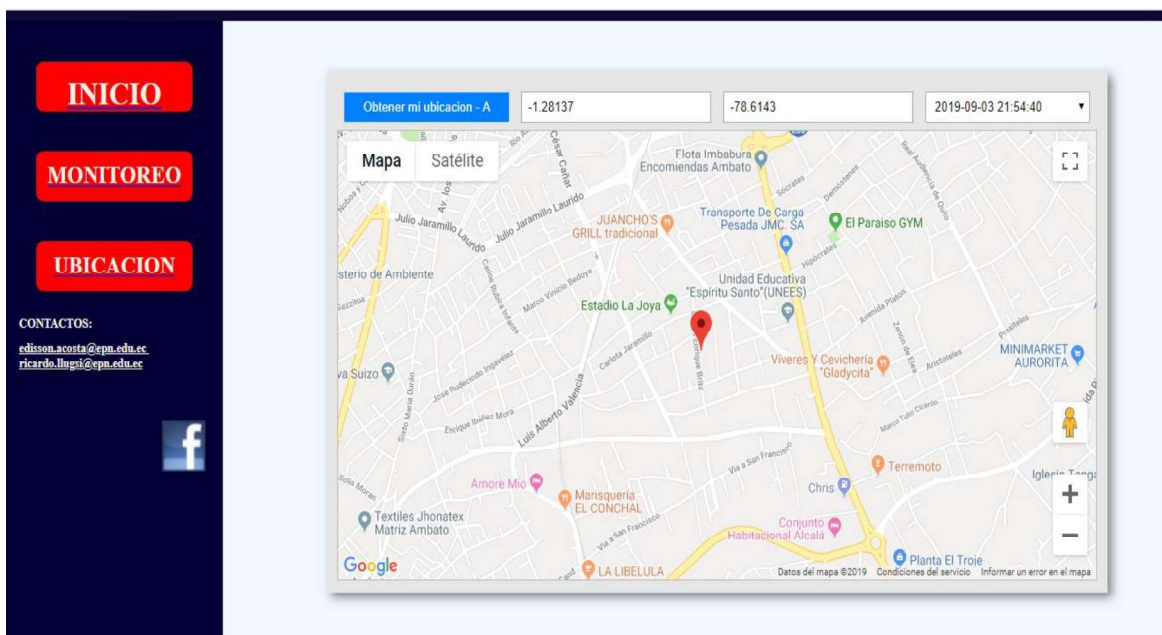


Figura 3.13 Visualización de la geolocalización en la página ubicacion.php

3.4. Presupuesto Referencial.

En esta sección se detallarán los precios de los elementos utilizados en el desarrollo del prototipo en protección, sistema de alimentación y hardware, respectivamente.

Tabla 3.4 Tabla de costos de construcción

| Elementos | Costo total \$ |
|---|----------------|
| Estructura de la maqueta del contenedor | 50 |
| Caja 25x25 cm | 13 |
| Caja acrílica (3) | 10 |
| Cables | 7 |
| Cubierta termofundente | 3 |
| Placa | 15 |
| Canaletas | 2 |
| Subtotal 1 | 100 |

Tabla 3.5 Tabla de costos de alimentación

| Elementos | Costo total \$ |
|----------------------------|----------------|
| Panel Solar 15 (W) 35x35cm | 35 |
| Batería | 18 |
| Controlador de carga solar | 15 |
| Subtotal 2 | 68 |

Tabla 3.6 Tabla de costos de hardware

| Elementos | Costo total \$ |
|---------------------|----------------|
| Raspberry Pi Zero | 25 |
| Sensor jsn-sr04t | 20 |
| Sensor hr (2)-sr04 | 10 |
| Sensor Ublox Neo 7M | 25 |
| Modem Huawei E3531 | 15 |
| Hub USB | 8 |
| otros | 37 |
| Subtotal 3 | 140 |

Tabla 3.7 Tabla de costo total

| Elementos | Costo total \$ |
|------------|----------------|
| Subtotal 1 | 100 |
| Subtotal 2 | 68 |
| Subtotal 3 | 140 |
| Total | 308 |

Cabe mencionar que es servicio de AWS y Google Maps es gratuito.

4. CONCLUSIONES Y RECOMENDACIONES

En este capítulo se detallan las conclusiones y recomendaciones del proyecto realizado.

4.1. Conclusiones

- El proyecto ayudará al mejoramiento del ornato de la ciudad, permitirá establecer nuevas rutas de recolección al visualizar los datos obtenidos en las páginas web.
- El prototipo desarrollado al estar expuesto a diferentes condiciones ambientales cumple satisfactoriamente con las especificaciones de protección propuestas, asegurando así su vida útil.
- La tarjeta de procesamiento Raspberry pi Zero es inestable a las variaciones bruscas de corriente, por lo que se reiniciará cada vez que sufra una descompensación.
- El sistema fotovoltaico cumple satisfactoriamente con la alimentación del circuito, no se evidenciaron sobrevoltajes, ya que la batería puede soportar hasta 24 V y 5 A, pero el controlador de carga siempre enviará 5 V y 3 A en la salida USB.
- El Modem USB Huawei E3531, no es completamente compatible con el sistema Raspbian, necesita configuraciones adicionales a los programas sakis3g y umtskeeper, hasta obtener una estabilidad en la conexión ppp0.
- El APN de la operadora Claro puede ser configurado de dos maneras, la primera automáticamente reconocida por el programa Sakis3g, "ba.amx" y la segunda editarla manualmente con el nombre "internet.claro.com.ec", siendo esta última la que brindó mejores prestaciones al momento de la conexión ppp0.
- Las pruebas realizadas con un solo sensor (JSN SR04T) no ofrecieron los resultados esperados, al no abastecer la totalidad de la superficie del contenedor, se añadieron dos sensores (HC SR04) para cubrir los espacios, la cantidad final es

un promedio de los tres sensores, pero la alerta de llenado se da cuando las mediciones de los tres sensores sobrepasan el nivel de lleno.

- El módulo GPS y modem USB, utilizados funcionan correctamente dentro de la caja de protección metálica, no existe afección en las ondas electromagnéticas.
- La Raspberry Pi Zero al tener una memoria RAM de 512 MB su arranque es lento, tarda alrededor de 3 minutos en arrancar el sistema y otros 3 minutos en establecer la conexión ppp0 para la conexión a internet.
- Para el envío de mensajes de alerta se optó por correos electrónicos y no por SMS, debido a que SMS utiliza el puerto ttyUSB0 el cual se encuentra ocupado por el servicio de internet de la interfaz ppp0.
- Las pruebas realizadas con el módulo GPS son satisfactorias la precisión en la adquisición de datos es alrededor de un 99%, su funcionamiento únicamente se verá afectado dependiendo la obstrucción que tenga a su alrededor.
- Se comprobó que el sensor JSN SR04T tiene mayor precisión y un mayor ángulo de medición que los sensores HC SR04 en las pruebas realizadas no obstante otro factor que afecta directamente la medición de la distancia para ambos sensores es el valor de la velocidad del sonido, obteniendo mejores resultados con 34300 cm/s en lugar de 34000 cm/s.

4.2. Recomendaciones

- El sensor JSN SR04T puede ser omitido debido a su costo, en el uso de un proyecto similar, si bien cuenta con mejores características para una medición directa, no cubre un espacio ancho, por lo que al usar los sensores HC SR04 se obtienen resultados satisfactorios.

- Antes de adquirir un modem USB 3G, se recomienda observar las características `usb_modeswitch` donde indica las identificaciones `default`, `target` y número de `message content` necesarios para cambiar el modo de funcionamiento de almacenamiento al modo modem de los modelos que son compatibles con Raspberry.
- La batería del sistema fotovoltaico se puede cambiar por una de mayor capacidad con la finalidad de brindar mayor tiempo en la autonomía de trabajo, especialmente para temporadas donde existe poca cantidad de luz solar ya que el panel solar no obtiene energía suficiente para abastecer el sistema de alimentación.
- Existen modem USB los cuales ofrecen un servicio broad band que al ser configurados en la Raspberry brindan dos interfaces de conexión `ppp0` y `wwan0`, al digitar el comando `"route -n"` se puede observar cuál de las dos tiene mayor prioridad en este caso es `wwan0`, esta interfaz necesita otro tipo de configuraciones que pueden o no ser exitosas para su conexión a Internet, una opción fue instalar `NetworkManager` sin éxito en el autoarranque, por lo cual es recomendable bloquear la interfaz `wwan0` y trabajar solamente con la interfaz `ppp0`.
- La API de Google Maps no es completamente gratuita, permite su visualización con todas sus prestaciones una vez al día, por lo que si desea tener un mayor número de visualizaciones al día debe crearse una cuenta para facturación, y cambiar la página web a modo seguro (`https`).
- Al momento de crearse una instancia en Amazon Ec2, la máquina virtual está configurada con la zona horaria de los servidores de Amazon, se recomienda cambiar la zona horaria por el terminal mediante el comando `"sudo dpkg-reconfigure tzdata"`.

- Una mejora del proyecto es poder visualizar en la página web de ubicación el trazo de rutas en Google Maps desde un punto neutral hacia el lugar dónde se encuentren los contenedores de basura.
- Para cubrir todo el espacio del contenedor de basura los sensores deben ubicarse en las tapas del contenedor de basura, con su respectiva protección.
- Para la implementación real de este prototipo, la ubicación del circuito principal, así como del sistema de alimentación deben encontrarse a un costado del contenedor de basura, ya que al momento de la recolección son vulnerables a posibles golpes que dañarían su funcionamiento.

5. REFERENCIAS BIBLIOGRÁFICAS

- [1] Naylamp Mechatronics SAC. "Sensor Ultrasonido JSN-SR04T". [Online]. Available: <https://naylampmechatronics.com/sensores-proximidad/326-sensor-ultrasonido-jsn-sr04t.html> [Accessed: Enero 2019]
- [2] J. Sanchez. "Qué son los residuos sólidos y cómo se clasifican",2018. [Online]. Available: <https://www.ecologiaverde.com/que-son-los-residuos-solidos-y-como-se-clasifican-1537.html>. [Accessed: Noviembre 2018]
- [3] Secretaría de Ambiente. "Políticas y planeación ambiental". [Online]. Available: <http://www.quitoambiente.gob.ec/ambiente/index.php/politicas-y-planeacion-ambiental/residuos-solidos>. [Accessed: Diciembre 2018]
- [4] SEMARNAT. "Impacto Ambiental y Tipos", 2013. [Online]. Available: <http://www.semarnat.gob.mx/temas/gestion-ambiental/impacto-ambiental-y-tipos>. [Accessed: Noviembre 2018]
- [5] M. Muñoz. Manual de Manejo de Residuos Sólidos Urbanos. Primera edición. Quito: Impremédios 2008.
- [6] Ecube Labs. (2016, July 8). Overflowing garbage bins: 5 impacts on health and environment, and how to prevent. [Online]. Available: <https://www.ecubelabs.com/overflowing-garbage-bins-5-impacts-on-health-and-environment-and-how-to-prevent/> [Accessed: Septiembre 2018]
- [7] OPS. Evaluación de los Efectos de la Contaminación del Aire en la Salud de América Latina y el Caribe. Washington, D.C: Organización Mundial de la Salud, 2005.
- [8] EMASEO. "Recolección Mecanizada". [Online]. Available: <http://www.emaseo.gob.ec/servicios/recoleccion-mecanizada/>. [Accessed: Septiembre 2018]
- [9] J. Regis. Comunicaciones inalámbricas de banda ancha. Primera edición. Madrid: McGRAW-HILL, 2003.
- [10] J. Huidobro Maya. Manual de Telecomunicaciones. Segunda edición. Bogotá: Ra-ma Editorial, 2015.

- [11] M. Abella. "Sistemas Fotovoltaicos". Tesis. Departamento de Rnergías Renovables. Escuela de Organización Industrial, Madrid.
- [12] Área Tecnología. "Regulador de carga solar". [Online]. Available: <https://www.areatecnologia.com/electricidad/regulador-de-carga-solar.html>. [Accessed: Julio 2018]
- [13] Prometec. "Llamar y enviar sms". [Online]. Available: <https://www.prometec.net/gprs-llamar-enviar-sms/#>. [Accessed: Diciembre 2018]
- [14] SimCom. "Sim800L Hardware Desing V1.0". [Online]. Available: https://img.filipeflop.com/files/download/Datasheet_SIM800L.pdf. [Accessed: Diciembre 2018]
- [15] Itead Studio. "GPRS Module: Sim 900 GSM/GPRS Module". [Online]. Available: <http://image.dfrobot.com/image/data/TEL0051/3.0/SIM908%20datasheet.pdf>. [Accessed: Enero 2019]
- [16] SimCom. "Sim908". [Online]. Available: <http://image.dfrobot.com/image/data/TEL0051/3.0/SIM908%20datasheet.pdf>. [Accessed: Enero 2019]
- [17] J. Maugard. "Módem USB 3G o 4G", 2017. [Online]. Available: <https://www.killmybill.es/modem-usb-3g-4g/>. [Accessed: Enero 2019]
- [18] Ublox. "NEO-7: u-blox 7 GPS modules Data Sheet". [Online]. Available: https://www.u-blox.com/sites/default/files/products/documents/NEO-7_DataSheet_%28UBX-13003830%29.pdf. [Accessed: Enero 2019]
- [19] Ublox. "NEO-6: u-blox 6 GPS modules Data Sheet". [Online]. Available: https://www.u-blox.com/sites/default/files/products/documents/NEO-6_DataSheet_%28GPS.G6-HW-09005%29.pdf. [Accessed: Enero 2019]
- [20] Electronic Components Datasheet. "VL53L0X Datasheet". [Online]. Available: <https://html.alldatasheet.com/html-pdf/948120/STMICROELECTRONICS/VL53L0X/1944/1/VL53L0X.html>. [Accessed: Enero 2019]
- [21] Electronic Components Datasheet. "VL53L0X Datasheet". [Online]. Available: <https://html.alldatasheet.com/html->

pdf/948120/STMICROELECTRONICS/VL53L0X/11694/6/VL53L0X.html. [Accessed: Enero 2019] (tabla)

[22] Datasheets Pdf. "HC-SR04 Datasheet: Ultrasonic Sensor user manual". [Online]. Available: <https://www.datasheetspdf.com/pdf/1291829/Cytron/HC-SR04/1>. [Accessed: Enero 2019]

[23] Arduino. "Que es Arduino". [Online]. Available: <https://www.arduino.cc/en/Guide/Introduction>. [Accessed: Febrero 2019]

[24] Arduino. "Arduino Uno". [Online]. Available: <https://store.arduino.cc/usa/arduino-uno-rev3>. [Accessed: Febrero 2019]

[25] Arduino. "Arduino Mega 2560". [Online]. Available: <https://store.arduino.cc/usa/arduino-mega-2560-rev3>. [Accessed: Febrero 2019]

[26] S. Monk. Ejercicios prácticos con Raspberry: Problemas y soluciones de Software y Hardware. Segunda edición español. Barcelona: Marcombo S.A., 2017

[27] Culturación. "Raspberry: Que es, características y precios". [Online]. Available: <http://culturacion.com/raspberry-pi-que-es-caracteristicas-y-precios/>. [Accessed: Marzo 2019]

[28] Naylamp Mechatronics SAC. "Raspberry Pi Zero". [Online]. Available: https://naylampmechatronics.com/raspberry-pi/450-raspberry-pi-zero-barebones-kit.html?search_query=raspberry&results=50. [Accessed: Marzo 2019]

[29] Naylamp Mechatronics SAC. "Raspberry Pi 3B+". [Online]. Available: https://naylampmechatronics.com/raspberry-pi/255-raspberry-pi-3-b.html?search_query=raspberry&results=50. [Accessed: Marzo 2019]

[30] MySQL. "MySQL Enterprise Edition". [Online]. Available: <https://www.mysql.com/products/enterprise/>. [Accessed: Marzo 2019]

[31] Apache. "Apache". [Online]. Available: <https://www.apache.org/>. [Accessed: Marzo 2019]

[32] phpMyAdmin. "Trayendo MySQL a la web". [Online]. Available: <https://www.phpmyadmin.net/>. [Accessed: Marzo 2019]

- [33] Códico facilito. “Qué es HTML”. [Online]. Available: <https://codigofacilito.com/articulos/que-es-html>. [Accessed: Julio 2019]
- [34] PHP. “Que puede hacer PHP”. [Online]. Available: <https://www.php.net/manual/es/intro-whatcando.php>. [Accessed: Julio 2019]
- [35] E. Tatayo. “Implementación de una mini red de sensores inalámbricos para la detección temprana de incendios forestales utilizando Hardware y Software open source”. Escuela Politécnica Nacional. Facultad de Electrónica y Telecomunicaciones. Quito, 2017
- [36] Python. “Python”. [Online]. Available: <https://www.python.org/>. [Accessed: Junio 2019]
- [37] Raspberry Pi. “Raspbian”. [Online]. Available: <https://www.raspberrypi.org/downloads/raspbian/>. [Accessed: Abril 2019]
- [38] Internet de las cosas. “Configurando una raspberry Pi Zero Wifi sin un monitor”. [Online]. Available: <https://www.internetdelascosas.cl/2017/10/30/configurando-una-raspberry-pi-zero-wifi-sin-un-monitor/>. [Accessed: Abril 2019]
- [39] Genius. “Programación en Raspberry Pi con MySQL IOT”. Secretaría Técnica del Sistema Nacional de Cualificaciones Profesionales. Quito, 2018.
- [40] Spell Foundry. “Configurando el puerto serio GPIO en Raspbian Jessie y Stretch incluyendo Pi 3”. [Online]. Available: <https://spellfoundry.com/2016/05/29/configuring-gpio-serial-port-raspbian-jessie-including-pi-3/>. [Accessed: Mayo 2019]
- [41] Medium Corporation. “Usando un módulo GPS (neo-7m) con Raspberry Pi 3”. [Online]. Available: https://medium.com/@DefCon_007/using-a-gps-module-neo-7m-with-raspberry-pi-3-45100bc0bb41. [Accessed: Mayo 2019]
- [42] Carlini’s Blog. “Internet móvil con el modem 3G USB Huawei E303 en la Raspberry Pi”. [Online]. Available: <http://carlini.es/internet-movil-con-el-modem-3g-usb-huawei-e303-en-la-raspberry-pi/>. [Accessed: Junio 2019]
- [43] Real Droid. “Monta tu propio servidor web en casa con Raspberry Pi, y ten tu página personal”. [Online]. Available: <https://www.realdroid.es/2017/01/20/monta-tu-propio-servidor-web-en-casa-con-raspberry-pi-y-ten-tu-pagina-personal/>. [Accessed: Mayo 2019]

- [44] Pixlr. "Editor". [Online]. Available: <https://pixlr.com/editor/>. [Accessed: Mayo 2019]
- [45] W3schools. "Tutorial HTML5". [Online]. Available: <https://www.w3schools.com/html/>. [Accessed: Mayo 2019]
- [46] Google. "Usando MySQL y PHP con Google Maps". [Online]. Available: <https://developers.google.com/maps/documentation/javascript/mysql-to-maps>. [Accessed: Mayo 2019]
- [47] Foros Raspberry. "Cómo encender y apagar tu raspberry a través de un botón". [Online]. Available: <https://www.fororaspberry.es/viewtopic.php?t=1572>. [Accessed: Julio 2019]
- [48] C. Terán. "Implementación de un sistema de captura de imágenes térmicas y su almacenamiento en la web". Escuela Politécnica Nacional. Facultad de Electrónica y Telecomunicaciones. Quito, 2018
- [49] ElectroSome. "Usando Raspberry Pi a través de VNC". [Online]. Available: <https://electrosome.com/using-raspberry-pi-vnc/> [Accessed: Junio 2019]
- [50] AWS. "Amazon Elastic Compute Cloud". [Online]. Available: https://docs.aws.amazon.com/es_es/AWSEC2/latest/UserGuide/concepts.html [Accessed: Junio 2019]
- [51] Alquienvas Group. "Contenedores metálicos". [Online]. Available: http://www.alquienvas.com/productos/18-1800,_2400_y_3200_litros-827-52.html. [Accessed: Junio 2019]
- [52] Digilogic. "GPIO en Raspberry Pi Zero". [Online]. Available: <https://www.digilogic.es/gpio-raspberry-pi-zero/>. [Accessed: Junio 2019]
- [53] Makerfabs. "Water Proof Ultrasonic Ranger- JSN-SR04T". [Online]. Available: https://www.makerfabs.com/index.php?route=product/product&product_id=185. [Accessed: Junio 2019]
- [54] Gearbest. "GYGPSV3 NEO-7M C06 APM2.5 Módulo Ublox GPS - Azul". [Online]. Available: https://es.gearbest.com/boards-shields/pp_248296.html. [Accessed: Junio 2019]

[55] Amazon. "Huawei e3531s-6 HiLink módem 3 G 21,6 Mbps". [Online]. Available: <https://www.amazon.com.mx/Huawei-e3531s-6-HiLink-m%C3%B3dem-Mbps-Desbloqueado/dp/B01BN9XQLQ> [Accessed: Julio 2019]

[56] Amazon. "Regulador de carga solar 12V-24V". [Online]. Available: <https://www.amazon.es/Regulador-12V-24V-Controlador-Inteligente-Pantalla/dp/B01J5A61B2>. [Accessed: Junio 2019]

ORDEN DE EMPASTADO