

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

DISEÑO DE UN SISTEMA PROTOTIPO PARA LA ADQUISICIÓN Y TRANSMISIÓN DE DATOS DE TEMPERATURA SUPERFICIAL DEL AGUA A TRAVÉS DE LA RED CELULAR EN UN AMBIENTE CONTROLADO

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

CRISTIAN RODRIGO CARRILLO BALDEÓN

cristian.carrillo@epn.edu.ec

DIRECTOR: ING. RICARDO XAVIER LLUGSI CAÑAR, MSc.

ricardo.llugsi@epn.edu.ec

CODIRECTOR: ING. MARCO ESTEBAN YACELGA PINTO, MSc.

marco.yacelgap@epn.edu.ec

Quito, julio 2018

AVAL

Certificamos que el presente trabajo fue desarrollado por Cristian Rodrigo Carrillo Baldeón, bajo nuestra supervisión.

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

**ING. MARCO ESTEBAN YACELGA PINTO, MSC.
CODIRECTOR DEL TRABAJO DE TITULACIÓN**

DECLARACIÓN DE AUTORÍA

Yo, Cristian Rodrigo Carrillo Baldeón, 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.

CRISTIAN RODRIGO CARRILLO
BALDEÓN

DEDICATORIA

Este logro se lo dedico a mi padre y hermanas por el inmenso apoyo, ejemplo y enseñanzas que he recibido incondicionalmente. Siempre estuvieron dándome la fortaleza para seguir adelante en los momentos difíciles, sin ellos a mi lado no hubiera logrado esta meta; este logro es de todos.

A mi ángel que siempre está a mi lado, mi madre, que desde el cielo me guía para cumplir todas las metas propuestas; espero que estés orgullosa por todo lo logrado y me sigas acompañando en este viaje hasta volvernos a encontrar.

Cristian R. Carrillo B.

AGRADECIMIENTO

A mi padre por la ayuda y consejos incondicionales brindados durante toda la carrera. Sin su apoyo no hubiera podido culminar esta etapa importante de mi vida.

A mis hermanas, que de una u otra manera siempre me ayudaron a resolver cualquier problema, nunca se negaron a ayudarme cuando las necesite, fueron el soporte necesario para seguir adelante.

Al Msc. Ricardo Llugsí y Msc. Marco Yacelga por su ayuda durante la elaboración del proyecto. Gracias por el tiempo, consejos y conocimiento brindado, gracias por confiar en mí para la elaboración de este proyecto.

A Katy por estar presente en todos los momentos alegres de mi vida, pero sobre todo por estar en los momentos más difíciles, apoyándome y sacándome una sonrisa sin esperar nada a cambio.

Cristian R. Carrillo B.

ÍNDICE DE CONTENIDO

AVAL	I
DECLARACIÓN DE AUTORÍA.....	II
DEDICATORIA.....	III
AGRADECIMIENTO.....	IV
ÍNDICE DE CONTENIDO.....	V
RESUMEN	IX
ABSTRACT	X
1. INTRODUCCIÓN.....	1
1.1 Objetivos	1
1.2 Alcance	2
1.3 Marco Teórico	4
1.3.1 Parámetros medioambientales del agua	4
1.3.1.1 Parámetros Físicos.....	4
1.3.1.2 Parámetros Químicos.....	6
1.3.1.3 Parámetros Biológicos.....	6
1.3.2 Tecnologías para la transmisión de datos	7
1.3.2.1 El espacio radioeléctrico.....	10
1.3.2.2 Tecnología Celular	11
1.3.2.3 Comunicaciones Satelitales	16
1.3.3 Sistemas fotovoltaicos con paneles solares y baterías.....	19
1.3.3.1 Módulos fotovoltaicos	19
1.3.3.2 Baterías	20
1.3.4 Conceptos para la creación de páginas web	21
1.3.4.1 HTML (<i>HyperText Markup Language</i> , Lenguaje de Marcado para Hipertextos)	21
1.3.4.2 PHP (<i>Hypertext Preprocessor</i> , Procesador de Hipertexto).....	22
1.3.4.3 Base de datos	22
1.3.4.4 Python	23
2. METODOLOGÍA.....	24
2.1 Plataformas de hardware libre.	24

2.1.1	Cubieboard	24
2.1.2	Intel Galileo.....	25
2.1.3	Arduino	26
2.1.3.1	Arduino Uno	27
2.1.3.2	Arduino Mega.....	27
2.1.4	Raspberry Pi.....	28
2.1.4.1	Raspberry Pi 2 Modelo B.....	29
2.1.4.2	Raspberry Pi 3 Modelo B.....	29
2.1.5	Selección de la plataforma de hardware libre	30
2.2	Sensores de Temperatura.....	32
2.2.1	Termopares	33
2.2.2	Termistores.....	33
2.2.3	Detectores de temperatura de resistencia	33
2.2.4	Sensores Infrarrojos	34
2.2.5	Sensor de Temperatura DS18B20.....	34
2.2.5.1	Características	35
2.2.5.2	Error en la medición	36
2.2.5.3	Métodos de alimentación.....	36
2.3	Modem 4G	37
2.3.1	Modem Mobile WiFi.....	38
2.4	Sistema Fotovoltaico	39
2.4.1	Dimensionamiento del sistema fotovoltaico	39
2.4.2	Batería	41
2.4.3	Paneles Solares.....	41
2.5.	Configuración de los equipos	43
2.5.1	Configuración de la Raspberry Pi	43
2.5.1.1	Instalación del sistema operativo	43
2.5.1.2	Configuraciones iniciales.....	45
2.5.2	Configuración de los sensores	47
2.5.2.1	Tiempo de muestreo de temperatura	47
2.5.2.2	Modo de conexión	47
2.5.2.3	Programa para tomar datos de los sensores.....	49
2.5.3	Programas adicionales	52
2.5.3.1	Programa Shutdown.....	53

2.5.3.2	Automatización de los scripts	54
2.5.4	Configuración del modem 4G	55
2.5.5	Configuración de la base de datos.	57
2.5.5.1	Instalación MySQL y phpMyAdmin.....	57
2.5.5.2	Creación de la base de datos con phpMyAdmin	58
2.5.6	Desarrollo sitio web	62
2.5.6.1	Desarrollo del Header.....	66
2.5.6.2	Desarrollo página web inicio.....	67
2.5.6.3	Desarrollo página web temperatura.php.....	71
2.5.6.4	Desarrollo página web tendencia.php	77
2.5.7	Sistema fotovoltaico.....	82
2.5.8	Circuitos adicionales.....	84
2.5.8.1	Circuito para conectar los paneles solares.....	84
2.5.8.2	Circuito para conectar los sensores	85
2.5.8.3	Circuito para encender/apagar la Raspberry Pi.....	85
2.5.9	Conexión de los equipos	85
2.5.10	Mecanismos para que el prototipo sea resistente al agua.....	88
2.5.11	Estructura de soporte para el prototipo.....	91
3.	RESULTADOS Y DISCUSIÓN	94
3.1	Laguna de Colta	94
3.2	Pruebas de Funcionamiento	95
3.2.1	Pruebas iniciales en un tanque.....	96
3.2.2	Pruebas finales en la Laguna de Colta	100
3.3	Análisis de resultados.	104
3.3.1	Gráficas de tendencias	104
4.	CONCLUSIONES	110
4.1	Conclusiones.....	110
4.2	Recomendaciones.....	111
5.	REFERENCIAS BIBLIOGRÁFICAS	113
6.	ANEXOS.....	117
ANEXO I	118
ANEXO II	122
ANEXO III	126
ANEXO IV	134

ANEXO V	148
ORDEN DE EMPASTADO	149

RESUMEN

Los ríos, lagos y lagunas son de gran importancia en la humanidad, no solo porque son proveedores de agua dulce para las personas y animales, sino que también ayudan en la industria y en la agricultura. Además el agua de los ríos ayuda a regular la temperatura del medio ambiente que es importante tanto para factores ambientales-climáticos como para plantas y animales.

Con este proyecto se busca adquirir y transmitir datos de temperatura superficial del agua a una página web, a través de la red celular en un ambiente controlado. Se medirá la temperatura del agua a un metro, tres metros y cinco metros con sensores digitales de temperatura tipo sonda resistente al agua. Los datos de temperatura serán procesados por un computador de placa reducida Raspberry Pi, esta se encargará de procesar la información recolectada por los sensores y enviarla a través de la red celular cada cierto tiempo de manera automática para poder ser mostrada en la página web. Se cuidará que el prototipo este bien protegido para que no sufra daños por el ambiente para el que es diseñado (resistente al agua, resistente a temperaturas altas por el sol, polvo, etc.)

Finalmente las pruebas de funcionamiento se realizarán en la laguna de Colta ya que presenta todas las características deseadas para probar el prototipo tanto en la recolección de datos como en la resistencia al ambiente.

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

ABSTRACT

Rivers, lakes and lagoons are of great importance in humanity, not only because they provide fresh water for people and animals, but also help in industry and agriculture. In addition, river water helps regulate the temperature of the environment, which is important both for environmental-climatic factors and for plants and animals.

This project seeks to acquire and transmit water surface temperature data to a web page, through the cellular network in a controlled environment. The temperature of the water will be measured at one meter, three meters and five meters with digital temperature sensors type waterproof probe. The temperature data will be processed by the Raspberry Pi board, this will be responsible for processing the information collected by the sensors and send it through the cellular network every certain time automatically to be displayed on the website. Care should be taken that the prototype is well protected so that it does not suffer damage by the environment for which it is designed (resistant to water, resistant to high temperatures by the sun, dust, etc.)

Finally, the performance tests will be carried out in the Colta lagoon since it presents all the desired characteristics to test the prototype both in the data collection and in the resistance to the environment.

KEYWORDS: Wireless Communications, Free Hardware, Database, Sensors.

1. INTRODUCCIÓN

La importancia de los ríos, lagos y lagunas en el planeta excede cualquier posibilidad de interpretaciones erróneas acerca de sus parámetros medioambientales. Los ríos tienen mucha importancia en la humanidad ya que son proveedores de agua dulce para las personas, industria y agricultura. Además, el agua de los ríos ayuda a regular la temperatura del medio en el que se encuentran.

La temperatura del agua es importante tanto para factores ambientales como para factores climáticos. La importancia radica en que tanto animales y plantas acuáticas están adaptados a vivir en condiciones determinadas; cambios bruscos o prolongados en la temperatura del agua pueden afectarles, llegando incluso a morir.

Se busca que el prototipo mida la temperatura del agua de los ríos, lagos o lagunas a tres diferentes alturas: a un metro, a tres metros y a cinco metros bajo el nivel del agua; ya que la temperatura cambia tanto estacionalmente como con la profundidad.

El prototipo puede fallar por motivos diversos, pero en particular por el ambiente en el que se encuentra puede ser por la acción corrosiva del medio ambiente (temperatura, humedad, polvo, contaminación). Tan solo pequeñas cantidades de agentes agresivos o de productos de corrosión bastan para causar una grave degradación de los componentes electrónicos, e incluso el fallo total del sistema.

La corrosión puede determinar la fiabilidad y vida útil de muchos de los equipos. Este es uno de los grandes problemas que surgen al instalar equipos electrónicos en ambientes como ríos del país ya que la temperatura, el polvo, el ingreso de agua a los equipos hacen que estos fallen.

1.1 Objetivos

El objetivo general de este Proyecto es: Adquirir y transmitir datos de temperatura superficial del agua de tres diferentes profundidades a una página web, a través de la red celular en un ambiente controlado.

Los objetivos específicos de este Proyecto son:

- a) Describir una visión general de los parámetros medioambientales de la temperatura del agua y de las tecnologías para transmisión de datos.

- b) Desarrollar el prototipo para tomar datos de temperatura del agua cada cierto tiempo a tres diferentes profundidades y enviar estos datos de manera autónoma a través de enlace celular hacia una página web.
- c) Usar sistema fotovoltaico para la alimentación autónoma del prototipo.
- d) Implementar un mecanismo de protección contra el ingreso de agua al armario de circuitos, buscando un mecanismo para adaptar y probar el prototipo en un ambiente controlado.
- e) Realizar pruebas de funcionamiento para que el prototipo transmita valores de temperatura sin problemas y que este sea resistente al agua.

1.2 Alcance

Con el presente trabajo se medirá la temperatura del agua a tres diferentes profundidades (un metro, tres metros y cinco metros) con sensores digitales de temperatura tipo sonda resistente al agua.

Se usará tres sensores digitales para la adquisición de temperatura, al ser digitales no se obtendrá degradación de la señal para las distancias establecidas en el proyecto. Estos sensores disponen de una precisión de $\pm 0,5$ ° C, un rango de trabajo de -55 a 125 ° C y una resolución de hasta 12 bits [1]. Como se observa los sensores tendrán una buena precisión, además de ser de bajo costo y muy fáciles de utilizar; ya que vienen en un envoltorio de acero inoxidable que lo hace resistente al agua.

Los datos de temperatura serán procesados por un computador de placa reducida Raspberry Pi, esta se encargará de procesar la información recolectada por los sensores y enviarla cada cierto tiempo de manera automática para poder ser mostrada por una página web. El intervalo de tiempo con el que se tomará los datos de temperatura dependerá del lugar en donde se instalará y probará el prototipo.

La alimentación del prototipo se hará a través de un sistema fotovoltaico con paneles solares y baterías que almacenen la energía. Cabe mencionar que como cualquier sistema deberá darse mantenimiento cada cierto tiempo.

Los datos serán enviados a través de la red celular. El computador de placa reducida Raspberry se conectará a la plataforma celular a través de un modem que permita dar por lo mínimo una conexión GPRS y podrá incrementar de tecnología de acuerdo a la cobertura del lugar de instalación.

Los datos podrán ser observados a través de una página web, donde se mostrarán los valores tomados de temperatura con su respectiva estampa de fecha y hora de una forma agradable al usuario; además, el usuario podrá interactuar de cierta manera con los datos observados.

Se cuidará que el prototipo este bien protegido para que no sufra daños por el ambiente para el que es diseñado (resistente al agua, resistente a temperaturas altas por el sol, etc.). Al tener una parte del prototipo en el agua (sensores) y otra parte fuera del agua (caja de comunicación y paneles solares), se deberá aislar el ingreso de agua a la caja de comunicación, ya que cualquier ingreso de agua causará humedad y corrosión en los elementos electrónicos que ahí se encontrarán.

El prototipo se probará en un ambiente controlado, para así verificar todas sus características que además de la adquisición y transmisión de datos de temperatura a una página web por medio de enlace celular, debe ser resistente al agua por el ambiente para el que será instalado. Se harán pruebas para verificar que no ingrese agua al armario de circuitos y que este aislado del agua.

Además se adecuará una estructura para poder probarlo en el ambiente controlado. Esta estructura deberá flotar y sostener todos los elementos del prototipo, además deberá ser fácil de colocar en el agua para tomar los datos de temperatura.

Finalmente, se busca que el prototipo sea de bajo costo, sin que esto implique usar elementos de bajas prestaciones, por lo contrario se buscará usar lo mejor que se pueda encontrar en el mercado para cumplir con los requerimientos del proyecto. Además que tenga una prolongada vida útil, tomando en cuenta que los sensores se deben cambiar cada cierto tiempo.

Si bien el prototipo se probará en un ambiente controlado, en un futuro se podrá usar en proyectos para masificar la información de la costa, ríos, lagos y lagunas ecuatorianas.

1.3 Marco Teórico

1.3.1 Parámetros medioambientales del agua

Los impactos en la calidad del agua significan cambios en los indicadores ambientales debido a acciones de una actividad o proyecto determinado. A modo de ejemplo, las descargas de aguas residuales de un proyecto industrial pueden disminuir la concentración de oxígeno en el agua, representando un impacto negativo en la calidad de sus aguas [2].

En el país se han establecido estándares para consumo humano, bebida de animales, vida acuática, uso recreativo y riego, además de aquellos establecidos para situaciones específicas como por ejemplo para descargas a sistemas de alcantarillado y residuos industriales.

La estimación de la calidad del agua, refiere a la manera más sencilla de definir ciertos parámetros y establecer límites que permitan definir categorías. Estas medidas del valor de los parámetros en la situación real y en otra situación que se considere admisible o deseable [3]. Existiendo así tres parámetros, físicos, químicos y biológicos, como se ilustra en la Figura 1.1.

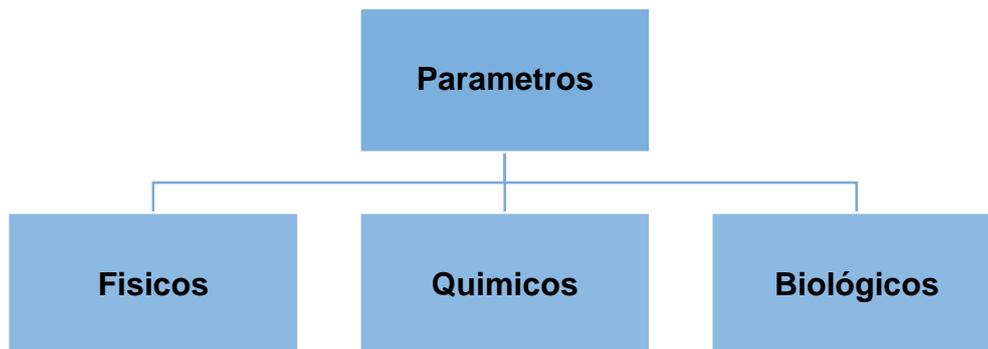


Figura 1.1. Clasificación de parámetros de la calidad del agua [2].

1.3.1.1 Parámetros Físicos

Los parámetros físicos se consideran como indicadores relativos, esto quiere decir que no son índices absolutos de contaminación.

De igual manera sus valores normales pueden variar considerablemente en cada caso; hay que medir la desviación respecto al valor normal. Los cambios pueden ser tan apreciables que un sólo parámetro llegue a dar una idea del grado de contaminación y de la extensión de la zona afectada [3].

Ahora bien, los principales parámetros físicos se muestran en la Figura 1.2.

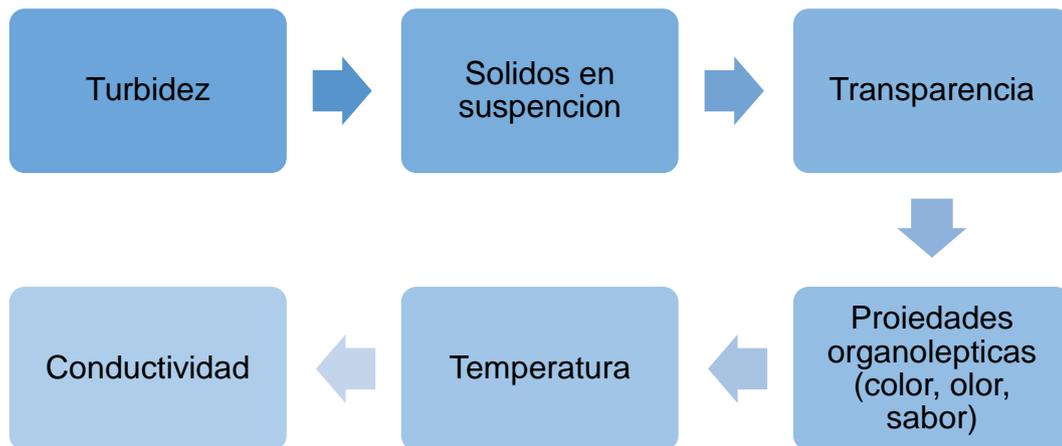


Figura 1.2. Parámetros físicos [3].

La temperatura del agua en los ríos presenta una considerable variabilidad, la cual se da por causas naturales y artificiales. Dentro de las causas naturales dependerá del clima de la zona y de las influencias del entorno. El estudio de su comportamiento ha llamado la atención de climatólogos, hidrólogos y varios expertos en el tema de cuidado y tratamiento de aguas superficiales [4] [5].

De igual manera la temperatura del agua afecta a la solubilidad de sales y gases, y en general a todas sus propiedades químicas y a su comportamiento microbiológico. La temperatura de las aguas superficiales está ligada a la absorción de radiación solar [6].

La temperatura del agua de un río puede dar una idea sobre las características medioambientales del agua, ya que la temperatura influye en elementos como el crecimiento de las plantas y el oxígeno disuelto [7].

- Las tasas de respiración de plantas y animales aumentan en aguas cálidas, así que usan el oxígeno más rápido.
- Organismos que requieren más oxígeno para su supervivencia pueden ser encontrados en aguas más frías donde hay más oxígeno disponible.

El aumento de las temperaturas de aguas dulces por el cambio climático origina el aumento de concentración de contaminantes en el agua. Debido a este cambio es necesario controlar la temperatura de los ríos y lagos, para así reducir los efectos negativos al ambiente.

La presencia de residuos en el agua sumada al cambio climático, acelera la producción de algas, plantas microscópicas perjudiciales para el sistema acuático porque consumen oxígeno para alimentarse, lo cual provoca la muerte de varias especies naturales que viven en las mismas aguas superficiales [4].

1.3.1.2 Parámetros Químicos

Los parámetros químicos permiten definir la calidad del agua, permiten identificar y cuantificar agentes causales de contaminación. Es por esto que se debe analizar los principales parámetros químicos mostrados en la Figura 1.3 [5].

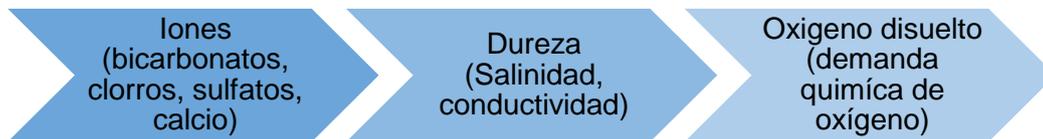


Figura 1.3. Parámetros Químicos.

Además, por el estudio o grado de contaminación, se pueden incluir varios parámetros extra los cuales se menciona a continuación.

- Compuestos de nitrógeno, fosfatos, hierro, cloro, azufre, pH.
- Fenoles, derivados del petróleo, detergentes, pesticidas.

1.3.1.3 Parámetros Biológicos

Al hablar de aguas superficiales se entiende por aquellas provenientes de los ríos, lagunas, lagos y mares. Dado que estas aguas se utilizan generalmente como fuentes de agua para el abastecimiento de las comunidades; estas están expuestas a una amplia gama de factores que pueden alterar la calidad del agua en diferentes niveles de intensidad y complejidad [2].

En base a esto se consideran parámetros biológicos a prácticamente todos los seres vivos acuáticos como bacterias, hongos, peces, etc. que viven en las aguas superficiales. Los principales parámetros biológicos se ilustran en la Figura 1.4.

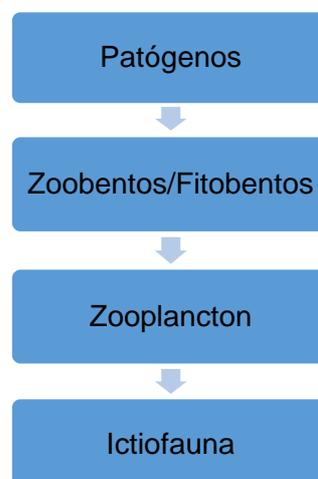


Figura 1.4. Parámetros Biológicos [3].

En varios ríos se puede encontrar una amplia gama de organismos no perceptibles a simple vista (Tabla 1.1). En condiciones normales, estos organismos permiten el desarrollo de los ciclos biológicos y químicos en el cuerpo de agua y no son necesariamente nocivos para la salud o para el tratamiento del agua.

Tabla 1.1. Organismos que forman parte de aguas superficiales

Algas	Son plantas de organización sencilla; existen formas unicelulares, coloniales y pluricelulares. Presentan clorofila.
Bacterias	Son seres de organización simple, unicelulares. Están distribuidas en una amplia variedad de sustratos orgánicos, tales como suelo, agua, polvo atmosférico. La mayor parte de bacterias son beneficiosas.
Protozoarios	Son organismos unicelulares, con una amplia distribución en los cuerpos acuáticos. La mayor parte de los protozoarios son beneficiosos para los ríos.
Insectos	En el agua generalmente se encuentran insectos en estado de larvarios y huevos de insectos. Estos conforman el zooplancton de forma temporal.

Se puede manifestar que la existencia de varios organismos de vida libre en condiciones normales es en cierta parte beneficiosa para las aguas superficiales; su presencia se convierte en dificultad al momento que existen en gran concentración y su composición altera la calidad del agua, presentándose así dificultades para el uso y tratamiento de esta.

1.3.2 Tecnologías para la transmisión de datos

Hoy en día los avances tecnológicos han mejorado la transmisión de datos, y cada vez se avanza en mejoras continuas perfeccionando las técnicas de transmisión.

Se puede mencionar que la electrónica actual conjuga los avances tanto del diseño digital como del analógico, por lo cual se puede afirmar que estas dos ramas, muy importantes e indispensables en la transmisión de datos, ofrecen soluciones para una gran diversidad de problemas.

La comunicación es la transmisión de datos de un punto a otro, para que esta comunicación se dé, es imprescindible de un equipo emisor, un mensaje, el canal por el cual se envían

los datos y el equipo receptor. Estos sistemas de telecomunicaciones ocupan un medio por el cual viajan las señales ópticas, eléctricas o electromagnéticas, durante la transmisión de datos [8].

Es a partir del siglo XIX que empieza el desarrollo acelerado de las telecomunicaciones, los mensajes se empiezan a transmitir a través de la corriente eléctrica, mediante el telégrafo primero y el teléfono después. Más adelante se desarrolló la comunicación a través de ondas electromagnéticas, que viajan a mayor velocidad que la corriente eléctrica, que no necesitan de cables para su transmisión y que se pueden transmitir en el espacio [9].

Según la naturaleza del canal por el que se transmiten la electricidad o las ondas, las comunicaciones pueden ser alámbricas o inalámbricas, como se muestra en la Figura 1.5.

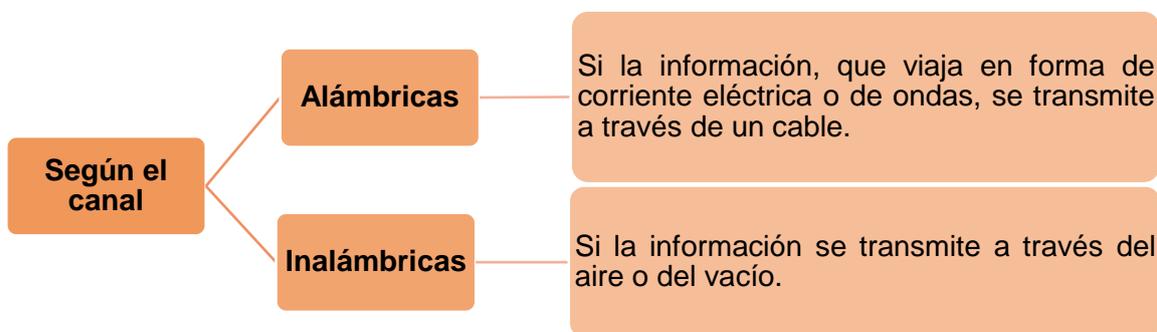


Figura 1.5. Transmisión de datos [8].

Los medios de transmisión alámbricos, también conocidos como medios físicos o guiados, deben hacerlo a través de ondas electromagnéticas en un cable. También hay cables que permiten la transmisión de luz (fibra óptica).

Los principales tipos de medios de transmisión alámbricos son:

- Cable de par trenzado
- Cable coaxial
- Cable de fibra óptica

Las características de los principales medios de transmisión guiados se muestra en la Tabla 1.2.

Mientras que los medios de transmisión inalámbrica, conocidos como medios no guiados, son aquellos que utilizan el espacio radioeléctrico para la transmisión de varios datos. Los principales tipos son:

- Radio
- Microondas
- Satélite

Las principales características de los medios no guiados se muestran en la Tabla 1.3.

Tabla 1.2. Características de medios de transmisión guiados.

	PAR TRENZADO	COAXIAL	FIBROPTICA
Tipo de Señal	Corriente Eléctrica	Corriente eléctrica	Luz (onda electromagnética)
Atenuación	Considerable	Intermedia	Muy escasa
Interferencia	Considerables	Escasas	Muy escasa
Ancho de Banda	Bajo/Intermedio	Alto	Muy alta
Coste	Bajo	Intermedio	Alta
Uso	Corta distancia	Media/larga distancia	Larga/muy larga distancia

Tabla 1.3. Características de medios de transmisión no guiados.

Medios no Guiados	Características	Alcance	Categorías	Ventajas	Desventajas
Radio	Ondas controladas por frecuencia	Distancia larga	VLF, VF, MF, HF, VHF	Cruzan obstáculos	Interferencia
Microondas	Onda unidireccional en línea recta	Distancia corta	Telefonía celular y larga distancia	Baratas	No cruzan obstáculos
Satélite	Comunicación entre estaciones terrestres	Sumamente larga	Televisión GPS Inteligencia militar	Área de cobertura Mayor ancho de banda	Costo elevado Mantenimiento Posicionamiento

1.3.2.1 El espacio radioeléctrico

Es un medio de comunicación que se basa en el envío de señales a través de ondas de radio. El espectro de radio es muy amplio, desde unos pocos Hz hasta ondas de frecuencias superiores en el orden de 10^{22} Hz, como se muestra en la Figura 1.6. El espectro radioeléctrico se divide en bandas, a cada una de las cuales se le asigna un nombre en función de su longitud de ondas. Las bandas que se utilizan habitualmente en las telecomunicaciones son las ondas de radio, las microondas, infrarrojos y luz visible. Pero existen otras bandas en el espectro, sobre todo en las altas frecuencias que cada vez se utilizan más en comunicaciones [10].

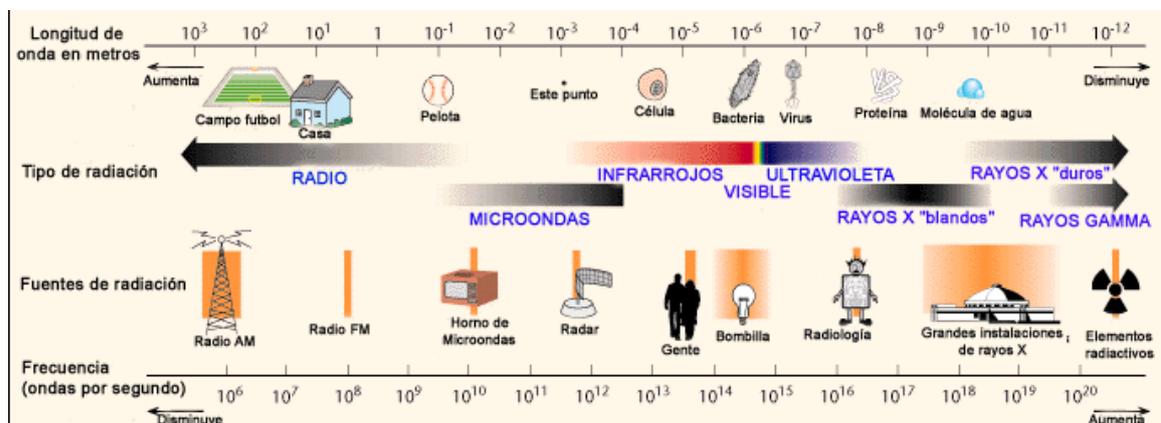


Figura 1.6. El espectro de frecuencias [10].

En la Tabla 1.4 se muestra las principales características de las bandas de frecuencia.

Tabla 1.4. Rango de Frecuencias de trabajo [10].

BANDA	FRECUENCIA	PROPAGACION	RANGO	USO
VLF	Muy Baja	Terrestre	3-30 kHz	Navegación Marítima
LF	Baja	Terrestre	30-300 kHz	Navegación y comunicaciones AM
MF	Media	Terrestre e ionosférica	300-3000 kHz	Radiodifusión AM
HF	Alta	Ionosférica	3-30 MHz	FM, TV, banda urbana
VHF	Muy alta	Directa de antena a antena	30-300 MHz	TV, radio FM

UHF	Ultra alta	Directa de antena a antena	300-3000 MHz	Tv, radar, comunicación por satélite
SHF	Microondas	Satélite	3-0 GHz	Radar, comunicación por satélite

1.3.2.2 Tecnología Celular

Las tecnologías móviles tienen mucho tiempo entre las personas simplificando las actividades cotidianas, facilitando trabajos, estudios o vida normal con sus innumerables aplicaciones disponibles que se han ido incrementando con el tiempo.

Actualmente, las tecnologías móviles han cubierto la mayoría de las áreas de servicio de comunicaciones y entretenimiento, enfocando sus aplicaciones a generar un mercado cautivo a los miles de usuarios que día a día adoptan el uso de servicios, tal como lo es, telefonía móvil, envío de mensajes de texto y multimedia; y en los últimos años la actualización de la información de las redes sociales [10].

Para la evolución de las tecnologías utilizadas en la telefonía móvil se emplea el concepto de “generación”, de forma que cada generación engloba un conjunto de estándares de transmisión de datos que ofrecen unas determinadas prestaciones y calidad de servicio como se muestra en la Tabla 1.5.

Tabla 1.5. Evolución de tecnología celular [9].

1G	Fue la primera generación de telefonía móvil y utilizaba tecnología analógica para la transmisión de información. Se utilizó en los años 80.
2G	Es la segunda generación de telefonía móvil que utiliza GSM la cual permite la transmisión tanto de voz como de datos.
2.5 G (GPRS)	Una extensión de GSM para la transmisión de datos. Permite velocidades de transferencia teóricas de 56 a 144 Kbps. En la práctica son unos 40 Kbps.
3G	Mejora significativamente la capacidad máxima de transferencia de datos, pudiéndose alcanzar tasas de descarga de hasta 14 Mbps.

4G	Cuarta generación de tecnologías de telefonía móvil, basada completamente en el protocolo IP. Esta tecnología puede ser usada por módems inalámbricos, móviles inteligentes y otros dispositivos móviles.
-----------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

En el campo de la electrónica se requiere frecuentemente conectar distintos equipos y sistemas. Para tal objeto, se emplean unidades remotas de transferencia de datos, las cuales son interconectadas. Sin embargo, hoy en día es posible emplear la red de telefonía móvil, la que ya cuenta con la tecnología adecuada para transmisión de datos. Ahora cualquier usuario de telefonía celular puede conectarse a Internet, enviar y recibir mensajes en cualquier momento [11].

➤ **Tecnología Móvil 3G (Tercera Generación)**

La tecnología de tercera generación, también llamada 3G es considerado como un servicio de comunicaciones inalámbricas, el cual permite conectarse de forma permanente a internet a través de un dispositivo móvil u ordenador portátil. La tecnología 3G propone una mejor calidad y efectividad en cuanto a la transmisión de datos, para lo cual presenta un ancho de banda superior. Este sistema presenta velocidad de datos de hasta 384 Kbps, siendo aproximadamente siete veces más rápida que una conexión telefónica estándar [12].

Esta generación está caracterizada por la convergencia de voz y datos con acceso inalámbrico a internet, esto quiere decir que se ofrece aplicaciones multimedia y altas transmisiones de datos. En relación con las generaciones anteriores a 3G propone mayores ventajas, este combina el acceso móvil de alta velocidad con los servicios basados en el protocolo IP. Esto quiere decir que hace referencia a poseer una conexión rápida a internet, por ende esto indica un ancho de banda elevado. Los usos que esta generación ofrece son grandes, ya que permite realizar transacciones bancarias a través de un teléfono móvil, hacer compras, consultar información, entre otros [13].

Los sistemas 3G cumplen varios requerimientos, entre los cuales se encuentran:

- Velocidad de bit mayor a 2 Mbps.
- Múltiples servicios con diferentes requerimientos de calidad en una sola conexión.
- Coexistencia de sistemas de segunda y tercera generación para aumentar cobertura y balanceo de carga.

- Alta eficiencia espectral [13].

De igual manera los sistemas 3G se han diseñado para la comunicación multimedia, con ello la comunicación entre personas puede ser aumentada con imágenes de alta calidad y video, además acceder a información y servicios en redes públicas y privadas.

Los servicios multimedia integran varios componentes como lo son la voz, datos, audio, video, imágenes, etc. dentro de una llamada. Para lo cual es necesario que exista una flexibilidad para añadir o eliminar recursos [14].

Según la UIT (Unión Internacional de Telecomunicaciones), las redes 3G utilizan una interfaz aérea CDMA (*Code Division Multiple Access*, Acceso Múltiple por División de Código), es por ello que ofrece velocidades mucho más altas de datos en aparatos inalámbricos portátiles. De igual manera se acopla con un sistema UMTS (*Universal Mobile Telecommunications System*, Sistema Universal de Telecomunicaciones Móviles), el cual pertenece a un sistema por capas.

La familia que incorpora los sistemas de tercera generación está definida por la UIT en el cual se desarrollan tres tipos de tecnologías de acuerdo a la zona geográfica, para lo cual la institución menciona tres [15].

- CDMA-2000
- UMTS
- TD-SCDMA (*Time Division Synchronous Code Division Multiple Access*, Acceso Múltiple por División de Código Síncrono de División de Tiempo).

Además, dentro de la generación 3G surgieron algunas evoluciones como:

- HSDPA (*High Speed Downlink Packet Access*, Acceso Descendente de Paquetes a Alta Velocidad)
- HSUPA (*High Speed Uplink Packet Access*, Acceso Ascendente de Paquetes a Alta Velocidad)
- HSPA+ (*High Speed Packet Access Plus*, Acceso de Paquetes a Alta Velocidad Evolucionado)

HSPA+ y LTE (*Long Term Evolution*, Evolución a Largo Plazo) ofrecen tasas de datos y capacidad similares cuando se utiliza la misma configuración de antena y la misma cantidad de ancho de banda, ya que ambas tecnologías dependen de las mismas mejoras.

Además, tanto HSPA+ como LTE aprovechan de la infraestructura y dispositivos de las operadoras [16]. En la Figura 1.7 se muestran las velocidades de las redes móviles 3G.

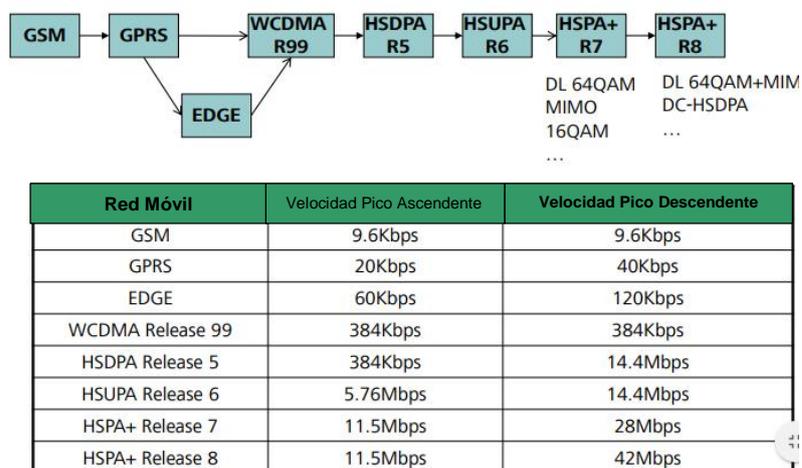


Figura 1.7. Evolución de la velocidad de datos [16].

De igual manera existen bandas de frecuencia de operación que han sido asignadas a las operadoras en Ecuador. Existen tres empresas que prestan sus servicios en el Ecuador: CNT, CONECEL S.A. conocida como Claro y OTECEL S.A. conocida como Movistar. CNT tuvo que alquilar la red de Movistar para pasar de la tecnología CDMA a GSM (*Global System for Mobile Communications*, Sistema Global para Comunicaciones Móviles), con el paso de los años cuenta con su propia red 3G. En la Tabla 1.6 se encuentran las bandas de frecuencia asignadas a estas operadoras.

Tabla 1.6. Bandas 3G de las operadoras de telefonía móvil en Ecuador [17].

Operadora	GSM MHz	3G MHz
CNT	-	1900
Claro	850, 1900	850, 1900
Movistar	850,1900	850, 1900

➤ Tecnología móvil 4G (Cuarta Generación)

Los avances de la tecnología sobre la telefonía celular enmarcan un mercado asiduo de servicios de valor agregado, capaces de superar los índices de penetración a partir de la segunda década del presente siglo.

La tecnología 4G se soporta en el estándar 3GPP (*3rd Generation Partnership Project*, Proyecto de Asociación para la Tercera Generación) que basa su sistema en IP, es decir es un sistema de sistemas y una red de redes, superándose posteriormente en la convergencia entre redes de cable o redes inalámbricas, ordenadores, dispositivos eléctricos-electrónicos entre otros para proveer de velocidades de acceso entre 100Mbps en movimiento y 1Gbps en reposo. Una de la ventaja que posee LTE consiste en que será capaz de ofrecer velocidades de descarga hasta 60Mbps y envíos de paquetes de información hasta 40Mbps.

En lo referente a servicios multimedia, LTE opera en un espectro mucho más flexible y mayor que tecnologías anteriores entre 1,4 y 20Mhz, las pruebas han resultados satisfactorias en la realización de llamadas y video llamadas entre terminales LTE, pero sobre todo las convergencias entre las redes actuales y la nueva red de cuarta generación, logrando como prueba de laboratorio una video llamada satisfactoria entre un teléfono en red LTE y un terminal en red 3G [18].

Varios de los usos que brinda esta tecnología parte de la capacidad de conexión inalámbrica a internet con velocidades que superan las normales redes 3G, VoIP (*Voice Over Internet Protocol*, Voz Sobre Protocolo de Internet), siendo uno de los usos más novedosos de la tecnología 4G, de igual manera televisión móvil en alta definición, juegos online, etc. [19].

Las frecuencias o bandas 4G LTE que usan Movistar, Claro, CNT y Tuenti en Ecuador se indican en la Tabla 1.7.

Tabla 1.7. Bandas 4G LTE en Ecuador [15].

Movistar	1900 MHz (Banda 2)
Claro	1700/2100Mhz (AWS o Banda 4)
CNT	1700/2100Mhz (AWS o Banda 4) o también 700MHz (APT o Banda 28)
Tuenti	1900MHz (Banda 2)

Una evolución de 4G es LTE-A (*Long Term Evolution Advanced*, Evolución a Largo Plazo Avanzada), que ha sido diseñada para permitir internet de alta velocidad y banda ancha en cualquier momento, en cualquier lugar [20].

Según el ITU, la tecnología LTE-A llega a velocidades máximas de datos de hasta 100 Mbit/s de alta movilidad y hasta 1 Gbit/s para escenarios de baja movilidad.

La tecnología LTE-A se centra en proporcionar elevadas tasas de bits de una manera rentable y, al mismo tiempo, cumplir con los requisitos establecidos por la UIT para la tecnología 4G. Las mejoras de la tecnología LTE-A con respecto a LTE se presentan a continuación:

- Velocidad máxima de datos aumentada DL 3 Gbps, UL 1.5 Gbps.
- Mayor eficiencia espectral.
- Mayor cantidad de suscriptores activos simultáneamente
- Rendimiento mejorado en los bordes de la celda, para DL 2x2 MIMO (*Multiple Input Multiple Output*, Múltiple Entrada Múltiple Salida) al menos 2,40 bps / Hz / celda [21].

En la Tabla 1.8. se muestra las principales características de LTE-A.

Tabla 1.8. Características de LTE-A [22].

Agregación de portadoras ¹ para mayor soporte de ancho de banda hasta 100 MHz mediante la agregación de bloques de 20 MHz, esto origina a las operadoras usar bandas de espectro diferentes.
Mejor funcionabilidad y capacidad de red mediante tecnología MIMO.
Permite una transmisión multipunto coordinada.
Brinda una velocidad de transmisión de datos de 1,2 Gbps en <i>downlink</i> y 568 Mbps en <i>uplink</i> .

1.3.2.3 Comunicaciones Satelitales

Una solución para la movilidad y rapidez en la transmisión de datos son las comunicaciones satelitales, que al pasar los años han avanzado a grandes pasos, satisfaciendo las necesidades y requerimientos de las personas.

El satélite es un dispositivo que actúa como reflector de las emisiones terrenas (Figura 1.8.), reflejando un haz de microondas que transporta información codificada. La función

¹ Agregación de portadoras es una funcionalidad de LTE-A que permite crear mayores anchos de banda al combinar diferentes asignaciones de espectro [22].

de reflexión se compone de un receptor y un emisor que operan a diferentes frecuencias. [23].



Figura 1.8. Satélite.

Las antenas parabólicas son usadas en comunicaciones satelitales, se caracterizan por tener un plato parabólico llamado así por su superficie paraboloide de revolución. Las antenas pueden ser transmisoras, que en ese caso llevan un BUC (*Block Up-Converter*, Conversor de Bloque de Subida), receptoras si llevan un LNB (*Low Noise Block*, Bloque de Bajo Ruido) o full dúplex cuando transmiten y reciben simultáneamente. En la figura 1.9 se ilustra una antena parabólica [24].



Figura 1.9. Antena parabólica.

Un BUC también denominado Bloque de Transmisión o TXB, es un dispositivo utilizado en la transmisión de enlace ascendente (*uplink*) de señales de comunicación vía satélite. Actúa de interfaz convirtiendo a la banda de frecuencias de la antena parabólica (típicamente desde la L hasta la Ka) las señales de frecuencia de los equipos conectados al módem satelital [24] [25]. En la Figura 1.10 se muestra el ejemplo de un BUC.



Figura 1.10. BUC.

Un LNB es un dispositivo que se sitúa en el foco de la antena parabólica, el cual se encarga de adaptar la señal recibida del satélite y repartirla, mediante cable coaxial hacia toda la instalación. Debido a que las frecuencias de enlace descendente (*downlink*) son difíciles de transmitir por cable coaxial por su alta frecuencia, es necesario instalar este dispositivo que adapte las frecuencias a niveles óptimos para el cable coaxial. En la Figura 1.11 se muestra un LNB [26].



Figura 1.11. LNB universal.

Un haz de microondas se transmite al satélite desde la superficie terrestre, este es recibido por el transponder del satélite el cual lo retransmite a la estación destino. El haz de la señal emitida por el satélite puede ser [23]:

- Ancho, para que pueda ser captado en un área extensa
- Fino, para que solo pueda captarse en un área limitada

Dentro de las diferentes funciones encargadas a los sistemas satélites como investigación, operaciones militares, comunicaciones móviles, etc. están las funciones relacionadas al medioambiente y clima. Los satélites meteorológicos proveen información sobre las condiciones atmosféricas, temperatura de mares y el clima de grandes áreas geográficas de la tierra. Hay dos tipos de satélites meteorológicos, de órbita geoestacionaria y satélites de órbita polar. Las características más importantes de estos dos tipos de satélites se muestran en la Tabla 1.9.

Tabla 1.9. Características de los tipos de satélites [27].

Satélite	Características
De órbita Geoestacionaria	<ul style="list-style-type: none"> - Recorren su órbita a una altura de 36000 km. - Buena cobertura sobre el Ecuador. - Completan una vuelta sobre la órbita en 24hrs, por eso siempre están sobre un mismo punto. - Los satélites meteorológicos que describen este tipo de órbita son: GOES (<i>Geostationary Operational Environmental Satellite</i>), METEOSAT (Europa),

	GMS (Japón).
De órbita Polar	<ul style="list-style-type: none"> - Se trasladan de polo a polo a alturas que oscilan entre 850 y 1500 Km. - Demora aproximadamente 100 minutos en completar una órbita alrededor de la Tierra. - Cada vez que completan una vuelta, la Tierra ha rotado 25° de longitud, por lo que monitorizan una franja diferente en cada órbita. - Los satélites de órbita polar más conocidos son los de la serie NOAA (<i>National Oceanic and Atmospheric Administration</i>).

1.3.3 Sistemas fotovoltaicos con paneles solares y baterías

La tecnología solar fotovoltaica consiste en la conversión directa de la radiación del sol en electricidad. Esta conversión se realiza a través de una célula solar, unidad básica en la que se produce el efecto fotovoltaico. La energía solar fotovoltaica está indicada para un amplio abanico de aplicaciones donde se necesite generar electricidad, bien sea para satisfacer necesidades energéticas de aquellos que no disponen de la red eléctrica o bien para generar energía a la red eléctrica [28]. Los sistemas fotovoltaicos se clasifican en función de su conexión a la red eléctrica convencional como se muestra en la Figura 1.12.

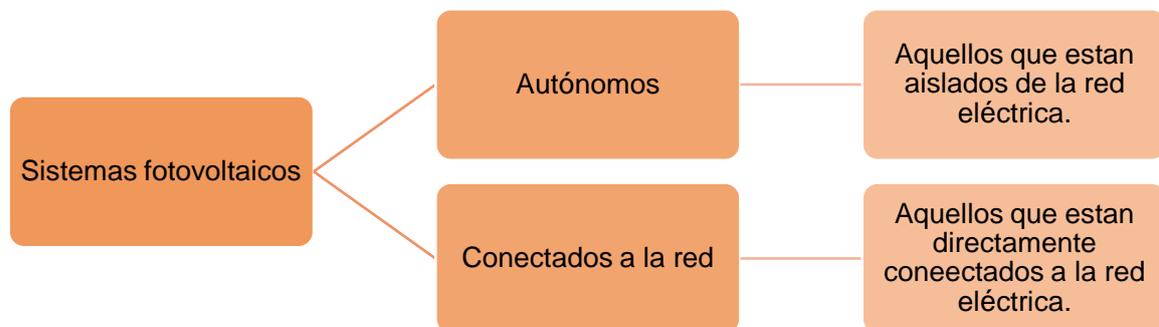


Figura 1.12. Clasificación de los sistemas fotovoltaicos [28].

1.3.3.1 Módulos fotovoltaicos

Están compuestos de células capaces de convertir la luz en electricidad. Todas las células del módulo están unidas entre sí, para poder sumar su potencia y alcanzar conjuntamente la potencia nominal del módulo, en la Figura 1.13 se ilustra un panel fotovoltaico [29].

Si se conectan varios módulos entre sí uniendo todos los polos negativos por un lado y todos los polos positivos por el otro, en los extremos se tiene el mismo voltaje y una

intensidad equivalente a multiplicar la intensidad unitaria de los módulos por el número de éstos conectados entre sí. En la Figura 1.14 se muestra un sistema fotovoltaico [28].



Figura 1.13. Panel fotovoltaico.

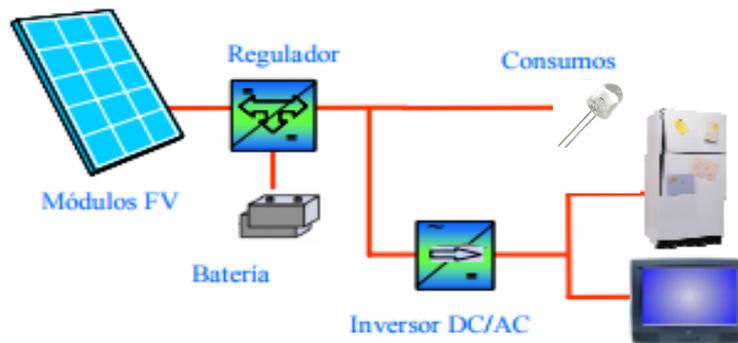


Figura 1.14. Estructura básica de un sistema fotovoltaico.

1.3.3.2 Baterías

En los sistemas fotovoltaicos, las baterías se utilizan como sistema de almacenamiento emergente, debido al desplazamiento temporal que puede existir entre los periodos de generación durante el día y periodos de consumo durante la noche.

Las horas del día en que los módulos fotovoltaicos generan electricidad corresponden al período en que el consumo suele ser más bajo, y en cambio en la noche, cuando no se produce electricidad, suele haber el consumo más elevado. Por tanto, hay que acumular la energía, y las baterías son una manera bastante eficiente de hacerlo [28]. En la Figura 1.15 se muestra una batería fotovoltaica.



Figura 1.15. Batería Fotovoltaica.

1.3.4 Conceptos para la creación de páginas web

Para la creación del sitio web es necesario tener claro ciertos conceptos que se muestran en la Tabla 1.10.

Tabla 1.10. Conceptos sobre página web [30].

Página web	Está compuesta por uno o varios documentos HTML (<i>HyperText Markup Language</i> , Lenguaje de Marcas de Hipertexto) relacionados entre sí mediante hipervínculos. Además, estos documentos pueden contener otros elementos como pueden ser imágenes, sonidos, animaciones multimedia, aplicaciones, etc.
Sitio web	Contiene las páginas web, gráficos, documentos, multimedia y otros archivos asociados que se almacenan en un servidor Web.
Dominio	Dirección web asociada a una página web.
URL	Cadena que proporciona la dirección de Internet de un sitio Web o un recurso del World Wide Web.
Servidor Web	Un servidor web es un programa informático que procesa una aplicación del lado del servidor, realizando conexiones bidireccionales o unidireccionales con el cliente y generando una respuesta en cualquier lenguaje o aplicación del lado del cliente [31].

1.3.4.1 HTML (*HyperText Markup Language*, Lenguaje de Marcado para Hipertextos)

HTML es un mecanismo básico de elaboración de una página web y se emplea para crear y representar visualmente una página. Establece el contenido de la página web, pero no su funcionalidad. HTML enseña a trabajar con el lenguaje de marcación que sirve para elaborar las páginas web. HTML es el primer paso que se debe realizar para dedicarse al desarrollo de web en general. Es además un conocimiento recomendado para cualquier persona que trabaje en el medio de Internet [32].

1.3.4.2 PHP (*Hypertext Preprocessor*, Procesador de Hipertexto)

PHP permite embeber minúsculos fragmentos de código dentro de una página creada con HTML. Esos scripts PHP permiten ejecutar actividades de una forma fácil y eficaz, realizando una variedad de tareas. Esta mezcla de PHP dentro del marco de un documento HTML es lo que facilita a los desarrolladores sin mucha experiencia en el campo, crear actividades atractivas sin mucho conocimiento. En resumen, con PHP se escribe scripts dentro del código HTML. Al estar familiarizado con HTML, iniciar el desarrollo con PHP es sencillo. PHP ofrece una variedad de funciones para la utilización de los recursos, entre estos están las bases de datos a las que se puede acceder sin problemas [33].

1.3.4.3 Base de datos

Las bases de datos permiten almacenar de una forma estructurada y eficiente toda la información de un sitio web. Además de almacenar información, ofrece otras herramientas más propias de la gestión de páginas web como sistemas de permisos para autorización de accesos, seguridad, transferencia de ficheros, recuperación de información dañada, indización, etc. En la Figura 1.16 se muestra el esquema de un sitio web con bases de datos [34].

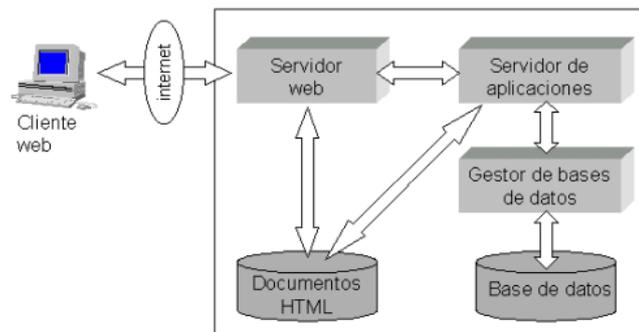


Figura 1.16. Esquema de un sitio web soportado por bases de datos [34].

El gestor de base de datos que se usará es MySQL (*My Structured Query Language*, Lenguaje de Consulta Estructurado), que es la base de código abierto más popular gracias a su rendimiento, confiabilidad y facilidad. MySQL es la principal elección de base de datos para aplicaciones basadas en la Web, utilizada por propiedades web de alto perfil como Facebook, Twitter, YouTube. Además, es una alternativa extremadamente popular como base de datos integrada [35].

Características

- Escrito en C y en C++.

- Probado con un amplio rango de compiladores diferentes.
- Funciona en diferentes plataformas.
- APIs disponibles para C, C++, Eiffel, Java, Perl, PHP, Python, Ruby, y Tcl.
- Un sistema de reserva de memoria muy rápido basado en *threads*.
- Tablas *hash* en memoria, que son usadas como tablas temporales [35].

1.3.4.4 Python

Python es un lenguaje de programación muy usado ya que es de fácil aprendizaje y uso; la sintaxis junto a la interpretación fácil hacen de este lenguaje ideal para el desarrollo de aplicaciones en diversas áreas y su compatibilidad hace que se use en la mayoría de plataformas.

Python con sus extensas bibliotecas están libres y a disposición para las principales plataformas. El uso de este lenguaje se hizo muy conocido gracias a su filosofía que resalta una sintaxis que favorezca un código legible, así usuarios de poco o bastante conocimiento podrán usar esta herramienta [36].

2. METODOLOGÍA

2.1 Plataformas de hardware libre.

El hardware libre tiene como objetivo generar diseños de aparatos informáticos de forma abierta, de manera que el público tenga acceso, como mínimo, a los planos de construcción de los dispositivos. Esta actividad no es algo de la actualidad, de hecho aparece en la década de los años 70, cuando algunos aficionados a las computadoras elaboraban sus propios equipos comprando las partes a diferentes lugares y diseñaban sus propios prototipos [37].

El diseño está al alcance de cualquier persona de tal modo que cualquiera puede aprender, modelar, distribuir, elaborar y distribuir el diseño o el hardware que se establezca en el diseño. Hardware libre utiliza partes y elementos de fácil adquisición y con los que se tiene el acceso a procesos estandarizados, infraestructura abierta, contenido no restringido y herramientas de diseño libres posibilitando el controlar la tecnología mientras se comparte el conocimiento y benefician el comercio intercambiando los diseños [38].

A continuación se analizarán las principales plataformas de hardware libre que se pueden encontrar en el mercado ecuatoriano y que se pueden usar en el proyecto.

2.1.1 Cubieboard

La Cubieboard es un ordenador de pequeñas dimensiones que puede ser conectado al monitor, teclado y a un mouse para configurar y usar. Puede realizar muchas de las actividades que realiza un computador de escritorio, como de hojas de cálculo, editar texto y variedad de juegos. En la Figura 2.1 se muestra el aspecto físico de la placa [39].

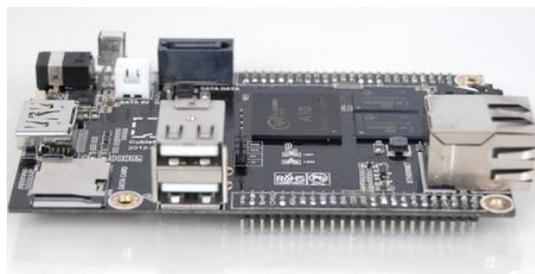


Figura 2.1. Cubieboard.

Este hardware permite la conexión de un disco duro SATA (*Serial Advanced Technology Attachment*) de hasta 2TB que incrementa las posibilidades de un proyecto; además cuenta con almacenamiento interno de 4GB en los que viene instalado de serie Android. Si se

quiere otro sistema operativo (Lubuntu, Fedora, Debian, Cubian, Buildroot, etc.) se puede instalar mediante una tarjeta microSD [39].

Ya que cuenta con un sistema operativo para realizar sus tareas, es factible descargar una variedad de aplicaciones, además es posible interactuar con tarjetas Arduino [40].

Existen cinco versiones de la placa: Cubieboard 1, Cubieboard 2, Cubietruck, Cubieboard 4 y Cubietruck Plus. En el mercado ecuatoriano se puede conseguir hasta la Cubietruck (Cubieboard3); las características de esta placa se muestran en la Tabla 2.1.

Tabla 2.1. Características Cubieboard 3 [41].

CPU	Procesador AllWinnerTech, ARM® Cortex™-A7 Dual-Core
Memoria RAM	1GB DDR3 a 480MHz
Almacenamiento	Ranura microSD Conector SATA 2.5 hasta 2TB
Alimentación	Entrada 5V a 2A o entrada USB OTG
Consumo	12.5 W
Puertos	Puerto Ethernet Puerto HDMI 2 puertos USB 2.0 y 1 puerto USB OTG
Otros	Receptor infrarrojo Conector de audio 96 pines para conexión de interfaces como: I2C, VGA, entre otras.

2.1.2 Intel Galileo

Son placas de desarrollo que permiten la realización de varios proyectos gracias a las prestaciones y características que presenta la placa. Galileo es una tarjeta electrónica, basada en un microcontrolador Intel Quark SoC X1000 que puede considerarse como un sistema Pentium de 32 bits, unido en un solo chip. La arquitectura de este chip tiene compatibilidad con *shields* para Arduino, de nivel pin a pin (Figura 2.2) [42].



Figura 2.2. Intel Galileo.

Las características de la placa Intel Galileo se presentan en la Tabla 2.2.

Tabla 2.2. Características Intel Galileo [43].

Voltaje de Entrada (recomendado)	5V
Voltaje de Entrada (límite)	5V
Pines de E/S Digital	14 (6 proveen salida PWM)
Pines de Entrada Análoga	6
Corriente DC Total de Salida en todos los pines de E/S	80 mA
Corriente DC para pin de 3.3V	800 mA
Corriente DC para pin de 5V	800 mA

2.1.3 Arduino

Es una placa con un microcontrolador creada con el fin de diseñar proyectos. Arduino es utilizado para desarrollar elementos autónomos, manteniendo conexión a diferentes dispositivos y a su vez, es posible interactuar tanto con el hardware como con el software [44].

Las placas Arduino tienen un menor costo comparándolas con otras plataformas microcontroladoras. La versión más barata tiene la posibilidad de ser ensamblada a mano.

Entre las características generales se tienen las siguientes [45]:

- **Multiplataforma:** El software se puede ejecutar con Windows, Macintosh OSX y GNU/Linux.
- **Entorno de programación simple y clara:** Arduino es fácil de instalar aun para principiantes, pero tienen flexibilidad para profesionales.
- **Código abierto y software extensible:** Arduino se establece con herramientas de código abierto, tiene una disponibilidad para programadores con experiencia. El lenguaje que utiliza es C++, y las personas que desean más detalles técnicos pueden ingresar a la página de Arduino para verificar el lenguaje de programación AVR-C.

- Código abierto y hardware extensible: Establecido en microcontroladores ATMEGA8 y ATMEGA168 de Atmel. Los planos para los módulos tienen licencia *Creative Commons*², por lo cual diseñadores con experiencia en la creación de circuitos pueden hacer su propia versión del módulo, extendiéndolo y mejorándolo.

2.1.3.1 Arduino Uno

Con Arduino Uno se puede desarrollar módulos o realizar una amplia variedad de proyectos de manera fácil. La placa Arduino Uno puede acceder a la información del entorno mediante sus entradas analógicas y digitales con la posibilidad de controlar elementos como luces, motores y otros actuadores. Tiene 6 pines de entrada analógica y tiene compatibilidad con campos como el Bluetooth, infrarrojo y otros sensores. Se muestran las características de la placa en la Tabla 2.3 [46].

Tabla 2.3. Característica Arduino Uno [46].

Microcontrolador	ATmega328P
Alimentación	5 - 12 Vcc
Frecuencia de operación	20 MHz
Puertos	6 puertos análogos de entrada 13 puertos de entrada/salida digital (incluyendo PWM).
Capacidad de memoria flash	32 KB
SRAM	2 KB
EEPROM	1 KB

2.1.3.2 Arduino Mega

Es el microcontrolador más capaz de la familia Arduino, cuenta con 54 pines digitales que trabajaban como entrada/salida; 16 entradas análogas, un cristal oscilador de 16 MHz, conectividad a USB, un botón de reset y un acceso para la alimentación de la placa. La conexión entre la computadora y Arduino se establece mediante del puerto serie. Posee un convertidor USB-serie, por lo que solo se necesita utilizar un cable USB para conectarlo. En la Figura 2.3 se muestra la placa Arduino Mega [47].

² Licencia *Creative Commons* es una organización que ofrece modelos de licencias libres que permiten a los autores poner sus obras de forma libre en internet [47].

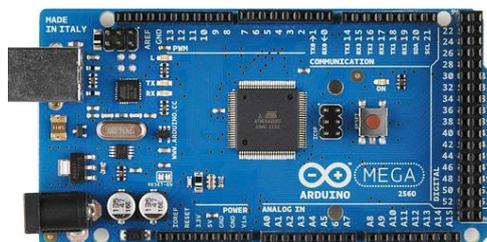


Figura 2.3. Arduino MEGA.

2.1.4 Raspberry Pi

La Raspberry Pi es un micro computador o una placa de ordenador SBC (*Single Board Computer*, Computadora de Placa Única) de precio económico perfeccionada en el Reino Unido por la Fundación Raspberry Pi, de tal manera que sirve para promover la educación de la computación y programación en los centros de enseñanza y con disponibilidad para cualquier persona. Posee un micro procesador ARM con potencia que llega a 1GHz, integrado en un chip Broadcom BCM2835. Tiene además 512 MB de RAM, un GPU Videocore IV, además de se pueden ejecutar programas básicos, utilizar el internet y también programar [48].

Es una placa de las dimensiones de una tarjeta de crédito que puede ejecutar varias distribuciones de Linux y otros sistemas operativos, así es útil para proyectos electrónicos ya que provee acceso a sus periféricos de bajo nivel y con la capacidad de hacer muchas de las tareas que realiza una computadora de escritorio, como ejecutar hojas de cálculo, editores de texto o juegos. Puede establecer conexión a una pantalla con conector HDMI (*High-Definition Multimedia Interface*, Interfaz Multimedia de Alta Definición) o a una televisión a través del conector RCA, además se puede conectar teclado, mouse y cámara. Algunas versiones cuentan con Wi-Fi y Bluetooth incorporados. Contiene un puerto USB 2.0, pero pueden expandirse mediante un hub con alimentación propia [48].

También cuenta con una ranura para microSD para almacenar el sistema operativo y programas. Los sistemas operativos soportados pueden ser: Debian (Raspbian), Fedora (Pidora), Arch Linux (Arch Linux ARM), Slackware Linux, Xbian, QtonPi, RISC OS, entre otros.

La alimentación puede hacerse por puerto micro USB o por conector de pines GPIO (*General Purpose Input/Output*, Entrada/Salida de Propósito General). Además los lenguajes de programación pueden ser Python, C, Java, Perl, BBC BASIC. [49]. Los modelos que se puede encontrar de la Raspberry Pi están en la Tabla 2.4.

Tabla 2.4. Modelos Raspberry Pi.

Modelos Raspberry Pi						
Raspberry Pi 1 Modelo A	Raspberry Pi 1 Modelo B	Raspberry Pi 1 Modelo B+	Raspberry Pi 2 Modelo B	Raspberry Pi 3 Modelo B	Raspberry Pi Zero	Raspberry Pi Zero W

Los modelos más usados y de mejores prestaciones son Raspberry Pi 2 Modelo B y Raspberry Pi 3 Modelo B, que se describirá a continuación.

2.1.4.1 Raspberry Pi 2 Modelo B

Este miniordenador permite trabajar con solo conectarlo a una pantalla y un teclado. La placa Raspberry 2 sólo pesa 45 gramos. El procesador ARM del Raspberry Pi 2 trabaja a 900 MHz y contiene un procesador gráfico con 1GB de RAM. Debido a su potencia tiene una compatibilidad con Linux y Windows 10. Se puede conectar a la red a través de su conexión RJ45 y se puede utilizar como servidor web o de correo. La entrada HDMI te permite reproducir videos en alta definición y es compatible con altavoces y auriculares mediante conector Jack 3.5 integrado. En la Tabla 2.5 se muestran las principales características de la placa [50].

Tabla 2.5. Características Raspberry Pi 2 Modelo B [50].

Procesador	Quad-Core Cortex A7 a 900MHZ
Memoria RAM	1GB
Interfaz	1 Interfaz de cámara (CSI) 1 Interfaz de Pantalla (DSI)
Puertos	4 USB 2.0 40 GPIO pin 1 HDMI 1 Ethernet 1 Combo audio/mic 1 Micro SD 1 Núcleo Grafico 3D

2.1.4.2 Raspberry Pi 3 Modelo B

El modelo Raspberry Pi 3 Modelo B incluye en su circuito un chipset Broadcom BCM2387 con cuatro núcleos ARM Cortex-A53 a 1.2 GHz. El procesador ejecuta videojuegos y

aplicaciones sin molestias, además de tener una gran potencia para correr diferentes tipos de programas. La GPU encargada de los gráficos es la Broadcom VideoCore IV, una solución Dual Core con compatibilidad para Open GL ES 2.0 y OpenVG que permite una resolución de Full HD. La capacidad le permite ejecutar Windows 10 y la distribución Ubuntu Snappy, versiones Slim [50]. Las características de la placa se encuentran en la Tabla 2.6.

Tabla 2.6. Características Raspberry Pi 3 Modelo B [50].

Procesador	1,2 GHz de cuatro núcleos ARM Cortex-A53
Memoria RAM	1GB LPDDR2.
Ethernet socket	Ethernet 10/100 BaseT
Wireless	LAN inalámbrica 802.11 b / g / n Bluetooth 4.1 (classic Bluetooth y LE)
Salida de vídeo	HDMI rev 1.3 y 1.4 RCA compuesto (PAL y NTSC)
Salida de audio	Jack de 3,5 mm de salida de audio, HDMI
USB	4 Conector USB 2.0
Conector GPIO	27 pines GPIO, así como 3,3 V, +5 V y GND líneas de suministro.
Conector de la cámara	15 pines cámara MIPI interfaz en serie (CSI-2)
Pantalla de visualización	Conector de la interfaz de serie (DSI) Conector de 15 vías plana flex cable con dos carriles de datos y un carril de reloj.
Ranura de tarjeta de memoria	Memoria empuje / tire MicroSD

2.1.5 Selección de la plataforma de hardware libre

Para el proyecto se debe elegir una plataforma de hardware libre que sea compatible con los sensores y periféricos que se usarán, también que el consumo de energía sea razonable. Además que permita procesar archivos de configuración sin mayor esfuerzo.

Una vez mencionadas en las secciones anteriores las plataformas de hardware libre más conocidas y que se puede encontrar en el mercado ecuatoriano, se selecciona la plataforma más adecuada que cumpla con los requerimientos del proyecto y que cumpla

con ciertos criterios técnicos como consumo de potencia, capacidad de memoria, conectividad y precio.

El consumo de energía es un parámetro importante a tomar en cuenta, sin embargo es difícil obtener, ya que dependerá de los periféricos que se conecten y el tiempo de uso o reposo al que trabajen. En la Tabla 2.7 se muestran los valores de consumo de las plataformas de hardware libre con sus respectivos modelos que se encuentra en el mercado.

Tabla 2.7. Consumo de energía de las plataformas Cubieboard 3, Intel Galileo 2Gen, Arduino Mega y Raspberry Pi 3 Modelo B [41] [43] [46] [50].

	Cubieboard 3	Intel Galileo 2Gen	Arduino Mega	Raspberry Pi 3 Modelo B
Consumo de energía	12.5 W	3 W	1 W	2.5 W

Analizando los datos de consumo se observa que la Cubieboard 3 es la de mayor consumo, aunque comparado con un ordenador es pequeño, pero comparado con las otras placas es bastante la diferencia. Las placas Intel Galileo 2Gen y Raspberry Pi 3 modelo B tienen un consumo similar. La placa de menor consumo en términos de potencia es Arduino Mega.

Otro aspecto a considerar es la memoria RAM (*Random Access Memory*, Memoria de Acceso Aleatorio) y el almacenamiento que tiene cada placa. La memoria RAM es uno de los factores que aumentara el rendimiento, haciendo que la placa sea más rápida en el procesamiento de actividades.

Todas las placas permiten almacenar datos a través de una memoria microSD, la capacidad dependerá de las necesidades de cada proyecto. Una ventaja de la Cubieboard es que tiene para conectar un disco duro SATA de hasta 2TB; pero ya que el prototipo almacenará los datos de temperatura en una base de datos de un servidor, no será necesario tener gran capacidad de almacenamiento. La memoria RAM con la que cuenta cada placa se muestra en la Tabla 2.8.

Tabla 2.8. Memoria RAM de las placas Cubieboard 3, Intel Galileo 2Gen, Arduino Mega y Raspberry Pi 3 Modelo B [41] [43] [46] [50].

	Cubieboard 3	Intel Galileo 2Gen	Arduino Mega	Raspberry Pi 3 Modelo B
Memoria RAM	1 GB	256 MB	8 KB	1 GB

Las tareas a ejecutar en la placa no demandarán gran cantidad de memoria RAM, 512 MB sería suficiente, sin embargo se recomienda por lo menos 1 GB de RAM para futuras aplicaciones del prototipo. Las placas que cumplen este requerimiento son la Raspberry Pi y la Cubieboard.

Por último se analizará los precios de las placas. Para tener una mejor referencia se tomaran los valores directamente del fabricante o su principal distribuidor. No se incluirá impuestos ni costos por envío. En la Tabla 2.9 se muestran los costos de cada placa.

Tabla 2.9. Precio de las plataformas de hardware libre [41] [43] [46] [50].

	Cubieboard 3	Intel Galileo 2GEn	Arduino Mega	Raspberry Pi 3 Modelo B
Precio (USD)	120,00	60,00	41.95	35,00

La placa Raspberry Pi 3 modelo B al ser la más popular al momento existe un mayor número de usuarios, por esto su precio es menor al de placas similares. Además al haber más usuarios se crea una comunidad con ideas y soluciones a varios problemas.

Al analizar los criterios de consumo de energía, memoria RAM y precios, se concluye que la plataforma de hardware libre que más se adapta a los requerimientos del proyecto es la **Raspberry Pi 3 Modelo B**. Se escoge esta placa por sus características (procesador de 1,2 GHz de cuatro núcleos, 1 GB de memoria RAM, etc.) y puertos con los que cuenta (Ethernet, GPIO, USB, etc.) que son de mucha utilidad para el proyecto. Además de tener un consumo bajo de potencia comparado con las otras placas, suficiente memoria RAM para las aplicaciones deseadas y un precio bajo comparado a las características brindadas por la placa.

2.2 Sensores de Temperatura

Los sensores de temperatura son dispositivos que transforman las variaciones de temperatura en señales eléctricas. La clasificación de los sensores se muestra en la Figura 2.4 [51].

A continuación se hablará brevemente de los cuatro tipos de sensores de temperatura, los termopares, los termistores, detectores de temperatura de resistencia y sensores infrarrojos.

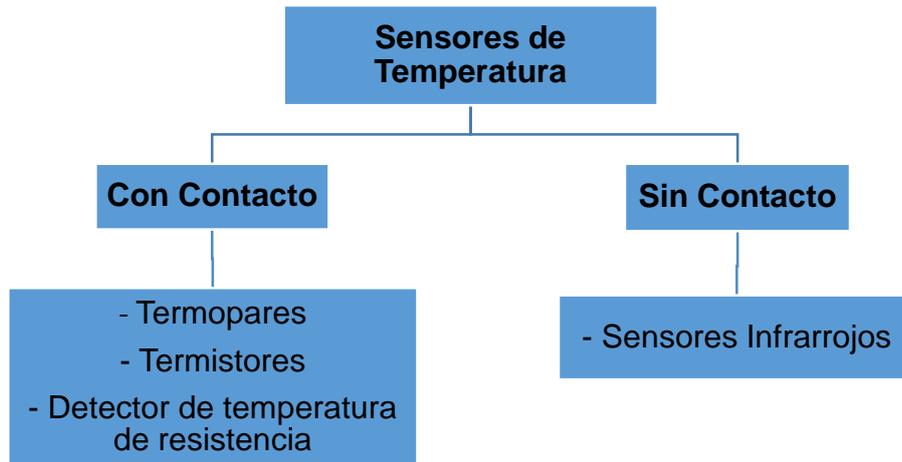


Figura 2.4. Tipos de Sensores de Temperatura.

2.2.1 Termopares

Se usa un par de empalmes o uniones que son de materiales distintos, así un empalme tiene una temperatura de referencia y el otro representa la temperatura a medir. Esto funciona cuando una diferencia de temperatura crea un voltaje (efecto Seebeck), a su vez este voltaje es convertido en una lectura de temperatura.

Estos sensores son usados porque son económicos, resistentes y confiables, no requieren batería y pueden ser usados en un gran rango de temperaturas [51] [52].

2.2.2 Termistores

Son sensores de temperatura de bajo costo y de fácil adquisición, asequibles y adaptables. Se utilizan para realizar fáciles mediciones de temperatura en lugares de alta temperatura.

Están contruidos de material semiconductor con una resistencia muy sensible a la temperatura. La resistencia de un termistor se reduce al ascender la temperatura, la variación de la resistencia crea una diferencia de voltaje que se transforma en temperatura. Los termistores son utilizados generalmente como limitadores de corriente de irrupción, sensores de temperatura, protectores contra sobrecargas de reinicio automático, y elementos de calentamiento autorregulados [52].

2.2.3 Detectores de temperatura de resistencia

Un RTD (*Resistance Temperature Detector*, Detector de Temperatura de Resistencia) es un sensor de temperatura fundamentado en las variaciones de resistencia de un conductor a medida de que asciende la temperatura. Los RTD tienen mejor precisión que los termopares y tienen estabilidad a largo plazo.

Los materiales empleados para la fabricación del RTD son platino, cobre, níquel y molibdeno. Generalmente se utilizan los sensores de platino ya que sus propiedades los hacen especiales para el trabajo [51].

2.2.4 Sensores Infrarrojos

Son utilizados para medir la temperatura en superficies que van desde -70 a 1000° C. Transforman la energía térmica lanzada desde un objeto en un rango de longitud de onda de 0.7 a 20 μm dependiendo el modelo, en una señal eléctrica para transformarla y expresarla en unidades de temperatura tras compensar cualquier temperatura del ambiente [52].

2.2.5 Sensor de Temperatura DS18B20

El sensor DS18B20 es un termómetro digital de gran precisión en su medición, su temperatura operativa se encuentra entre -50 y 125 grados Celsius. Su precisión, en un rango entre -10 y 85 grados, es de ± 0.5 grados, es accesible por su bajo costo, Su interfaz es de fácil funcionamiento, además que es práctico para mediciones de temperatura que requieran precisión y confiabilidad. Existen tres modelos disponibles; como transistor suelto, como módulo (con la resistencia incorporada, para ser usado) o a su vez en el modelo sumergible, como se muestra en la Figura 2.5 [53].

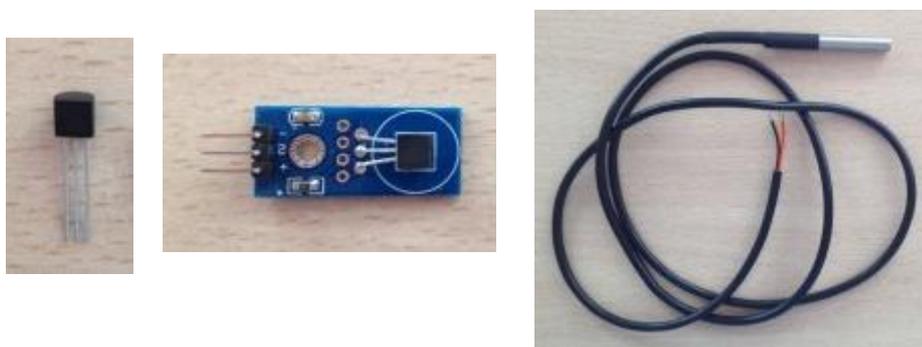


Figura 2.5. Encapsulados DS18B20 (Transistor, con modulo y sumergible).

Este dispositivo que se comunica de forma digital, cuenta con tres terminales, los dos de alimentación y el pin *data*. Utiliza la comunicación 1-Wire, la cual es un poco complicada para los que no tienen mucha experiencia en el área de electrónica digital. Básicamente se trata de un protocolo especial que permite enviar y recibir datos utilizando un solo cable, a diferencia de la mayoría de los protocolos que requiere dos vías [53].

2.2.5.1 Características

Es importante saber qué rango de temperaturas es capaz de medir un sensor, también conocer el error que puede llegar a tener en las mediciones y la resolución del sensor de temperatura. El rango de temperaturas que el sensor DS18B20 puede medir varía entre -55°C y 125°C [53].

Una de las características importantes de este sensor es que se puede trabajar con diferentes resoluciones, refiriéndose a la variación mínima que se puede medir entre dos temperaturas. El DS18B20 acepta resoluciones de 9-bit, 10-bit, 11-bit y 12-bit. Y por defecto utiliza la de 12-bit [53]. Las variaciones para cada resolución se la puede observar en la Tabla 2.10.

Tabla 2.10. Variaciones para cada resolución.

RESOLUCIÓN	TEMPERATURA
9-bit	0,5°C
10-bit	0,25°C
11-bit	0,125°C
12-bit	0,0625°C

El DS18B20 incorpora una memoria de 64-bit (equivalente a 8 bytes) con el objetivo de almacenar el identificador o ubicación única de cada sensor. El primer byte reconoce la tipología del componente, por ejemplo para los DS18B20 es el número 28 en hexadecimal.

Esta dirección única es indispensable dentro del bus 1-Wire para reconocer cada uno de los sensores de temperatura DS18B20 conectados al bus de comunicación. Debido a esta comunicación, se consiguen dos cosas; por un lado robustez en la transmisión de los datos ya que trabaja con datos digitales, y por otro lado permite conectar varios sensores a un pin digital [54].

Utiliza el sistema de verificación de redundancia cíclica CRC (*Cyclic Redundancy Code*, Código de Redundancia Cíclica) para la verificación de errores en los datos, el código CRC se almacena en la memoria [53]. También guarda la temperatura conseguida y tiene dos alarmas que se disparan si la temperatura es mayor o menor que un umbral de temperatura máxima o mínima. El DS18B20 se convierte en un sensor bastante potente con gran capacidad, superior a otros sensores de temperatura de la misma clase. Un resumen de las características mencionadas se muestra en la Tabla 2.11.

Tabla 2.11. Características DS18B20 [55].

Rango de temperatura	-55 a 125°C
Resolución	9 a 12 bits
Precisión	±0.5°C (de -10°C a +85°C) ±2°C (de -55°C a +125°C)
Alimentación	3v a 5.5v

2.2.5.2 Error en la medición

El sensor de temperatura DS18B20 tiene errores debido al ruido inherente en los circuitos eléctricos y a variaciones con respecto al entorno físico. Para temperaturas entre -10°C y 85°C se puede tener un margen de error de ±0,5°C. Para el resto de temperaturas oscila entre -55°C y 125°C el error es de ±2°C (Figura 2.6.). Esto equivale a decir que si el sensor DS18B20 suministra una temperatura de 23°C el valor real estará entre 22,5°C y 23,5°C. Si por el contrario suministra un valor de 90°C el valor real estará entre 88°C y 92°C [53].



Figura 2.6. Error de medición en el sensor DS18B20.

2.2.5.3 Métodos de alimentación

Se pueden usar dos métodos para realizar la alimentación al sensor DS18B20.

El primer método es empleando el pin de datos. Así el sensor obtiene energía del pin de datos cuando este se encuentre en un nivel alto y almacena carga en un condensador para cuando la línea de datos este en un nivel bajo. Se usa esta configuración cuando el sensor este a grandes distancias o el espacio sea limitado. En la Figura 2.7 se muestra el sensor conectado a un microprocesador con alimentación empleando pin de datos [55].

El otro método es alimentación mediante fuente externa, así el sensor se alimenta a través del pin de voltaje, de esta forma el voltaje será independiente y estable. Esta configuración

es la más recomendada y usada. En la Figura 2.8 se muestra la conexión del sensor a un microprocesador con alimentación de fuente externa [55].

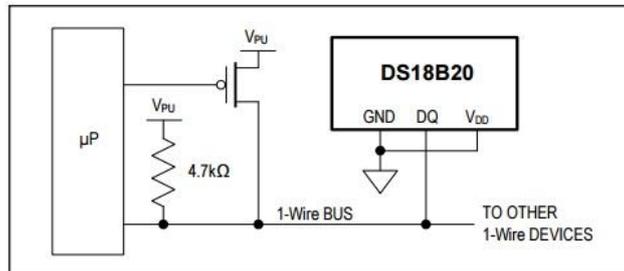


Figura 2.7. Alimentación empleando el pin de datos.

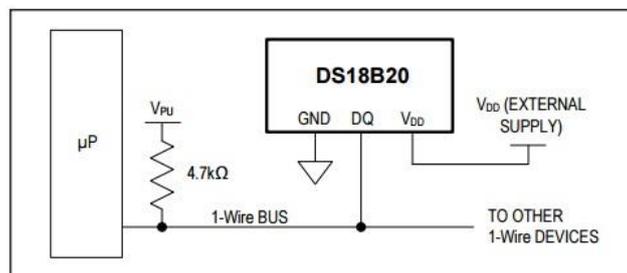


Figura 2.8. Alimentación empleando fuente externa.

Por todas las características ofrecidas de rango de temperatura, resolución, precisión y precio se escoge el **sensor de temperatura DS18B20** en su encapsulado resisten al agua, encapsulado que sirve mucho para este proyecto y el ambiente en el que se lo va a usar.

2.3 Modem 4G

Existen módulos 4G que se pueden conectar directamente a los pines GPIO de la Raspberry Pi, estos son conocidos como *Pi HATs* o *Shields*, y se los configura mediante comandos AT. Hasta el momento no se han popularizado este tipo de módulos 4G, esto se puede dar porque su precio es muy elevado como se ve en la Figura 2.9 donde venden el módulo 4G para Raspberry en Reino Unido a 446,29 USD (372 EUR); es un precio muy elevado ya que para este proyecto se busca que el prototipo sea de bajo costo [56].



Figura 2.9. Shield 4G para Raspberry Pi.

Se puede conseguir módulos 3G o 2G con tecnologías WCDMA (UMTS) o GSM/GPRS respectivamente, pero para este proyecto se propuso usar tecnología 4G LTE, en los lugares donde haya disponibilidad.

Así que una solución es usar un modem comercial para conectarlo a la placa mediante puerto Ethernet o con interfaz Wi-Fi, aprovechando que la placa viene con esta interfaz de fábrica. También algunos módems comerciales se pueden conectar a la placa mediante puerto USB.

2.3.1 Modem Mobile WiFi

El Huawei Mobile WiFi E5573 se conecta a una computadora o dispositivo mediante puerto USB o Wi-Fi, permite acceder a internet mediante tecnología 3G o 4G dependiendo de la disponibilidad de la red celular y de la operadora de telefonía móvil. El Mobile WiFi es compatible con cualquier tipo de sistema operativo.

Con este dispositivo se puede navegar desde cualquier lugar dependiendo de la cobertura que se tenga en el sitio. Si bien Ecuador tiene cobertura 4G bastante amplia, la realidad dependerá del operador del servicio celular usado [57].

Entre sus ventajas principales se puede encontrar:

- La movilidad: es lo que diferencia de cualquier otro mecanismo de conexión a la red. Y si bien, en cobertura y velocidad no es la mejor opción, la movilidad es de mucha utilidad.
- Su instalación es rápida y sencilla: Puede saltar de una red a otra para ofrecer mejor cobertura y velocidad.
- Puede trabajar con cualquier ordenador o dispositivo, y con la mayoría de sistemas operativos.

Para este proyecto se adquirió el modem LTE Huawei Mobile WiFi E5573 provisto por la empresa Claro, como se muestra en la Figura 2.10.



Figura 2.10. Modem Huawei WiFi E5573.

Las características más importantes del modem se muestran en la Tabla 2.12.

Tabla 2.12. Características modem Huawei WiFi E5573 [57].

Sistema de Comunicación	LTE: FDD 3G: WCDMA 2G: EDGE/GPRS/GSM
Velocidad	LTE FDD: DL 150 Mbps / UL 50 Mbps DC-HSPA+: DL 43,2 Mbps HSPA + hasta 21,6 Mbps HSPA: DL 14,4 Mbps, UL 5,76 Mbps UMTS: 384 Kbps EDGE: 236,8 Kbps GPRS: 85,6 Kbps
Pantalla	1.45" TFT LCD
Batería	1500 mAh
Tiempo máximo de trabajo	6 horas
Protocolo de WI-FI	IEEE 802.11a/b/g/n
Tiempo de espera máximo	300 horas mientras Wi-Fi está apagado

2.4 Sistema Fotovoltaico

El sistema fotovoltaico permitirá a partir de la insolación, producir energía eléctrica con el fin de ser aprovechada por los equipos que se usarán en el proyecto. El sistema consta de los siguientes elementos:

- Un generador solar, dispuesto por un conjunto de paneles fotovoltaicos, que absorben la insolación luminosa procedente del sol y la convierten en corriente continua a baja tensión.
- Un acumulador, que guarda la energía producida por el generador y permite tener corriente eléctrica fuera de las horas de luz o en días nublados [58].

2.4.1 Dimensionamiento del sistema fotovoltaico

La carga total de potencia diaria del prototipo se calcula a partir del uso de horas de funcionamiento en el día, como se muestra en la Tabla 2.13.

Tabla 2.13. Energía consumida por las cargas

Equipo	Potencia [W]	Uso (horas/día)	Energía (Wh/día)
Raspberry Pi	2.5	24	60
Modem	1.5	24	36
Sensores	0.1	24	2.4
Otros	0.6	24	14.4
Carga Total Diaria (Wh/día):			112.8

Determinada ya la potencia diaria de consumo se puede calcular la intensidad de corriente total que necesitará el prototipo, como se ilustra en la Ecuación 2.1.

$$I_T = \frac{\text{Carga Total Diaria}}{\text{Voltaje}} = \frac{112.8 \text{ Wh/día}}{5 \text{ V}} = 22.56 \frac{\text{Ah}}{\text{Día}}$$

Ecuación 2.1. Intensidad de corriente total.

Como en todo diseño se emplea un factor de seguridad, como se muestra en la Ecuación 2.2.

$$I_T = 22.56 \frac{\text{Ah}}{\text{Día}} * 1.2 = 27.072 \frac{\text{Ah}}{\text{Día}}$$

Ecuación 2.2. Intensidad con factor de seguridad.

Para obtener el valor de capacidad de la batería se considera una autonomía de 1 día, como se ilustra en la Ecuación 2.3.

$$\text{Capacidad de batería} = 27.072 \frac{\text{Ah}}{\text{Día}} * 1 \text{ día} = 27.072 \text{ Ah}$$

Ecuación 2.3. Capacidad de la batería.

Se necesitará una batería de $\approx 30 \text{ Ah} = 30000 \text{ mAh}$

Para dimensionar los paneles solares se necesita el índice de radiación, como es un diseño se considera el peor caso ($4.14 \frac{\text{kWh}}{\text{m}^2 \text{ día}}$), que sería el menor valor de la Tabla 2.14.

Tabla 2.14. Índice de Radiación [58].

Mes	Ene	Feb	Mar	Abr	May	Jun	Jul	Ago	Sep	Oct	Nov	Dic
Insolación global [kWh/m ² /día]	4,41	4,48	4,65	4,36	4,27	4,14	4,30	4,62	4,97	4,88	4,94	4,83

Conocido el índice de radiación a usar, se usa la fórmula de carga total diaria para encontrar la potencia del panel, como se muestra en la Ecuación 2.4.

$$Carga\ total\ diaria = P_{panel} * 4,14 = 112.8\ Wh/dia$$

Ecuación 2.4. Fórmula para sacar la potencia necesaria del panel solar.

Despejando en la Ecuación 2.4 el valor de potencia del panel se obtiene el valor de potencia necesario para los paneles solares.

$$P_{panel} = 28.2\ W$$

2.4.2 Batería

La estación de energía Aomaso de 42000mAh cumple con los requerimientos de capacidad calculados en la sección anterior, es más, tiene mayor capacidad a la calculada (30000 mAh) dando un factor de seguridad.

La batería pesa 800g y contiene 3 salidas USB y 1 salida de encendedor de cigarrillos, de los tres puertos USB dos son a 5V y 2.1A, y el otro puerto USB es a 5V y 1A. La salida de cigarrillo es DC 12V/15A para cargadores de automóvil.

La cubierta de silicona de larga duración bloquea el agua y el polvo, haciendo a esta batería resistente al agua. Los dispositivos simplemente se conectan a la salida USB del dispositivo, pudiendo usarse los cuatro puertos al mismo tiempo. No es necesario apagar el dispositivo cuando no se esté usando, ya que se apaga automáticamente después de segundos inactivos. El LED brillante significa que ha equipado una lámpara confiable y una alerta de emergencia por la noche y al aire libre. En la Figura 2.11 se muestra la batería AOMASO de 42000 mAh [59].



Figura 2.11. Batería AOMASO de 42000 mAh.

2.4.3 Paneles Solares

Los paneles solares Giaride de 21 W cumplen con la resistencia al agua necesitada para el proyecto, además para cumplir con los requerimientos de capacidad, se usará dos

paneles solares Giaride de 21W para cumplir con los 28.2 W calculados, así los dos paneles cumplen de sobremanera la potencia deseada teniendo un factor de seguridad.

Los paneles solares pueden reconocer diferentes dispositivos y ajustar la corriente y voltaje para ahorrar el tiempo de carga, además cuenta con carga rápida para los dispositivos. Cuenta con alta eficiencia de conversión 22%-25% que proporciona más energía a los dispositivos.

Cuenta con puerto USB a 5V/2.5A y puerto de carga de 18V/1A. El puerto de 18V está pensado para cargas de ordenadores portátiles, baterías de 12V y otros dispositivos que funcionen a 18V. El puerto USB está pensado para apoyo de baterías externas, tablets, teléfonos móviles y otros dispositivos de 5V USB.

Su diseño es compacto y portátil con tecnología *eyeholes* de acero inoxidable, pet laminado de alta temperatura; resistente al agua y a golpes, diseñado para exteriores.

Además estos paneles mantienen a los dispositivos seguros contra la protección de cortocircuitos, protección de sobrecarga y protección contra el sobrecalentamiento. En la Figura 2.12 se muestra como son los paneles solares Giaride de 21 W [60].



Figura 2.12. Panel Solar Giaride de 21 W.

En la Tabla 2.15 se muestra las especificaciones de los paneles solares Giaride de 21W.

Tabla 2.15. Especificaciones paneles solares Giaride 21W [60].

Panel Solar	21W 18V
Eficiencia de transformación	22% a 25%
Salida DC	18V a 1A USB: 5V a 2.5A
Dimensiones	Plegable: 6.2 x 11.8 x 0.7 pulgadas Abierto: 25 x 11.8 x 0.1 pulgadas
Peso	1.9 onzas

2.5. Configuración de los equipos

Una vez seleccionados los equipos en base a los requerimientos y características del proyecto, ahora se implementará el prototipo para tomar datos de temperatura de la superficie del agua y enviarlo a través de la red celular para poder ser observado por los usuarios en una página web.

La Raspberry Pi tomará datos de temperatura de los sensores ubicados a un metro, tres metros y cinco metros bajo el nivel del agua; una vez obtenido los datos de temperatura se enviará a través del modem celular a la base de datos que se encontrará en un servidor web. En el servidor también se almacenará el sitio web donde los usuarios podrán ingresar para observar los datos de temperatura con su respectiva estampa de tiempo; además podrán observar gráficas de la tendencia de la temperatura a las tres diferentes profundidades. En la Figura 2.13 se muestra un esquema de los elementos más importantes que se usarán en este proyecto.

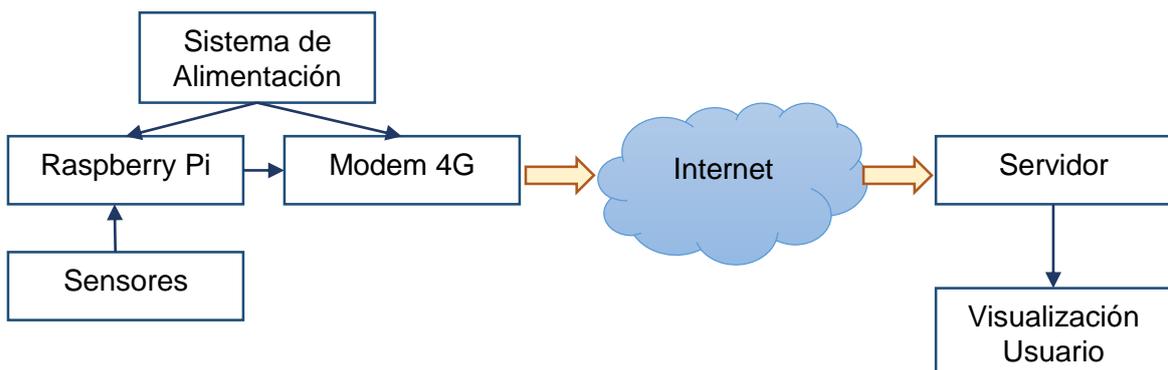


Figura 2.13. Esquema de elementos usados en el proyecto.

2.5.1 Configuración de la Raspberry Pi

La plataforma de hardware libre seleccionada fue la Raspberry Pi 3 Modelo B, esta placa es uno de los principales elementos para el procesamiento de la información obtenida de los sensores.

2.5.1.1 Instalación del sistema operativo

Para usar la placa primero se debe instalar un sistema operativo, ya que la placa de fábrica viene sin ningún sistema. Para instalar el sistema operativo se adquiere una memoria microSD de 16 GB que además almacenará las aplicaciones y programas para el funcionamiento del prototipo.

El sistema operativo seleccionado es el recomendado para la Raspberry Pi, Raspbian, una distribución del sistema operativo GNU/Linux basado en Debian.

Para instalar Raspbian, primero se descarga NOOBS (*New Out of Box Software*) de la página oficial de Raspberry (<https://www.raspberrypi.org/downloads/>), que permite instalar cualquier distribución que el usuario desee de manera fácil. En la página oficial se encuentran dos versiones de NOOBS para descargar, la normal y la *LITE* como se observa en la Figura 2.14, la versión normal no necesita de internet para ser instalada a diferencia de la versión *LITE* que necesita de conexión a internet obligatoriamente.



Figura 2.14. Versiones para descargar de NOOBS en la página oficial de Raspberry Pi.

Se selecciona la versión normal para descargar, y se descomprime el archivo en la tarjeta microSD después de formatearla; echo esto, la tarjeta estará lista para ser ingresada en la Raspberry Pi.

Después, se conecta un monitor, teclado, ratón y la tarjeta microSD; una vez hecho esto, se alimenta a la Raspberry Pi para que arranque; apenas inicie la placa la herramienta NOOBS permitirá elegir qué sistema operativo se quiere instalar (Figura 2.15); además, se encargará de todas las tareas adicionales como el formateo y la preparación del sistema.

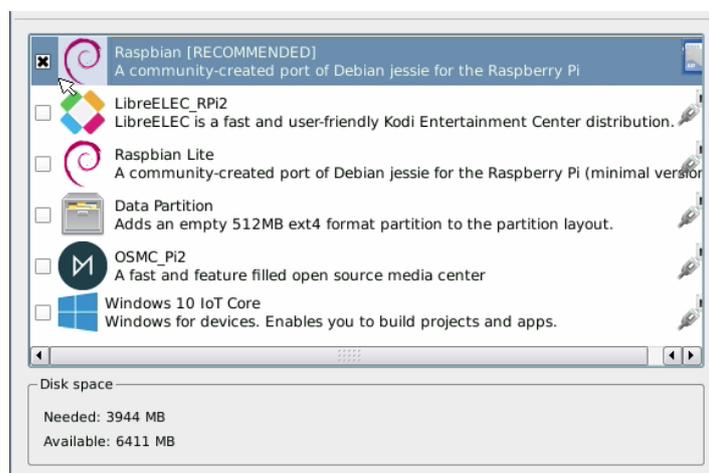


Figura 2.15. Selección del sistema operativo a instalar.

Una vez seleccionada la versión recomendada, se tiene que esperar un tiempo a que se instale el sistema operativo, como se muestra en la Figura 2.16.



Figura 2.16. Instalación del sistema operativo Raspbian.

Finalmente la Raspberry Pi estará lista para ser usada y configurada para el proyecto.

2.5.1.2 Configuraciones iniciales

La Raspberry Pi necesita de configuraciones iniciales como autenticación, configuración de zona horaria, configuración de interfaz de red y Wi-Fi, etc. Si bien la mayoría de configuraciones se realizaron a través de NOOBS al iniciar la Raspberry Pi, es necesario hacer otras configuraciones adicionales y eliminar ciertos programas.

Raspbian es un sistema operativo con seguridad fuerte, por esto hay usuarios con derechos restringidos. El usuario por defecto es pi, para el proyecto se deja con esta configuración por defecto ya que no habrá personas externas manipulando la Raspberry Pi.

Por motivos de seguridad, ciertos comandos necesitan derechos más altos de los que vienen por defecto, que solo lo puede hacer el usuario *root*; para cuando se use estos comandos que necesitan un nivel de acceso superior, no es necesario cambiar de usuario, simplemente se llama a los derechos de *root* mediante el comando *sudo* (*SUPERuser DO*) precediendo al comando que se desea usar.

Para configurar e ingresar comandos a la Raspberry se lo hace a través de la línea de comandos, abriendo el terminal (Figura 2.17.).

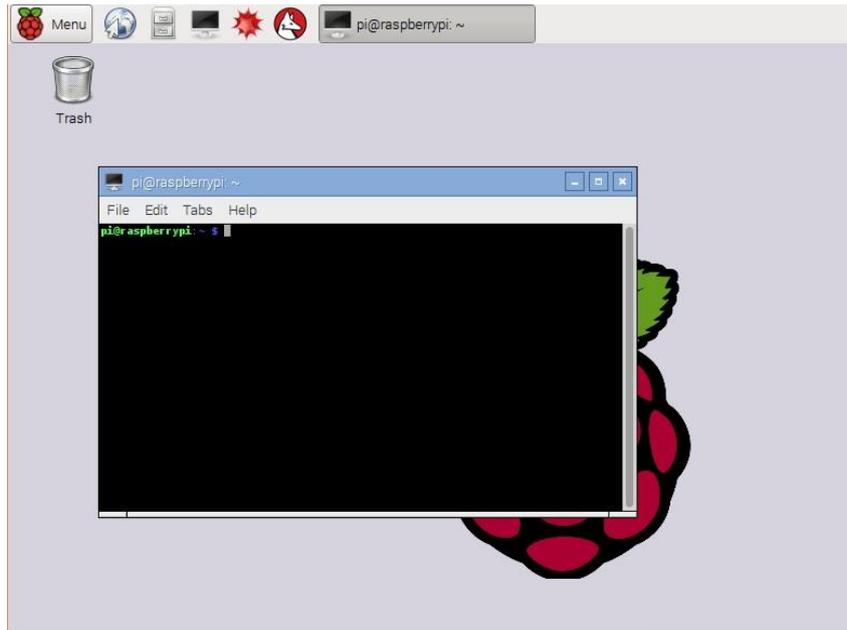


Figura 2.17. Línea de comandos a través del terminal.

Con el terminal abierto, se ingresan los comandos para eliminar software innecesario que no se usa en el proyecto. Se ingresa el siguiente comando:

```
sudo apt-get purge claws-mail gpicview xpdf raspberrypi-artwork bluej greenfoot  
idle* pypy pypy-lib debian-reference-en debian-reference-common dillo  
epiphany-b rowser gvfs* libservlet2.5-java netsurf-* zenity* libreoffice*  
minecraft-pi nodered nuscratch penguinspuzzle python-minecraftpi python-pygame  
python3-pgzero python3-piface* python3-pygame python3-uno scratch sense-hat  
smartsim sonic-pi timidity wolfram-engine
```

Después, para eliminar los paquetes que fueron instalados como dependencias y que no son necesarios, se usa el siguiente comando:

```
sudo apt-get autoremove --purge
```

Ahora se actualiza a la última versión todos los programas y aplicaciones restantes con los siguientes comandos:

```
sudo apt-get update  
sudo apt-get dist-upgrade  
sudo apt-get clean
```

Finalmente se configura la zona horaria a través de una interfaz en donde se selecciona la región y ciudad en donde se encuentra, la interfaz mencionada se abre con el comando:

```
sudo dpkg-reconfigure tzdata
```

2.5.2 Configuración de los sensores

Para la adquisición de datos de temperatura superficial del agua se usa sensores de temperatura tipo sonda resistente al agua DS18B20.

2.5.2.1 Tiempo de muestreo de temperatura

Para la adquisición de los datos de temperatura del agua, se debe establecer un tiempo de muestreo para adquirir y transmitir estos valores, es decir, cada que tiempo se debe tomar datos de temperatura en los tres sensores.

Este tiempo debe ser adecuado, no tan grande para perder datos importantes, y no tan pequeño para obtener datos redundantes que no aporten al estudio de la temperatura superficial del agua.

Además el tiempo de adquisición y transmisión dependerá del lugar en donde se realizarán las pruebas, ya que la temperatura superficial del agua varía dependiendo de la temperatura del lugar en donde se encuentre instalado el prototipo.

Como las pruebas se las realizará en la Laguna de Colta, se debe observar como varía la temperatura ambiental del lugar en relación al tiempo. Se observó que no hay cambios significativos en cortos periodos de tiempo; así se selecciona el tiempo de adquisición y transmisión de temperatura cada **10 minutos**; este tiempo es idóneo ya que no se obtendrá abundante información y no se perderá datos de temperatura importantes. Además este parámetro de temperatura superficial del agua no es crítico para actualizarlo a un menor tiempo del seleccionado.

2.5.2.2 Modo de conexión

Aprovechando que los sensores DS18B20 cuentan con el protocolo 1-Wire, se tomará valores de temperatura de los sensores a través de un único hilo, conectado a un pin de la placa (pin número 7, GPIO 4). La alimentación de los sensores se hará a través del pin de 3,3 V (pin número 1 de la placa) y al pin de tierra (pin número 6), como se muestra en la Figura 2.18.

Además, entre el hilo positivo y el de datos de los sensores se coloca una resistencia de *Pull-Up*, cuyo valor dependerá de la distancia del sensor a la placa. En la Tabla 2.16 se muestra valores de resistencia de Pull-Up, dependiendo de la distancia del cable.

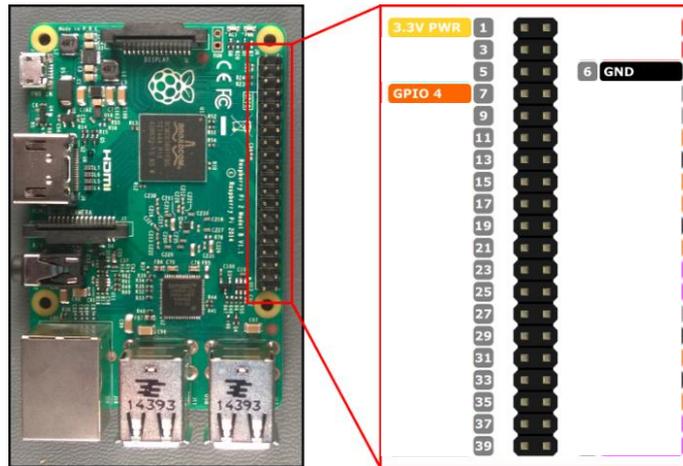


Figura 2.18. Pines usado de la Raspberry Pi para conectar los sensores de temperatura.

Tabla 2.16. Resistencia de Pull-Up dependiendo de la distancia del sensor.

Resistencia de Pull-Up	Distancia del Cable
4,7 K Ω	De 0m a 5m
3,3 K Ω	De 5m a 10m
2,2 K Ω	De 10m a 20m
1,2 K Ω	De 20m a 50m

Como la distancia máxima del sensor de temperatura a la Raspberry Pi es de cinco metros, se usa una resistencia de Pull-up de 4,7K Ω , que sirve para los tres sensores, como se muestra en la Figura 2.19.

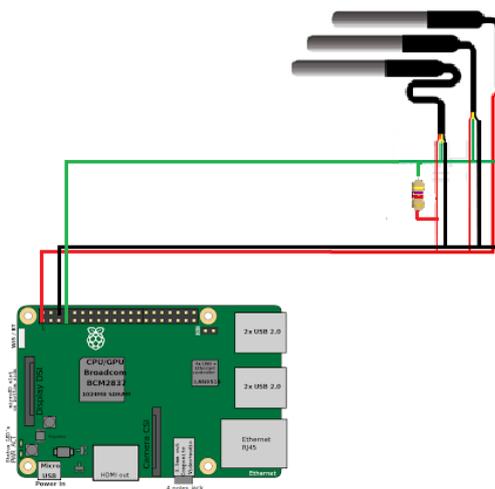


Figura 2.19. Conexión de los sensores a la placa con resistencia de Pull-Up.

2.5.2.3 Programa para tomar datos de los sensores

Una vez conectados los sensores a la placa, de la manera que se especificó en la sección anterior, se configura la placa y se crea un script en Python para obtener los datos de temperatura de cada sensor y almacenarlos en la base de datos.

Primero se configura la placa para trabajar con el protocolo 1-Wire y se especifica por qué pines están conectados los sensores. Para esto se edita el fichero *config.txt* que se encuentra en */boot* con el siguiente comando:

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

En el archivo se añade una línea que permite cargar módulos en el Kernel del sistema para manejar el bus 1-Wire del conector. Además se declara que se usará el GPIO 4 (pin número siete, ver Figura 2.18.). En la Figura 2.20 se muestra el archivo “config.txt” editado.

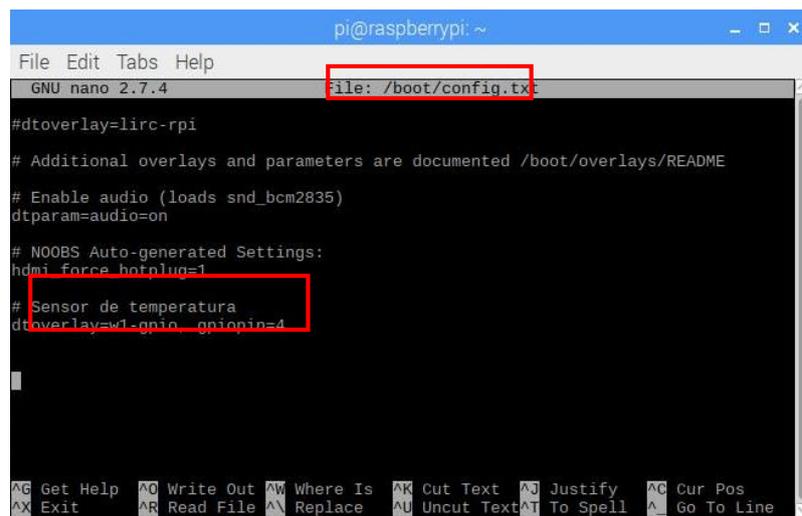


Figura 2.20. Edición del archivo config.txt.

Después, se guarda el archivo que se acabó de editar y se reinicia la placa. Ahora, los sensores ya podrán trabajar con la placa Raspberry Pi sin ningún problema.

La forma en que trabajan los sensores con la Raspberry Pi es creando un fichero, donde escriben los datos de temperatura. Para observar que la placa lee todos los datos de los sensores, hay que mirar las carpetas que inician “28” dentro del directorio que maneja las comunicaciones del bus 1-Wire que es el siguiente.

```
ls /sys/bus/w1/devices/
```

Cada una de las carpetas que aparecen en el directorio anterior y que empiezan con “28”, corresponde al número de dirección de cada sensor de temperatura como se ilustra en la

Figura 2.21. Si falta alguna carpeta, se debe comprobar la conexión entre el sensor y la placa.

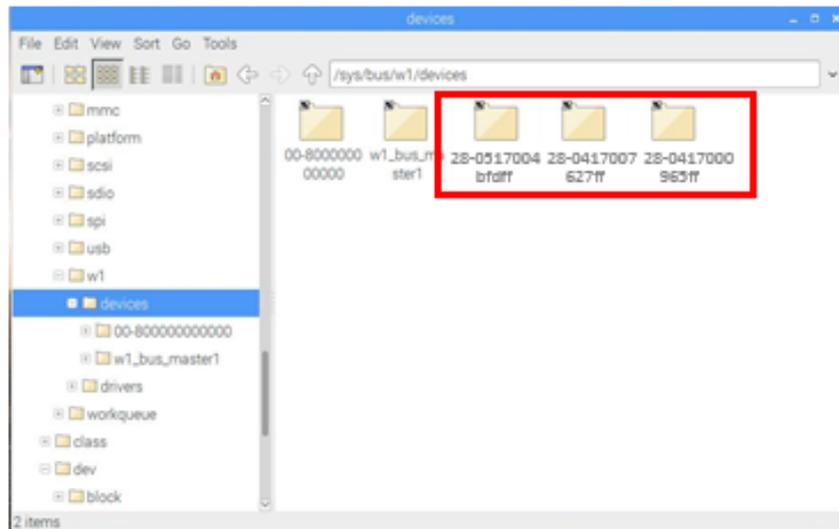


Figura 2.21. Carpetas donde se almacenan los datos de temperatura de cada sensor.

En la Figura 2.21 se observa las carpetas de los tres sensores de temperatura, cada carpeta empieza con “28” y después contiene la dirección de cada sensor:

- Sensor a un metro: 28-0517004bfdf
- Sensor a tres metros: 28-0417007627f
- Sensor a cinco metros: 28-0417000965f

Para leer los datos de cada sensor se tendrá que acceder a cada una de las carpetas de los sensores, y leer el fichero “w1-slave”. En este fichero se tendrá el valor de temperatura obtenida por el sensor * 1000, es decir, para obtener el valor real de temperatura en grados centígrados se dividirá el valor del parámetro t para 1000.

Para tomar los datos de temperatura de los tres ficheros cada 10 minutos y almacenar estos valores en la base de datos, se usará un script desarrollado en Python llamado “temperatura_agua.py”.

Una vez creado el script con el nombre “temperatura_agua”, se empieza a escribir líneas de código para importar librerías. En el Segmento de código 2.1 se añade librerías para el manejo de ficheros, directorios del sistema, tiempo y base de datos.

```

6 #Importamos librerias necesarias
7
8 import sys
9 import os
10 import curses
11 import time
12 import MySQLdb
13
14 from datetime import datetime
15

```

Segmento de código 2.1. Importación de librerías necesarias para el programa “temperatura_agua”.

Después, se carga los módulos del kernel del sistema para manejar el bus 1-Wire del conector GPIO de la Raspberry Pi, como se ilustra en el Segmento de código 2.2.

```

16
17 # Carga de modulos para sensor DS18B20 OneWire
18 os.system("sudo modprobe wire")
19 os.system("sudo modprobe w1-gpio")
20 os.system("sudo modprobe w1-therm")
21

```

Segmento de código 2.2. Módulos para manejar el bus 1-Wire.

A continuación, se crea una función por cada sensor para tomar datos de temperatura del fichero donde se almacena el valor de temperatura. Así se podrá llamar a la función cada vez que se quiera obtener el dato de temperatura de cada sensor, en el Segmento de código 2.3 se muestra la función para el sensor de un metro.

```

23 # Funcion para la lectura de temperatura del sensor a 1 metro
24
25 def dametempl(): # Nombre de la funcion dametempl.
26     tfile = open("/sys/bus/w1/devices/28-0517004bfdff/w1_slave") # Se abre el fichero donde se almacena la temperatura del sensor a 1m.
27     lectura = tfile.read() # Se almacena la informacion del fichero en la variable lectura.
28     tfile.close() # Se cierra el fichero.
29     lineal = lectura.split("\n")[1] # Se busca dentro del fichero el valor de t.
30     temperaturaleida = lineal.split(" ")[9]
31     temperatura = float(temperaturaleida[2:])
32     temperatura = temperatura / 1000 # Se divide la temperatura para 1000 para obtener el valor real
33     rtemp = round (temperatura,3) # Se redondea el valor de temperatura a tres decimales
34     return rtemp # La funcion retorna la variable rtemp con el valor de temperaturtra
35

```

Segmento de código 2.3. Función que devuelve el valor de temperatura del sensor a un metro.

Para el proyecto se cree conveniente que los valores de temperatura tomados, además de almacenarlos en la base de datos, se muestren en la pantalla de la Raspberry Pi con la respectiva estampa de fecha y hora. Así se podrá observar en pantalla los valores que se van tomando. Para mostrar estos datos en pantalla se lo hace a través de cursores, por tanto hay que declararlos y definir colores y bordes, como se muestra en el Segmento de código 2.4.

```

61
62 # Utiliza curses para visualizar salida a pantalla
63 stdscr = curses.initscr()
64 curses.cbreak()
65 curses.start_color()
66
67 # Definicion de paletas de colores, borde
68 curses.init_pair(1, curses.COLOR_RED, curses.COLOR_BLACK)
69 curses.init_pair(2, curses.COLOR_YELLOW, curses.COLOR_BLACK)
70 stdscr.border(0)
71 hsize = curses.COLS
72 vsize = curses.LINES
73 stdscr.keypad(1)
74 stdscr.nodelay(1)
75
76 try:
77     curses.curs_set(0)

```

Segmento de código 2.4. Definición de cursores para mostrar la temperatura en pantalla.

Así, para tomar datos cada diez minutos, almacenarlos en la base de datos y a su vez mostrarlos en pantalla, se crea un lazo para que vaya realizado todo el proceso periódicamente; y únicamente saldrá de este lazo cuando el usuario quiera y presione la tecla Q, como se muestra en el Segmento de código 2.5.

```

79 # Rutina principal del programa
80 while True:
81     char = stdscr.getch()
82     if (char == 113 or char == 81):
83         break # q
84     else:
85         conexion = MySQLdb.connect('localhost','sensores','c2875','monitoreo') # Se conecta a la base de datos
86         cursor = conexion.cursor()
87
88         # Ejecutamos las funciones de cada sensor para almacenar el valor en la columna de la base de datos temperatura1, temperatura2 y
89         # temperatura3 respectivamente
90
91         valor1 = "INSERT INTO temperatura(temperatura1,temperatura2,temperatura3)VALUES(%s,%s,%s)" %(dametemp1(), dametemp2(),dametemp3())
92
93         cursor.execute(valor1)
94         conexion.commit()
95         cursor.close()
96         conexion.close() # Se cierra la conexion con la base de datos.
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113         time.sleep(600) # Introduce el tiempo de espera, en segundos
114
115
116 # Salir del programa
117 finally:
118     curses.nocbreak()
119     stdscr.keypad(0)
120     curses.echo()
121     curses.endwin() # Se cierran los cursores
122

```

Segmento de código 2.5. Código para tomar datos de temperatura de los sensores.

Para mayor detalle de todo el programa “temperatura_agua.py” revisar el Anexo I.

2.5.3 Programas adicionales

Además del programa explicado en la sección anterior, se debe usar dos programas adicionales. Un programa servirá para ejecutar el script de toma de datos de temperatura (“temperatura_agua.py”) automáticamente a penas se encienda la Raspberry Pi; y el otro programa servirá para encender o apagar la Raspberry Pi a través de un botón, ya que el

prototipo cuando se vaya a probar no contará con un monitor o periféricos para poder apagar o prender la placa de manera adecuada.

2.5.3.1 Programa Shutdown.

Se configura la Raspberry Pi para poder encenderla y apagarla a través de un botón conectado a los pines GPIO. Para la conexión se usará cuatro pines GPIO de la placa, dos pines para el botón, y los dos pines restantes para un led que indique el estado de la placa (encendida/apagada), en la Figura 2.22 se muestra la forma de conectar los cuatro pines.

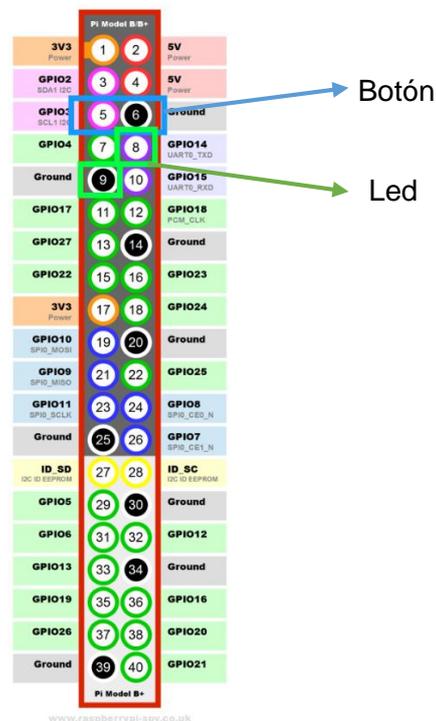


Figura 2.22. Pines usados para conectar el botón y led de encendido/apagado.

Establecido en donde se realizará la conexión, se configura la placa para que el botón y el led funcionen encendiendo y apagando la placa. Antes de crear el script, se instala librerías de Python para usar puertos GPIO, como se muestra a continuación:

```
sudo apt-get install python-dev python-rpi.gpio
```

Para configurar el botón y el led, se crea un script que se llamará "shutdown.py", en el script se escribe el código para apagar y encender la placa, además de encender el led. Se declara que el botón está en el GPIO 3, y que al presionarlo se apague o encienda la placa. También se declara que el led está en el GPIO 14, para que muestre el estado de la placa, como se muestra en el Segmento de código 2.6.

```
shutdown.py*
import RPi.GPIO as GPIO
from subprocess import call
GPIO.setmode(GPIO.BCM)

GPIO.setup(3, GPIO.IN)

#Indicador led.

GPIO.setup(14, GPIO.OUT)
GPIO.output(14, True)

GPIO.wait_for_edge(3, GPIO.FALLING)

call(['shutdown', '-h', 'now'], shell=False)
```

Segmento de código 2.6. Script Shutdown.py.

2.5.3.2 Automatización de los scripts

Para automatizar los dos scripts de Python, es decir para que se inicien apenas se encienda la Raspberry Pi de forma automática se modifica el archivo “*autostart*”, como se detalla a continuación.

Se debe editar la configuración de autoarranque de LXDE para el usuario. Se debe ingresar al archivo a través de la línea de comandos como se muestra:

```
sudo leafpad ~/.config/lxsession/LXDE-pi/autostart
```

Una vez abierto el archivo se añade dos líneas, una por cada script (una para el programa “*temperatura_agua.py*”, y el otro para el programa “*shutdown.py*”). La línea de comando que deberá aumentarse en los dos casos es:

```
/usr/bin/python /home/pi/”nombre del archivo”
```

El script “*temperatura_agua.py*” se abrirá y visualizará en la pantalla a penas se encienda la placa, así que se añade al inicio del comando anterior `@lxterminal -e` para que suceda esto; mientras que el script “*shutdown.py*” se ejecutará sin mostrarlo en pantalla. En la Figura 2.23 se observa el archivo “*autostart*”, con las dos líneas de comando adicionales.



```
autostart
File Edit Search Options Help
@lxpanel --profile LXDE-pi
@pcmanfm --desktop --profile LXDE-pi
@xscreensaver -no-splash
@point-rpi

@lxterminal -e /usr/bin/python /home/pi/temperatura_agua.py
@/usr/bin/python /home/pi/shutdown.py
```

Figura 2.23. Archivo “*autostart*” añadido los dos programas para que se inicien automáticamente.

2.5.4 Configuración del modem 4G

La placa Raspberry Pi se enlazará al modem HUAWEI Mobile WiFi E5573 para acceder a internet y subir los datos de temperatura tomados de la superficie del agua. El dispositivo usa la red LTE, 3G o 2G del operador, en este caso Claro, para acceder a internet, como se muestra en la Figura 2.24.

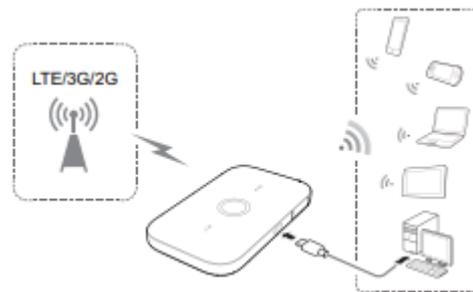


Figura 2.24. Mobile WiFi enlazándose a redes LTE/3G/2G para dar internet a dispositivos.

Al modem celular se le debe ingresar un chip para que por medio de este se enlace a la red celular. Además el chip debe tener algún plan contratado para conectarse a internet, además hay que asegurarse que la batería del dispositivo este cargada, para que pueda encenderse y funcionar sin ningún problema. Después, se coloca la batería como indica la Figura 2.25.

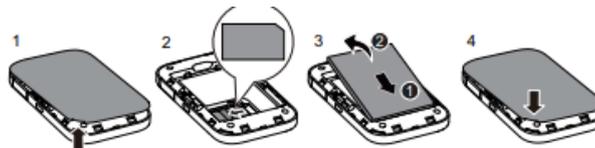


Figura 2.25. Conexión del chip y la batería al Mobile WiFi.

La conexión entre la Raspberry Pi y el dispositivo se la hará vía Wi-Fi, para esto se activa el Wi-Fi en la Raspberry Pi y se enlaza a la red con SSID (*Service Set Identifier*): HUAWEI-E5573-A310 cuya clave es: nqb3b878, estos parámetros vienen por defecto en el Mobile WiFi, como se ilustra en la Figura 2.26.

A continuación, se abre una ventana del navegador y se ingresa a <http://192.168.8.1> en el cuadro de direcciones. Para ingresar a la página pedirá usuario y contraseña (Figura 2.27.), el nombre de usuario y contraseña predeterminado es *admin* en los dos parámetros (Figura 2.26.).



Figura 2.26. SSID y clave del Mobile WiFi.

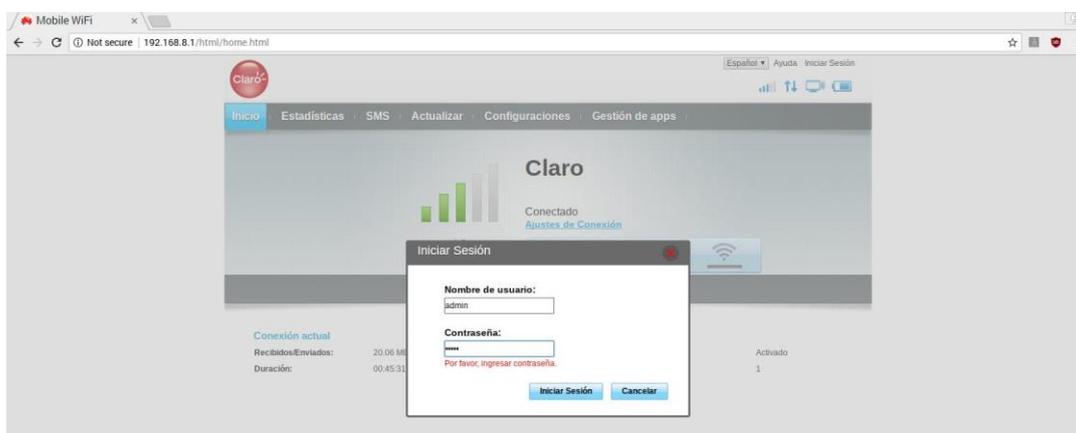


Figura 2.27. Iniciando sesión para administrar Mobile WiFi.

Una vez que se ingresa a la página, se muestra unas pestañas donde se puede cambiar algunos parámetros como las configuraciones básicas de WLAN, dentro de esto está el SSID y la contraseña que viene por defecto. Para tener más seguridad y evitar que otros dispositivos no deseado se conecten al Mobile WiFi, se cambiará el SSID a: Temperatura_Agua y la clave a: temperatura2875, como se muestra en la Figura 2.28.



Figura 2.28. Cambio de SSID y contraseña del Mobile WiFi.

Finalmente se selecciona la opción aplicar y se guardaran los cambios. Se deberá enlazar la placa Raspberry Pi nuevamente al dispositivo con la nueva SSID y contraseña.

2.5.5 Configuración de la base de datos.

La base que almacenará los datos de temperatura superficial del agua será creada en MySQL (*My Structured Query Language*, Lenguaje de Consulta Estructurado), y se la administrará a través de la herramienta phpMyAdmin.

phpMyAdmin es una herramienta escrita en PHP que permite la administración de bases de datos de MySQL a través de páginas web, ya sea de forma local o de forma remota a través de internet.

La base de datos se la llamará “monitoreo”, y dentro de la base de datos se creará una tabla llamada “temperatura”. En esta tabla se guardará la fecha y hora de cada dato de temperatura obtenido, además, del valor de temperatura de los tres sensores; por tanto la tabla tendrá cuatro columnas, una de fecha y hora, una de temperatura1 (temperatura a un metro), otra de temperatura2 (temperatura a tres metros) y temperatura3 (temperatura a cinco metros), como se observa en la Figura 2.29.

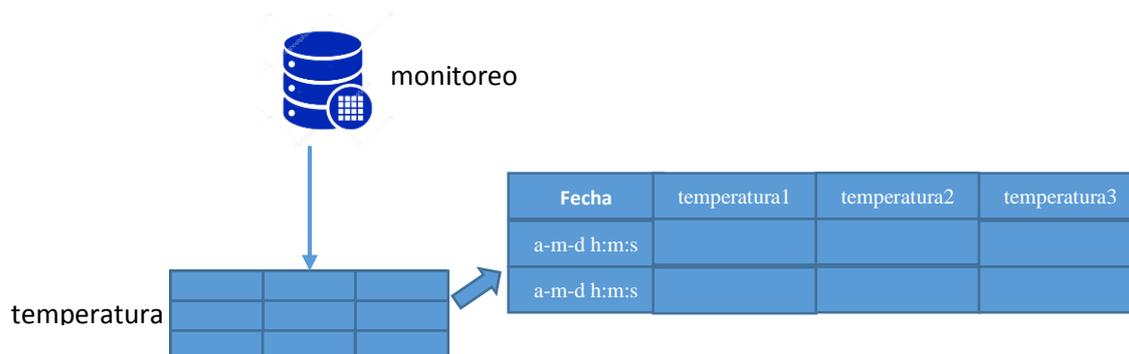


Figura 2.29. Estructura de la base de datos “monitoreo”.

2.5.5.1 Instalación MySQL y phpMyAdmin.

Primero se instala el servicio de MySQL en el servidor con el siguiente comando:

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

Después, se reinicia los servicios con los siguientes comandos:

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

A continuación se instala la librería de Python para usar con MySQL:

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

El siguiente paso es instalar phpMyAdmin, se lo hace ingresando en la línea de comandos lo siguiente:

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

Finalmente, ya instalado los servicios y la base de datos, se debe crear un usuario y contraseña para poder ingresar a través de phpMyAdmin. El usuario que se crea es “administrador” y la clave es “c2875”, esto se lo hace con los siguientes comandos:

```
create user 'administrador'@'localhost' identified by 'c2875';  
grant all privileges on *.* to 'administrador'@'localhost' with grant option;  
flush privileges;  
exit
```

2.5.5.2 Creación de la base de datos con phpMyAdmin

Para crear la base de datos, se debe ingresar a una ventana del navegador con la dirección del servidor además de phpMyAdmin, así por ejemplo: <http://192.168.X.X/phpmyadmin>. En la Figura 2.30 se muestra la página de bienvenida de phpMyAdmin, donde se ingresa usuario y contraseña.

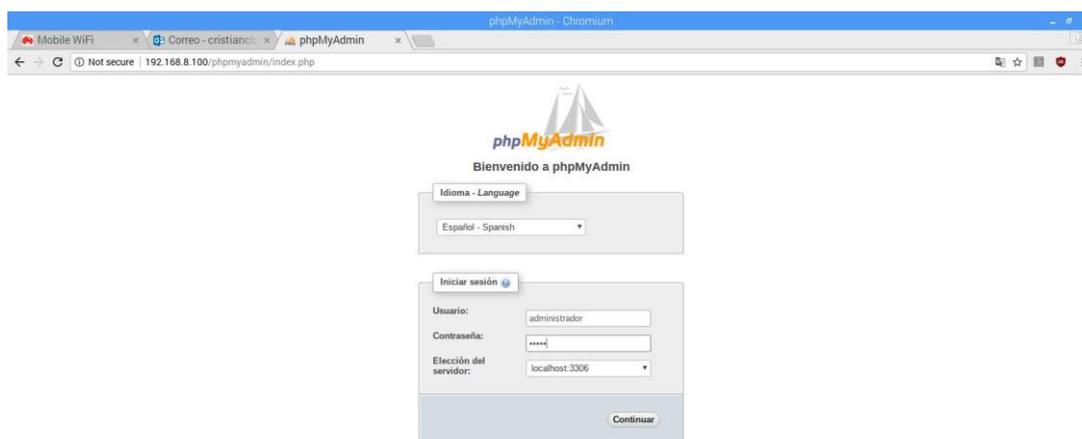


Figura 2.30. Página de bienvenida phpMyAdmin.

Una vez identificados, se ingresará a la página principal donde se podrá realizar varias actividades, entre ellas, crear una nueva base de datos; para esto se selecciona en “Nueva” como se ilustra en la Figura 2.31.

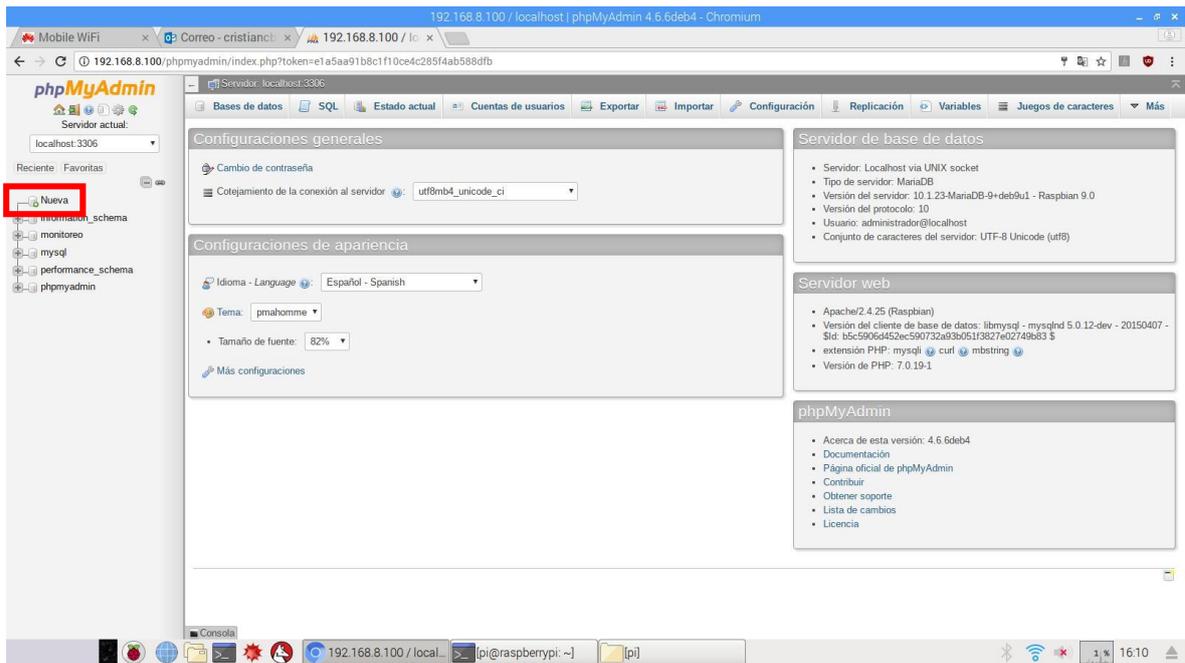


Figura 2.31. Página principal de phpMyAdmin.

Después de seleccionar la opción “Nueva”, se debe llenar ciertos parámetros de configuración, como nombre de la base; para el proyecto se llamará “monitoreo”. Después se pulsa “crear”, como se observa en la Figura 2.32.

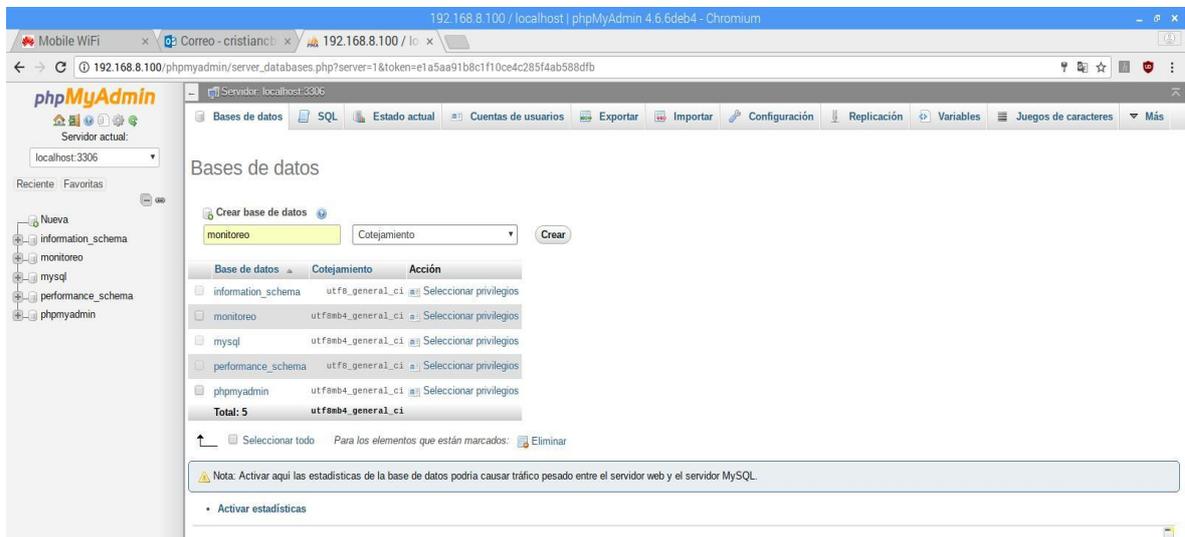


Figura 2.32. Creación de base de datos “monitoreo”.

Dentro de la base de datos “monitoreo” se crea una tabla llamada “temperatura”, la misma que tendrá cuatro columnas como se ilustra en la Figura 2.33.

Después se tiene que configurar el tipo de datos que se va almacenar en cada columna. El campo fecha es de tipo *TIMESTAMP* y como valor predeterminado va a tener

CURRENT_TIMESTAMP, esto quiere decir que cuando se inserte cualquier valor en la tabla, se almacenará también la hora automáticamente. Los otros tres campos (*temperatura1*, *temperatura2* y *temperatura3*) son de tipo *Double* como se observa en la Figura 2.34.

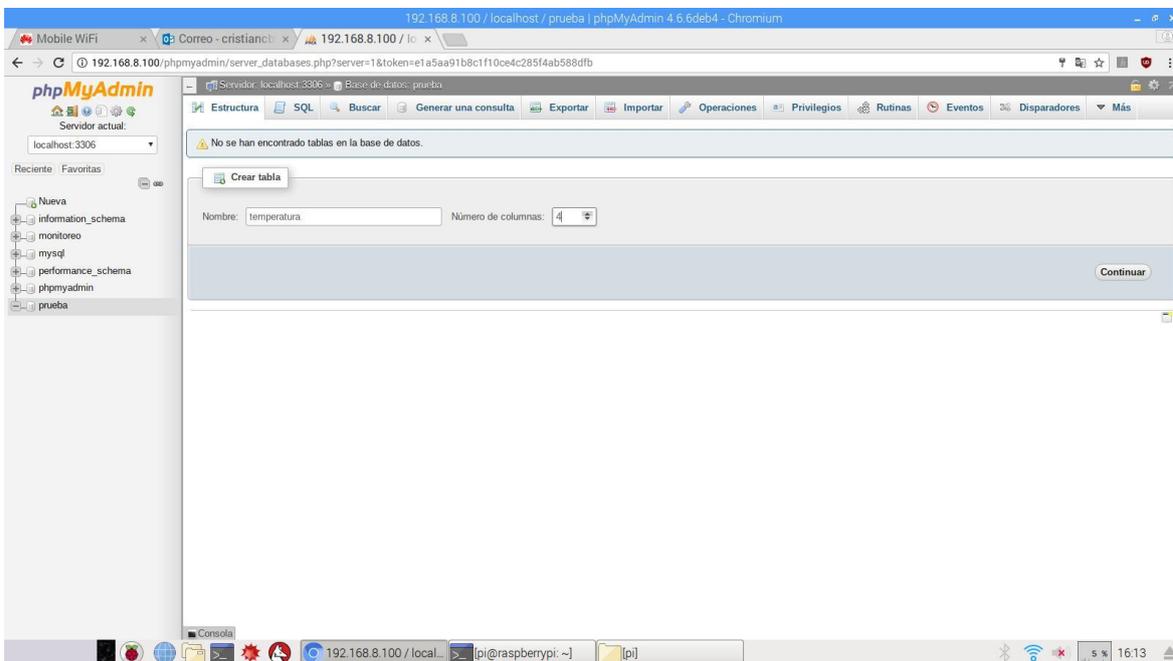


Figura 2.33. Creación de la tabla “temperatura” en la base de datos.

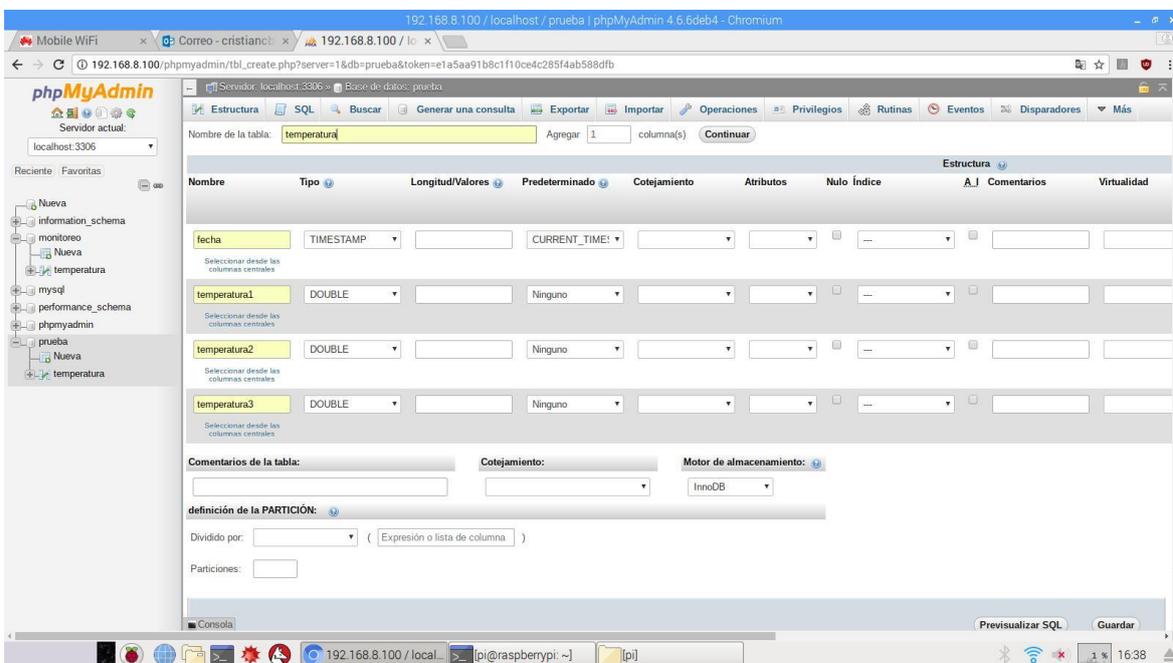


Figura 2.34. Configuración de cada columna de la tabla “temperatura”.

2.5.6 Desarrollo sitio web

Para el desarrollo del sitio web, donde se mostrarán datos obtenidos del prototipo, se usará el programa **Adobe Dreamweaver CC 2015**, ya que permite crear sitios web desde una interfaz completamente gráfica y sin conocimiento avanzado acerca de desarrollo web.

Dreamweaver es un programa que lleva años siendo uno de los mejor valorados dentro del campo de la programación web tanto por profesionales como por aficionados. Dreamweaver facilita enormemente la tarea de realizar sitios web complejos y bien estructurados. Se puede diseñar páginas con un interfaz completamente gráfica mientras se observa simultáneamente el código generado. Incorpora numerosas herramientas para el tratamiento de tablas y otros componentes avanzados, así como la inserción de objetos no HTML, como películas flash o javascript y programas en PHP [61].

Al abrir Dreamweaver CC 2015 se encontrará con una página de inicio y una barra de opciones en la parte superior, en esta barra se escoge la opción “Sitio:” y después se selecciona “Nuevo sitio”, como se observa en la Figura 2.37.

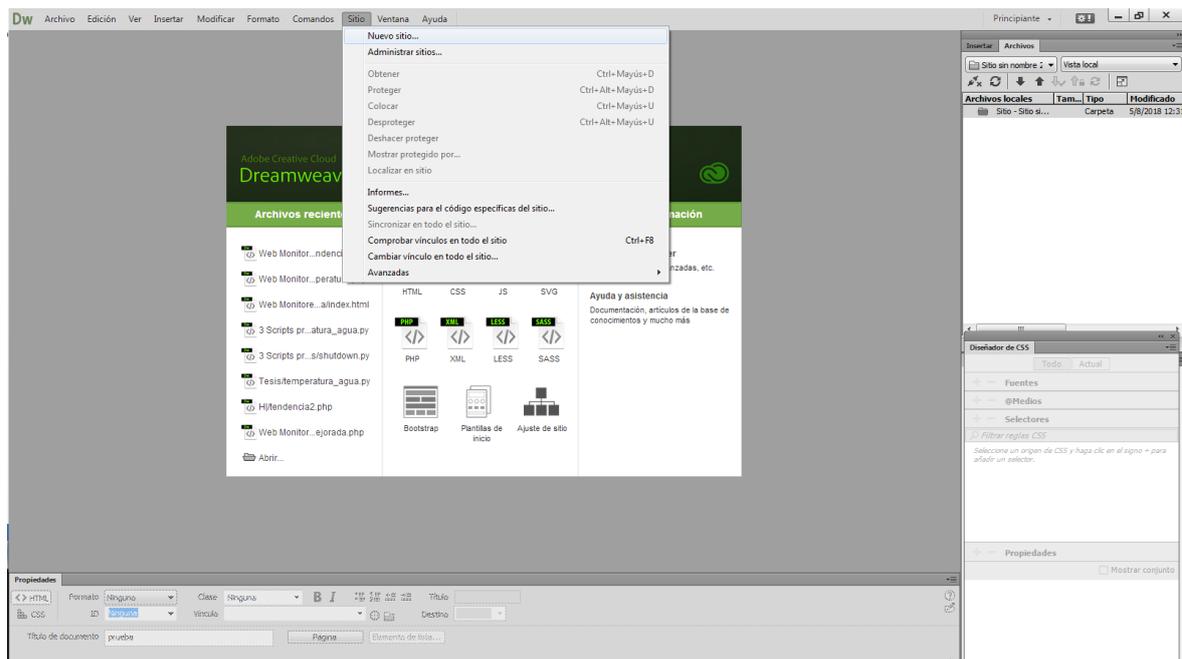


Figura 2.37. Página de inicio Adobe Dreamweaver CC 2015.

A continuación, se abrirá una ventana con cuatro opciones: “Sitio”, “Servidores”, “Control de versión” y “Configuración Avanzada”. Se configura solo la primera opción “Sitio”, donde se pone un nombre al proyecto “Web Monitoreo de Temperatura del agua”, además se coloca la dirección de la carpeta en donde se almacenará el proyecto, que será el escritorio de la PC, como se observa en la Figura 2.38.

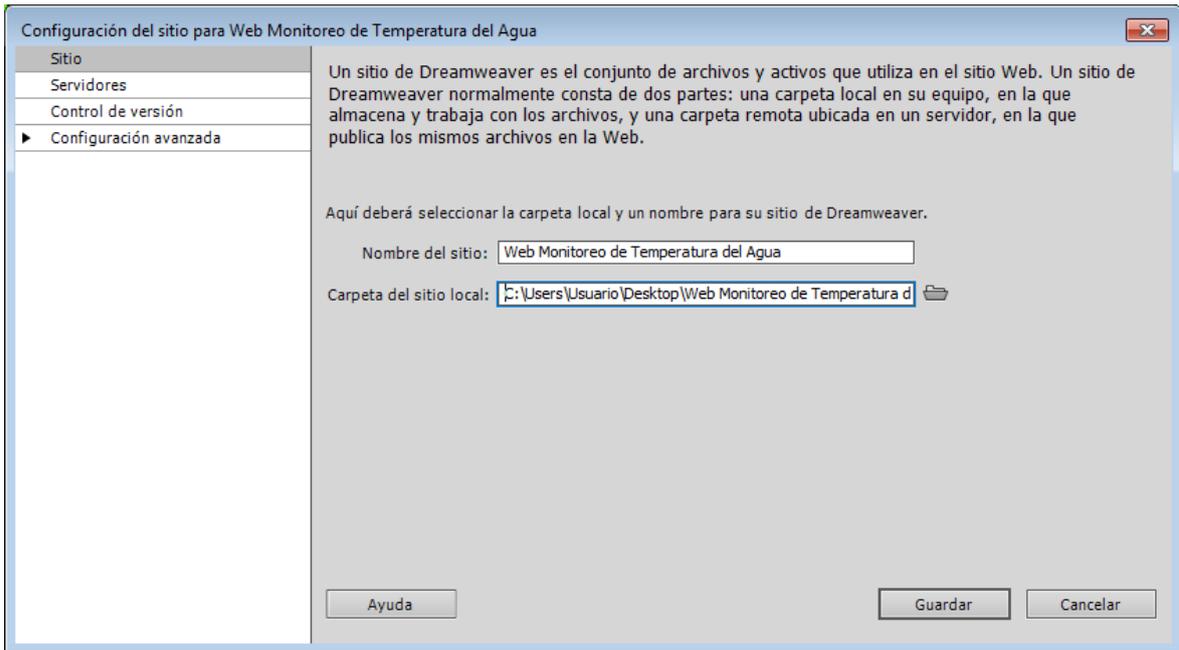


Figura 2.38. Configuración del sitio “Web Monitoreo de Temperatura del Agua”.

Después se debe crear el primer archivo, que será la primera página web que se denomina “index” o “inicio”. Para crear el “index” se selecciona “crear nuevo HTML” como se muestra en la Figura 2.39. Se debe fijar en la ventana superior derecha, que esté dentro del sitio web creado antes (“Web Monitoreo de Temperatura del Agua”).

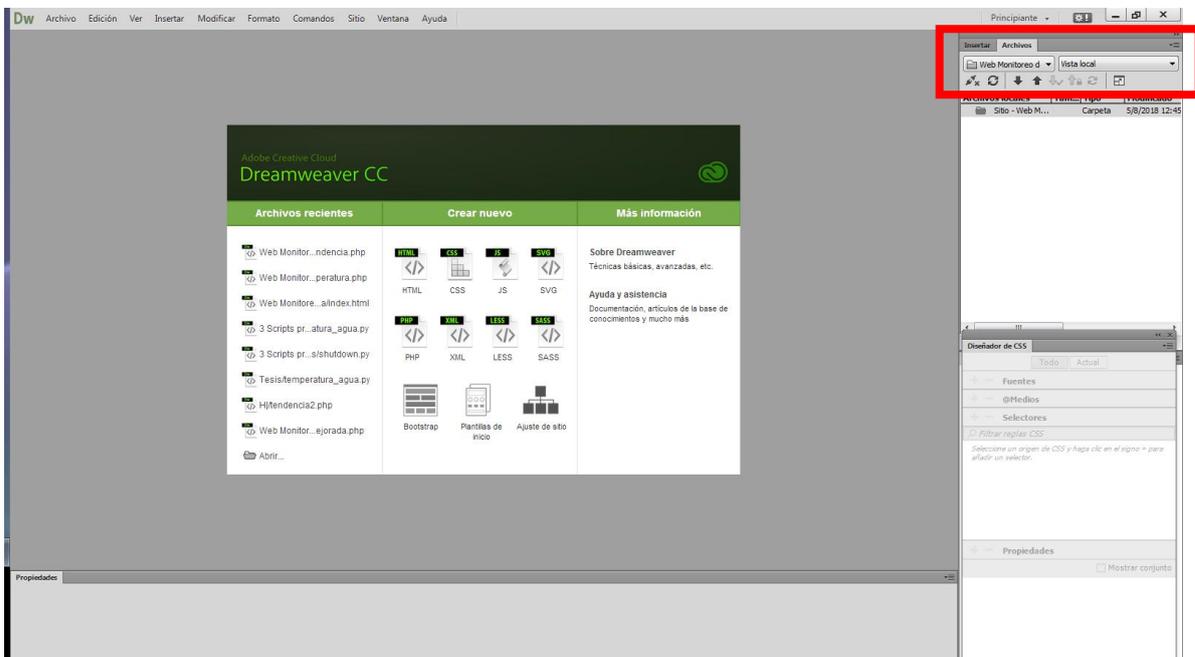


Figura 2.39. Página para crear archivos nuevos dentro de un sitio creado.

Una vez seleccionado “nuevo documento HTML”, se pone el título de “Index” y el “tipo de documento” que es HTML5, que viene seleccionado por defecto, como se muestra en la Figura 2.40.

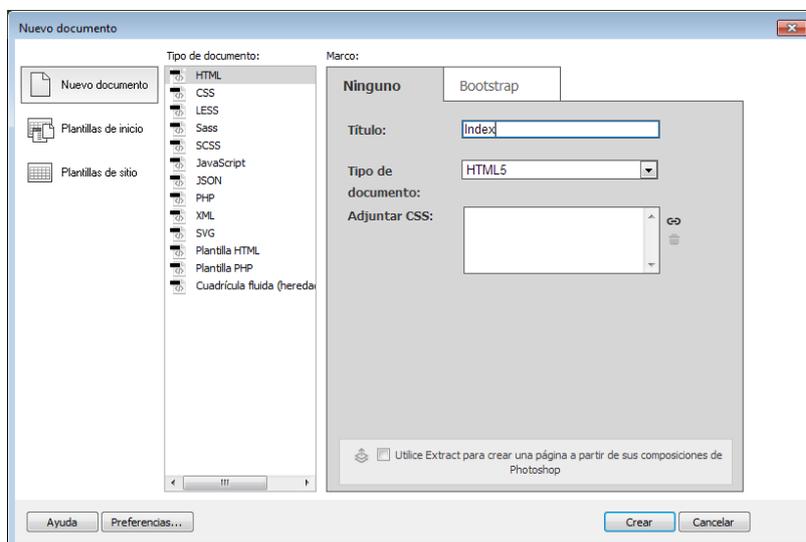


Figura 2.40. Creación de nuevo documento dentro del sitio web.

Finalmente, se creará el contenido de todo el sitio web, como se observa en la Figura 2.41, se tiene listo todas las opciones y herramientas que brinda Dreamweaver para crear la página web de acuerdo a los requerimientos.

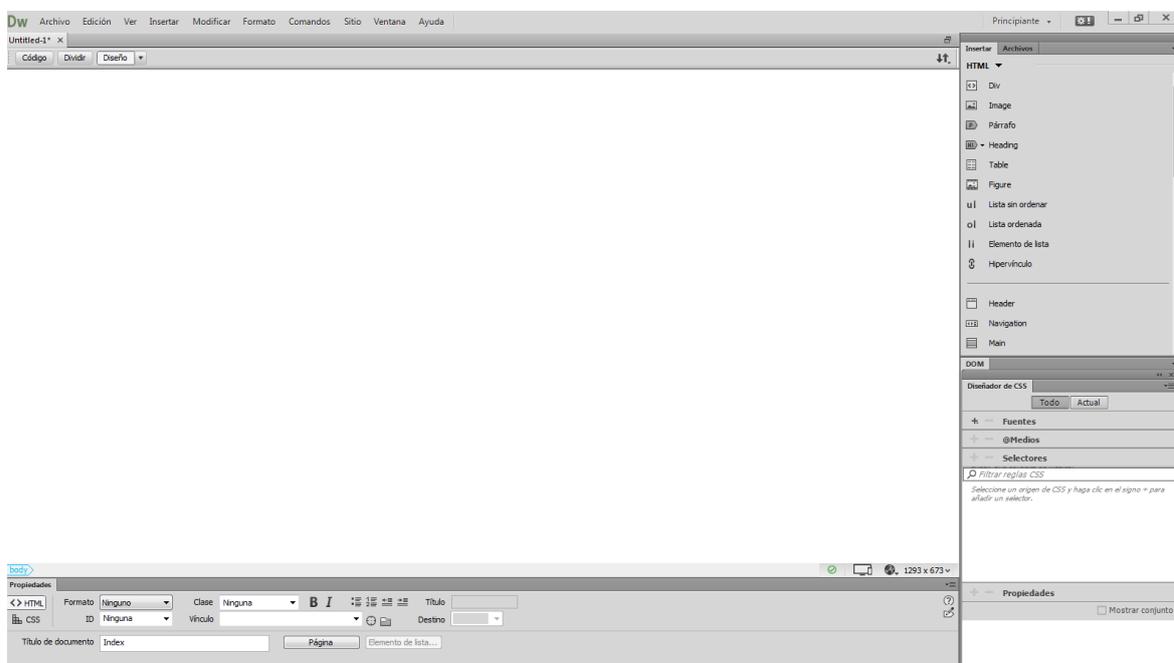


Figura 2.41. Pantalla de trabajo para crear el contenido de la página web.

El sitio web contará con tres páginas web, la primera que es el “index” y que muestra información general acerca del proyecto, la segunda página web mostrará la temperatura superficial del agua a un metro, tres metros y cinco metros con su respectiva estampa de tiempo, y finalmente habrá una página web donde se mostrará gráficas de tendencia de la temperatura para cada sensor.

Las tres páginas web tendrán la misma estructura, estarán divididas en tres partes con la herramienta *Div*, que permite hacer divisiones lógicas dentro de la página web. La primera división lógica será el menú de opciones donde se encontrará botones que direccionaran a las tres páginas del sitio (inicio, temperatura y tendencias), además de un enlace a redes sociales como Twitter y Facebook. La segunda parte lógica será la “carátula” o “*header*” del sitio web. Y finalmente la tercera división lógica será el contenido como tal de cada página web, como se muestra en la Figura 2.42.

Además se establece el tamaño de cada bloque para que sea legible y agradable al usuario. Se utilizó medidas estándar, usadas en sitios web; estas medidas están en unidades de pixeles. Las medidas en pixeles también se muestran en la Figura 2.42.

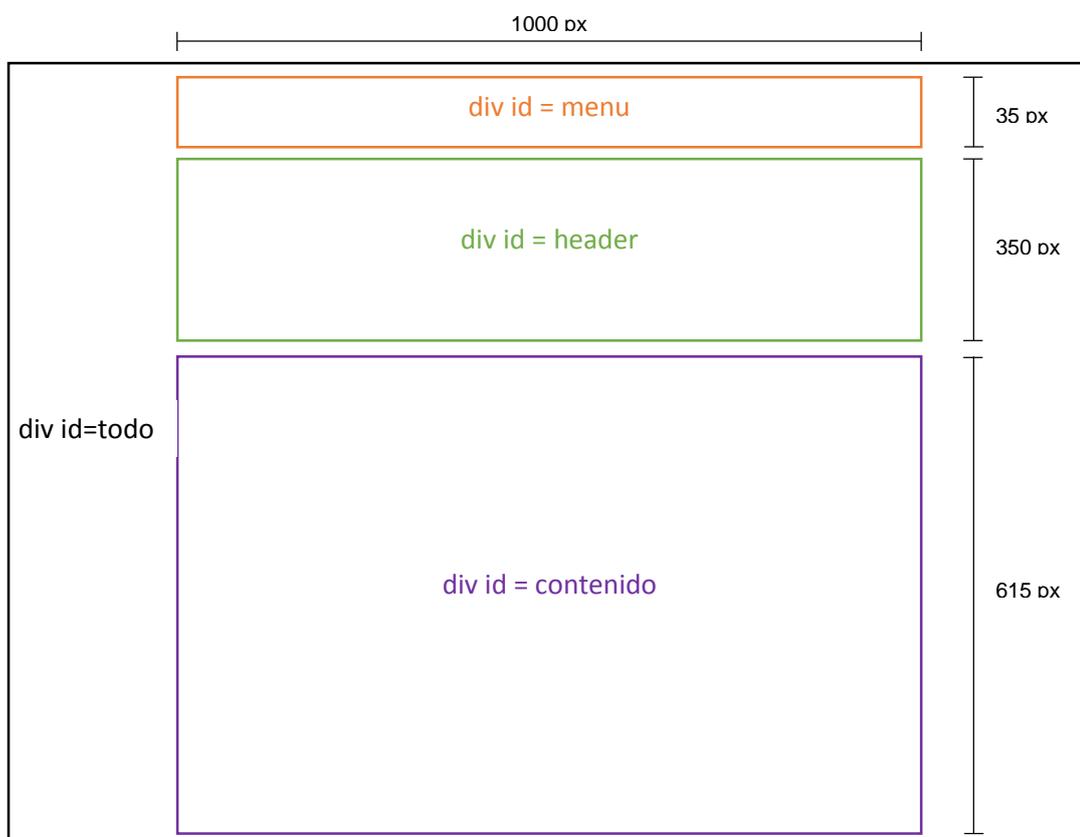


Figura 2.42. Divisiones lógicas de las páginas web con sus medidas en pixeles.

2.5.6.1 Desarrollo del Header

Para desarrollar el “header” o “carátula” del sitio web, se usó la aplicación “Canva”, que es una herramienta de diseño para crear imágenes web, logotipos, tarjetas de presentación, invitaciones, etc. Además, esta herramienta es totalmente gratuita y fácil de usar para personas que no tienen gran conocimiento en diseño gráfico y programas más especializados como Photoshop o Illustrator. Esta herramienta se puede usar totalmente en línea, es decir, sin necesidad de descargar e instalarla en la PC, simplemente se usa ingresando a la dirección web y creando una cuenta. Creada la cuenta, se puede usar todas las plantillas y herramientas de “Canva”, como se observa en la Figura 2.43.

A continuación, se crea el diseño del “header” o “carátula” del sitio web, que se usará en las tres páginas web. Se diseñó tratando de que sea agradable al usuario, mostrando la información necesaria como el título de “Escuela Politécnica Nacional” con su logo y otro título de lo que trata el sitio “Temperatura Superficial del Agua”, como se aprecia en la Figura 2.44.

La imagen de la carátula diseñada se la guarda en la dirección del sitio web, dentro de una carpeta llamada imágenes. Además, la herramienta “Canva” ayudará para la creación de los botones usados en el menú de las páginas web.

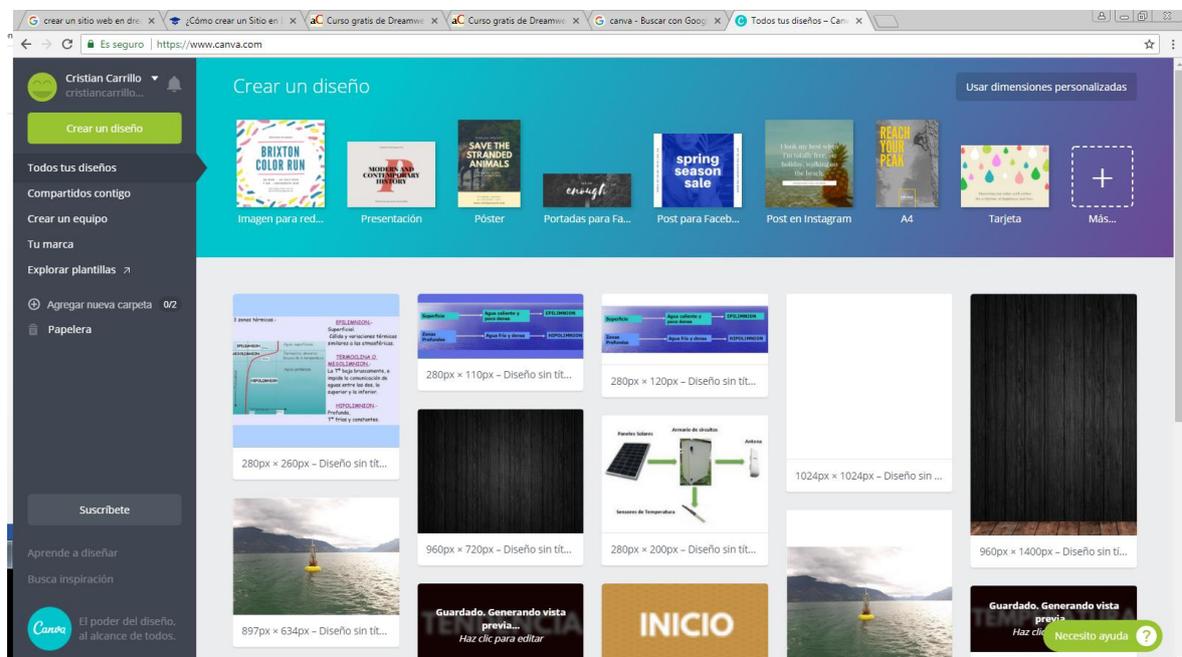


Figura 2.43. Página de inicio de la herramienta de diseño Canva.



Figura 2.44. Diseño del “header” o “carátula” del sitio web.

2.5.6.2 Desarrollo página web inicio

La primera página será la de “index” o “inicio” y tendrá contenido estático, es decir, siempre mostrará la misma información acerca del proyecto, esta no cambiará.

Se sigue el formato que se estableció en la Figura 2.42, con los divisores lógicos y el tamaño de cada sección en pixeles.

Como se observa en la Figura 2.45, se tiene las secciones establecidas, con el tamaño indicado como se muestra en el recuadro verde de la imagen. También se puede observar en el recuadro rojo, que dentro del divisor lógico “todo” se encuentra las divisiones lógicas “menú”, “header” y “contenido” como se estableció en un inicio.

Cada una de las secciones lógicas, se irán llenado con la información necesaria, por ejemplo, en la sección menú irán los botones que direccionaran a las tres páginas web que existirán; en el *header* estará la carátula que se realizó previamente en “Canva”; y en el contenido irá información general acerca del proyecto. Cabe mencionar que este formato mostrado, e incluso la información del menú y *header* será la misma para las tres páginas; solo cambiará el contenido del divisor lógico “contenido”.

Así por ejemplo, para insertar la imagen creada en “Canva” se selecciona la sección “header” y en la ventana de opciones ubicada a la derecha, se selecciona “imagen” como se muestra en el recuadro rojo de la Figura 2.46. Después, se abrirá una ventana en la cual

se podrá seleccionar la imagen diseñada previamente, y que fue guardada en el directorio del sitio web, ver Figura 2.46.

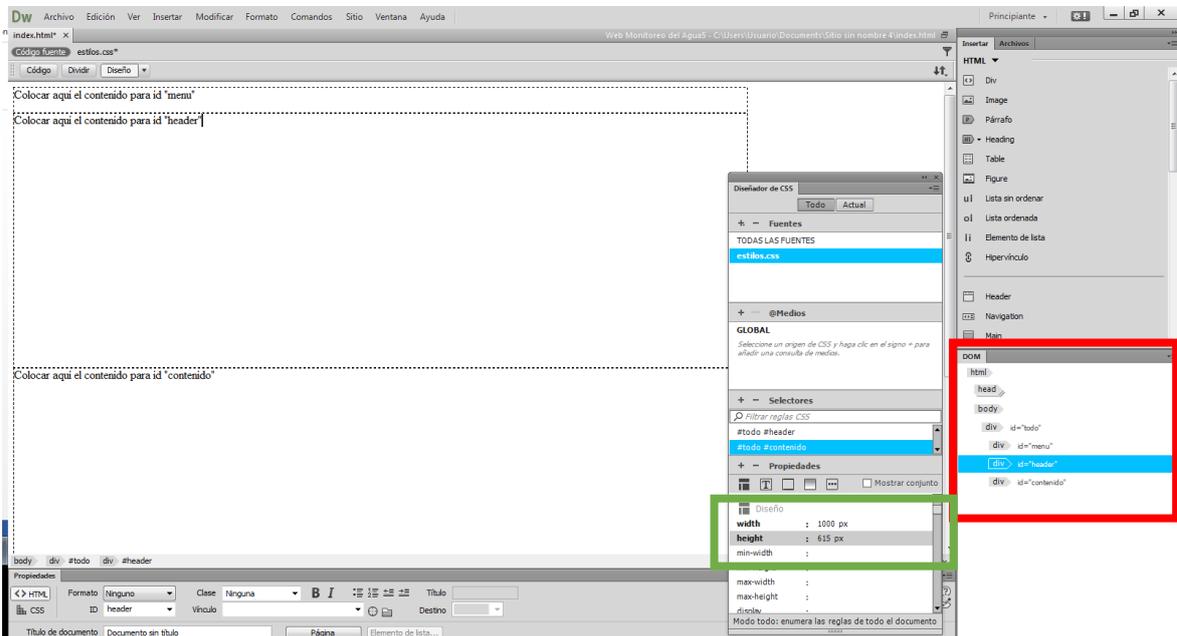


Figura 2.45. Construcción de la página web con los divisores lógicos.

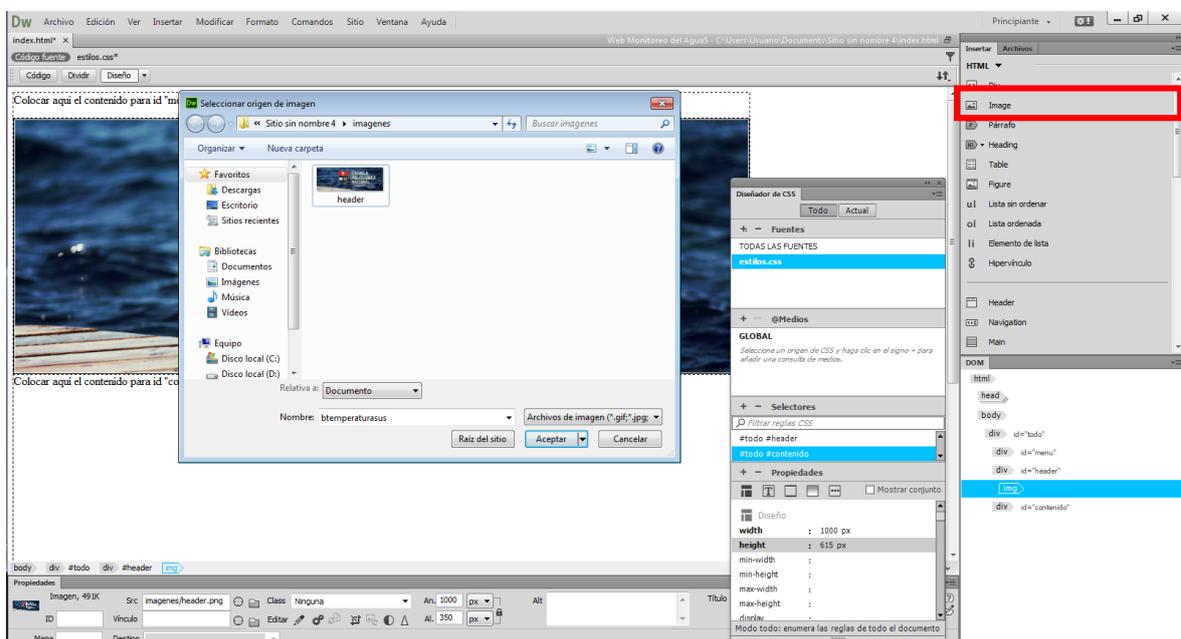


Figura 2.46. Insertar imagen en el *Header* de la página web.

A continuación se pone colores de fondo tanto en el índice (color rojo), en el contenido (plomo con blanco) y al fondo de toda la página que es un color oscuro, como se observa en la Figura 2.47. Con esto se busca que la página sea agradable al usuario.



Figura 2.47. Diseño de las páginas web.

En la parte de menú, se añade botones que direccionaran a las tres páginas existentes en el sitio web (inicio, temperatura y tendencia), además se añade dos botones que direccionarán a redes sociales como Twitter y Facebook, como se observa en la Figura 2.48. El diseño de los botones se realizó con la ayuda de la herramienta “Canva”, al finalizar el diseño se guardó en el directorio del sitio web para solo añadir las imágenes de los botones. Para poder tener un vínculo entre los botones y su respectiva página web, se debe tener creadas las otras páginas web faltantes.



Figura 2.48. Colocación de botones en menú.

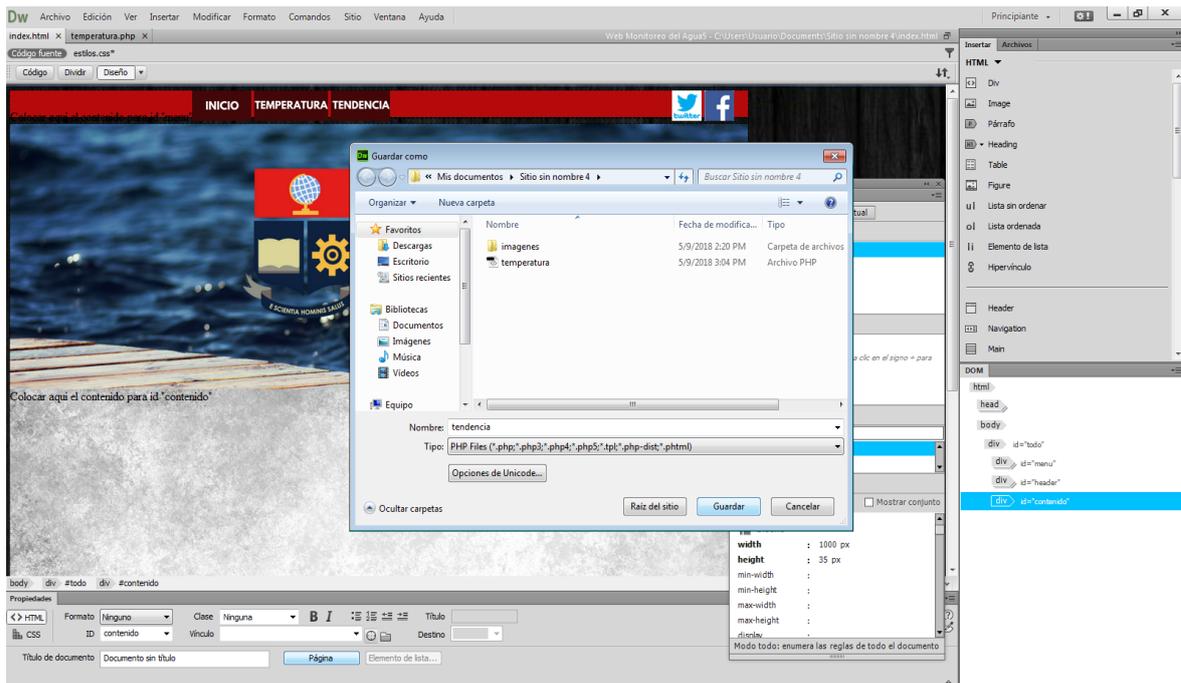


Figura 2.49. Guardando las páginas temperatura.php y tendencia.php

El formato mostrado en la Figura 2.48 será igual para las tres páginas web, ya que como se mencionó, solo se modificará el divisor lógico “contenido” de cada página. Para no hacer todos los pasos anteriores en cada página, simplemente se usa el formato ya diseñado de esta página. Se usa la opción “guardar como” de la página ya diseñada para almacenar dos páginas web que serán “temperatura.php” y “tendencia.php” como se observa la Figura 2.49.

Las dos páginas tendrán contenido dinámico y esto se lo hace a través de PHP; por tanto, tendrán extensión .php así: “temperatura.php” y “tendencia.php”, a diferencia de la página de “inicio” o “index” que es “index.html”.

Como se observa en el recuadro rojo de la Figura 2.50, se tiene las tres páginas web almacenadas en el directorio del sitio web, se tiene “inicio.html”, “temperatura.php” y “tendencia.php” listas con el formato, fondos y botones; listas para ser configuradas con el contenido único de cada página, como se observa en la Figura 2.50.

Finalmente para acabar de configura la página de inicio, se añade el contenido estático que es la información general del proyecto. En la Figura 2.51 se muestra el aspecto final de la página web “index.html”.



Figura 2.50. Modelo de las tres páginas web.

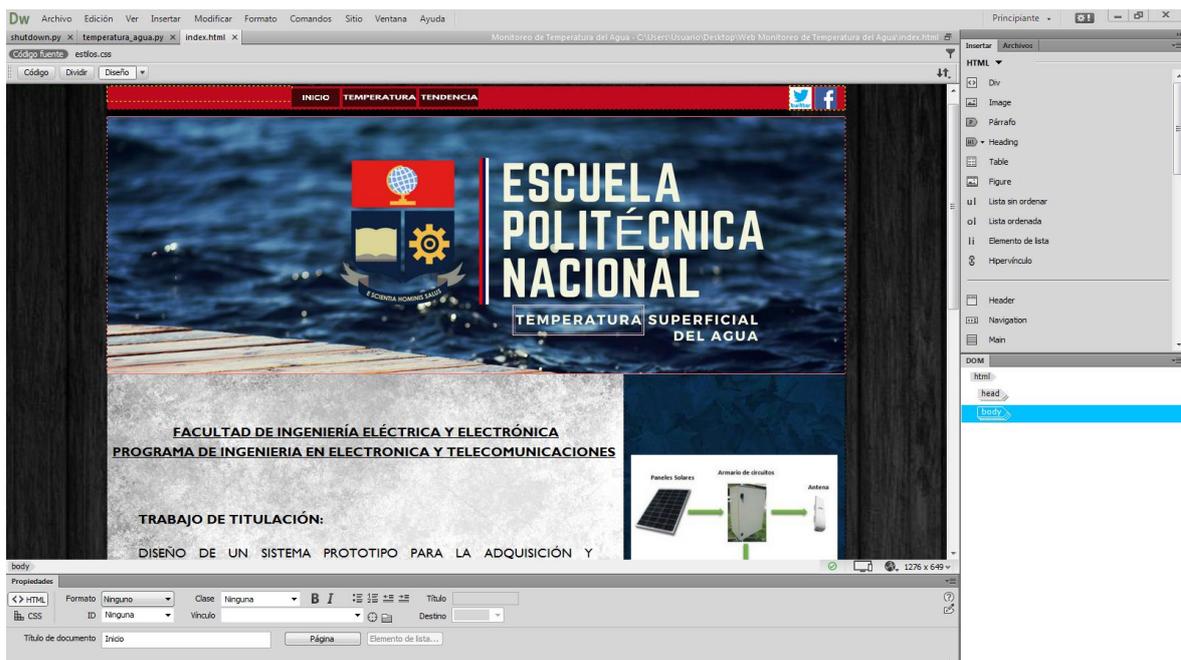


Figura 2.51. Visualización página web "index.html".

2.5.6.3 Desarrollo página web temperatura.php

La página web "temperatura.php" tendrá contenido dinámico, ya que se irá actualizando cada 10 minutos con la temperatura superficial tomada por los sensores. Esta página además de mostrar la temperatura, mostrará una tabla con valores mínimos, máximos y promedios de cada sensor.

Como ya se tiene la página creada con el formato, solo se edita la parte de “contenido” ingresando los títulos y textos de acuerdo a como se desee que aparezcan, es decir, ingresando tipo de letra, tamaño y espacios.

Después, se deja los espacios en donde se va a mostrar el contenido dinámico con PHP, como se observa en la Figura 2.52 donde aparecen unas etiquetas PHP en los recuadros. El contenido dinámico mostrará la información que se tiene en la base de datos; además de mostrar ciertos cálculos como valores mínimos, máximos y promedios de la base.

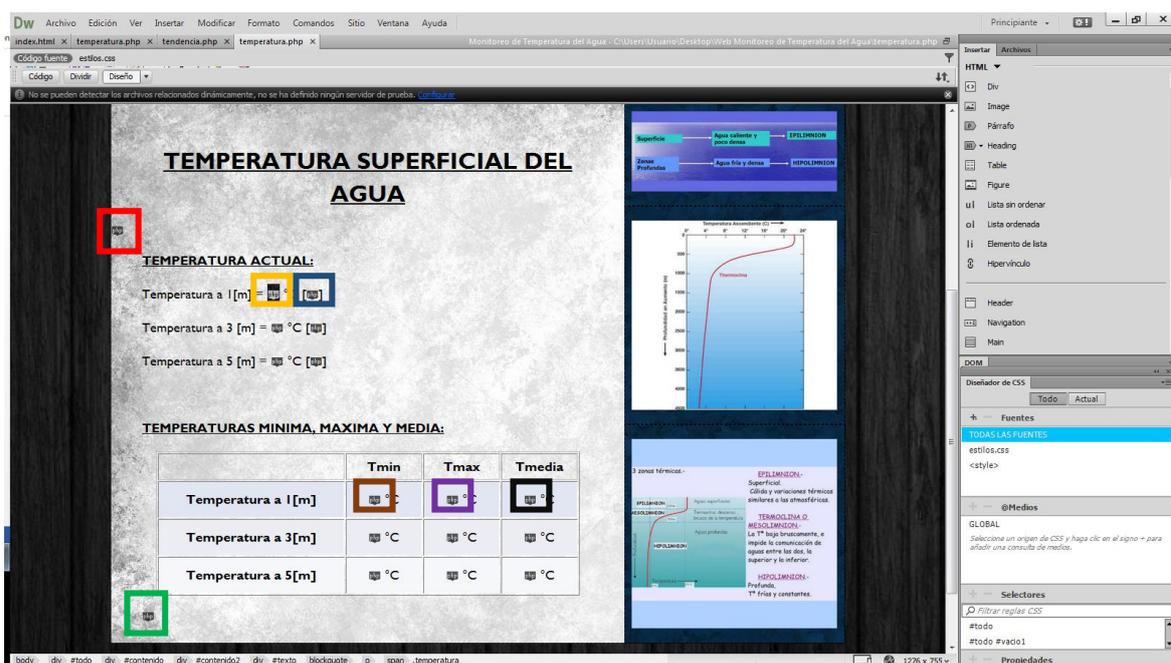
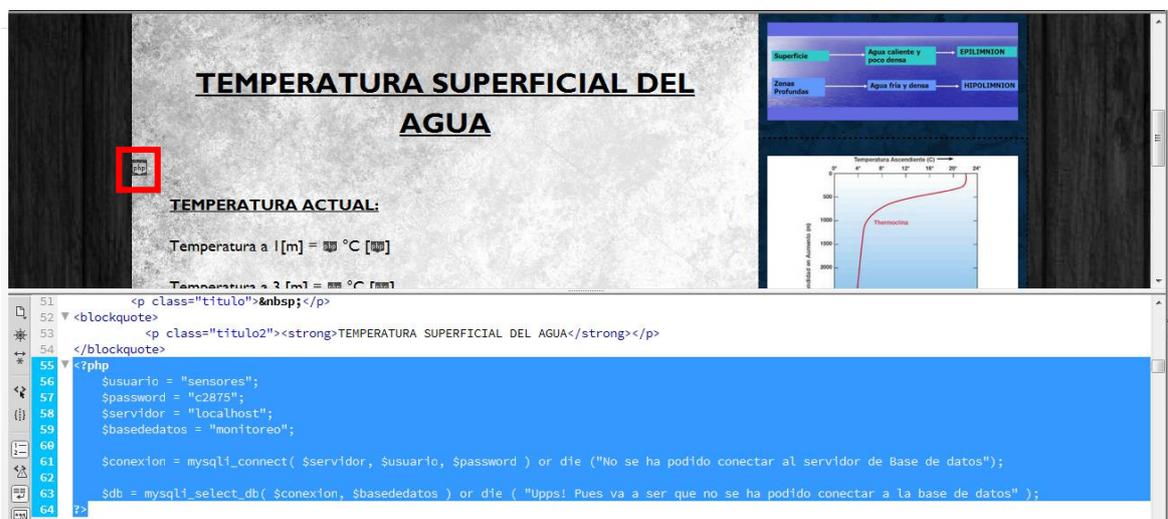


Figura 2.52. Página web “temperatura.php”

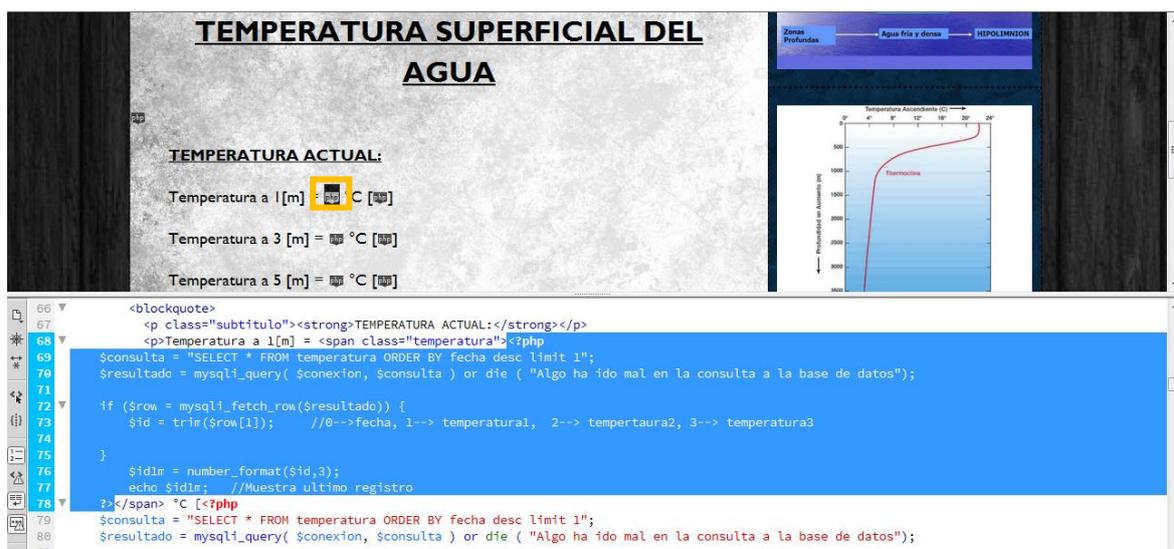
Para el contenido dinámico, el primer paso es establecer la conexión con la base de datos. Sin una conexión creada no se podrá insertar o mostrar valores de la tabla de la base. En la conexión se configura ciertos valores como usuario, *password*, servidor y a que base de datos se va a conectar, como se observa en el Segmento de código 2.7. Una vez hecho esto, se podrá trabajar con la base de datos en la página web.

El siguiente paso es tomar los datos de la base mediante una iteración. Primero se ordena la base de datos desde el último valor ingresado hasta el primero; para así obtener los últimos datos tomados por los sensores. Se usa un bucle “*if*” para buscar el dato que se quiere (fecha, temperatura1, temperatura2 o temperatura3) dentro de esta última fila de datos; se lo hace con el comando “*row [0]*” si es 0 se obtiene la primera fila que es la fecha y hora, si es 1 la temperatura del sensor a un metro, si es 2 la temperatura del sensor a tres metros y si es 3 la temperatura del sensor a 5 metros, como se observa en la Segmento

de código 2.8. Finalmente se hace que el valor se muestre con tres decimales (number_format()).



Segmento de código 2.7. Conexión a la base de datos.



Segmento de código 2.8. Tomar datos de temperatura de la base y mostrar en la página web.

Se usa el mismo código para obtener la temperatura de los otros sensores y la estampa de fecha y hora como se observa en la Segmento de código 2.9.

Para obtener valores mínimos, máximos y promedio existen comandos en PHP, que ayudan mucho para estos cálculos. Se realiza una consulta a la base y se pide que devuelva el valor mínimo, máximo o promedio de una columna (en este proyecto sea de la temperatura a un metro, tres metros o cinco metros), como se observa en el segmento de código 2.10.

TEMPERATURA SUPERFICIAL DEL AGUA

TEMPERATURA ACTUAL:

Temperatura a 1 [m] = °C

Temperatura a 3 [m] = °C

Temperatura a 5 [m] = °C

```

74 }
75
76     $id1m = number_format($id,3);
77     echo $id1m; //Muestra ultimo registro
78 }></span> °C [?php
79 $consulta = "SELECT * FROM temperatura ORDER BY fecha desc limit 1";
80 $resultado = mysqli_query( $conexion, $consulta ) or die ( "Algo ha ido mal en la consulta a la base de datos");
81
82 if ($row = mysqli_fetch_row($resultado)) {
83     $id = trim($row[0]); //0-->fecha, 1--> temperatura1, 2--> temperatura2, 3--> temperatura3
84
85 }
86     echo $id; //Muestra ultimo registro
87 }></p>
88     <p>Temperatura a 3 [m] = <span class="temperatura">?php

```

Segmento de código 2.9. Tomar datos de fecha y hora de la base y mostrar en la página web.

Temperatura a 5 [m] = °C

TEMPERATURAS MINIMA. MAXIMA Y MEDIA:

	Tmin	Tmax	Tmedia
Temperatura a 1 [m]	<input type="text" value="23.5"/> °C	<input type="text" value="23.5"/> °C	<input type="text" value="23.5"/> °C
Temperatura a 3 [m]	<input type="text" value="23.5"/> °C	<input type="text" value="23.5"/> °C	<input type="text" value="23.5"/> °C

```

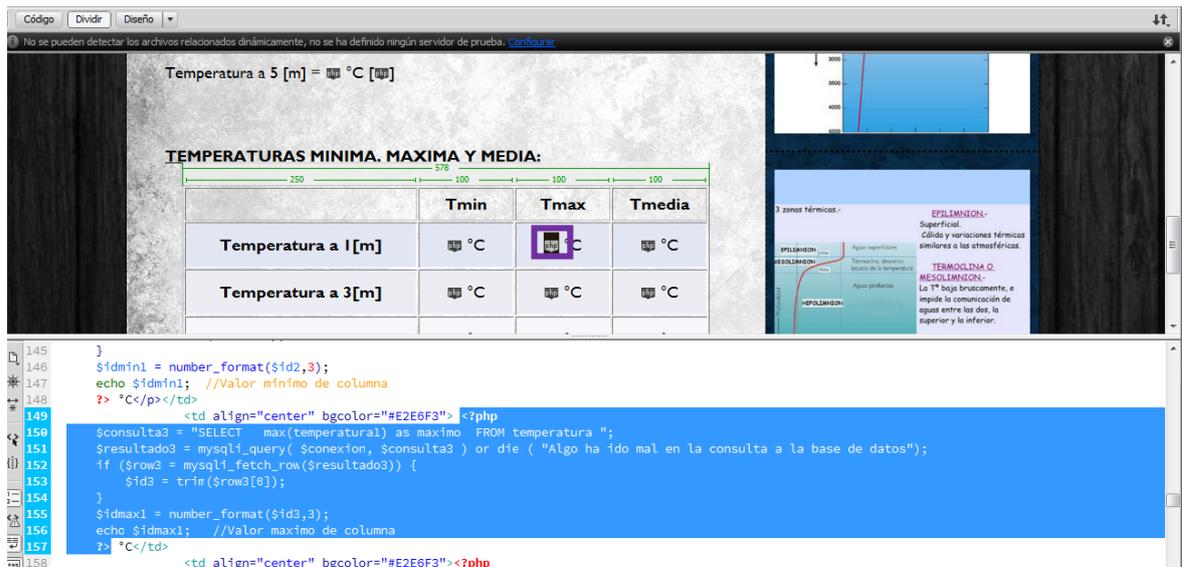
135 <th width="100" align="center" scope="col">Tmax</th>
136 <th width="100" align="center" scope="col">Tmedia</th>
137 </tr>
138 <tr>
139 <th height="50" align="center" bgcolor="#E2E6F3" scope="row"><strong>Temperatura a 1[m]</strong></th>
140 <td align="center" bgcolor="#E2E6F3">?php
141 $consulta2 = "SELECT min(temperatura1) as minimo FROM temperatura "; //Seleccionar de cual queremos, temperatura1, temperatura2, temperatura3.
142 $resultado2 = mysqli_query( $conexion, $consulta2 ) or die ( "Algo ha ido mal en la consulta a la base de datos");
143 if ($row2 = mysqli_fetch_row($resultado2)) {
144     $id2 = trim($row2[0]);
145 }
146 $idmin1 = number_format($id2,3);
147 echo $idmin1; //Valor minimo de columna
148 }> °C</p></td>

```

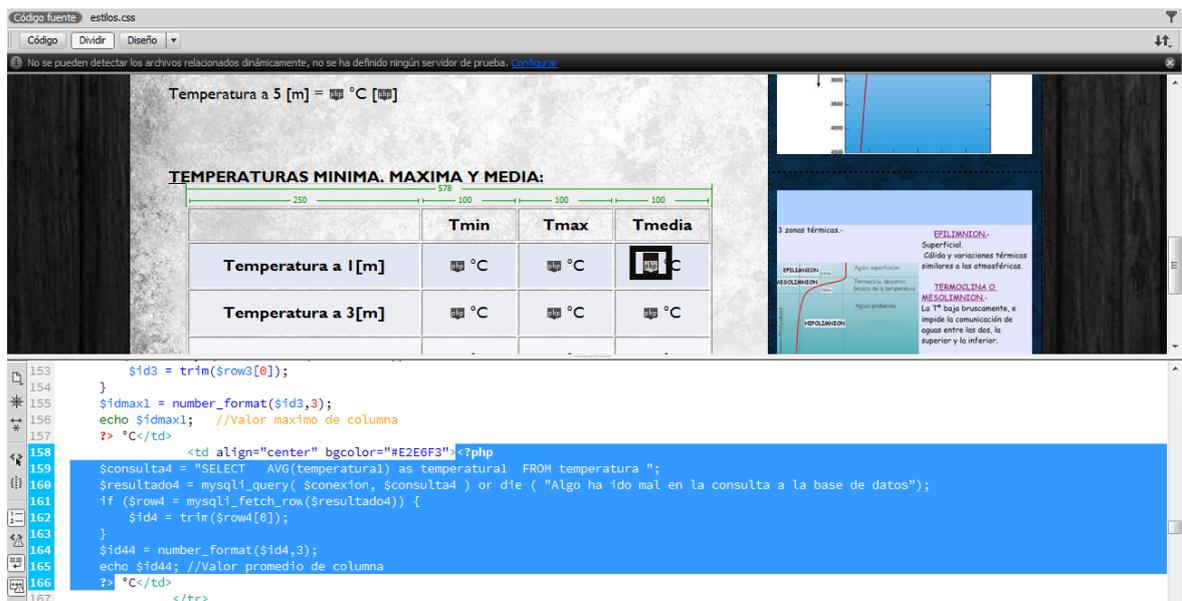
Segmento de código 2.10. Sacando el valor mínimo de una columna y mostrándolo en la página web.

De la misma manera para sacar el valor máximo de una columna se realiza la consulta a la base y se pide que devuelva el valor máximo, en este ejemplo de la columna "temperatura1" que es la columna de los valores obtenidos por el sensor a 1 metro de profundidad (Segmento de código 2.11.)

Tal como se hizo con los valores mínimos y máximos, se realiza algo semejante para obtener el valor promedio de una columna, se cambia de comando a "AVG (average)" de PHP para obtener el valor promedio de la columna, que son los valores obtenidos del sensor a un metro de profundidad, como se observa en la Segmento de código 2.12.



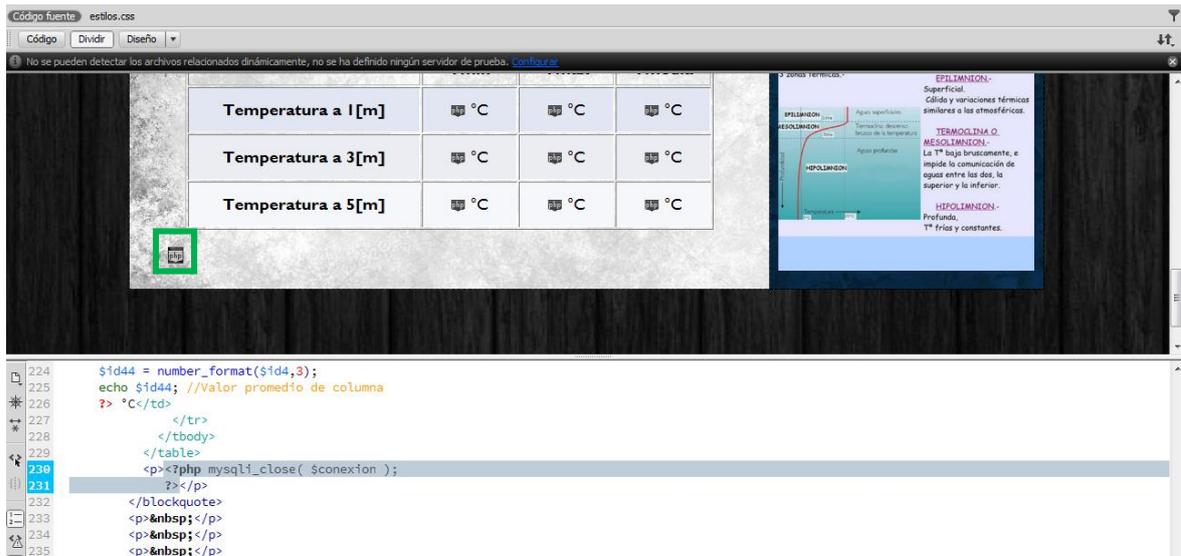
Segmento de código 2.11. Sacando el valor máximo de una columna y mostrándolo en la página web.



Segmento de código 2.12. Sacando el valor promedio de una columna y mostrándolo en la página web.

Así como se estableció la conexión al inicio de la página, lo correcto es cerrarla al terminar de trabajar con la base de datos. Para este proceso se usa el Segmento de código 2.13.

Finalmente cuando se tenga valores en la base de datos se puede observar la página web como se muestra en la Figura 2.53. Así será como el usuario final podrá observar los valores de temperatura con su respectiva estampa de fecha y hora en la página web además de observar la tabla con los valores mínimos, máximos y promedios.



Segmento de código 2.13. Código para cerrar la conexión con la base de datos.

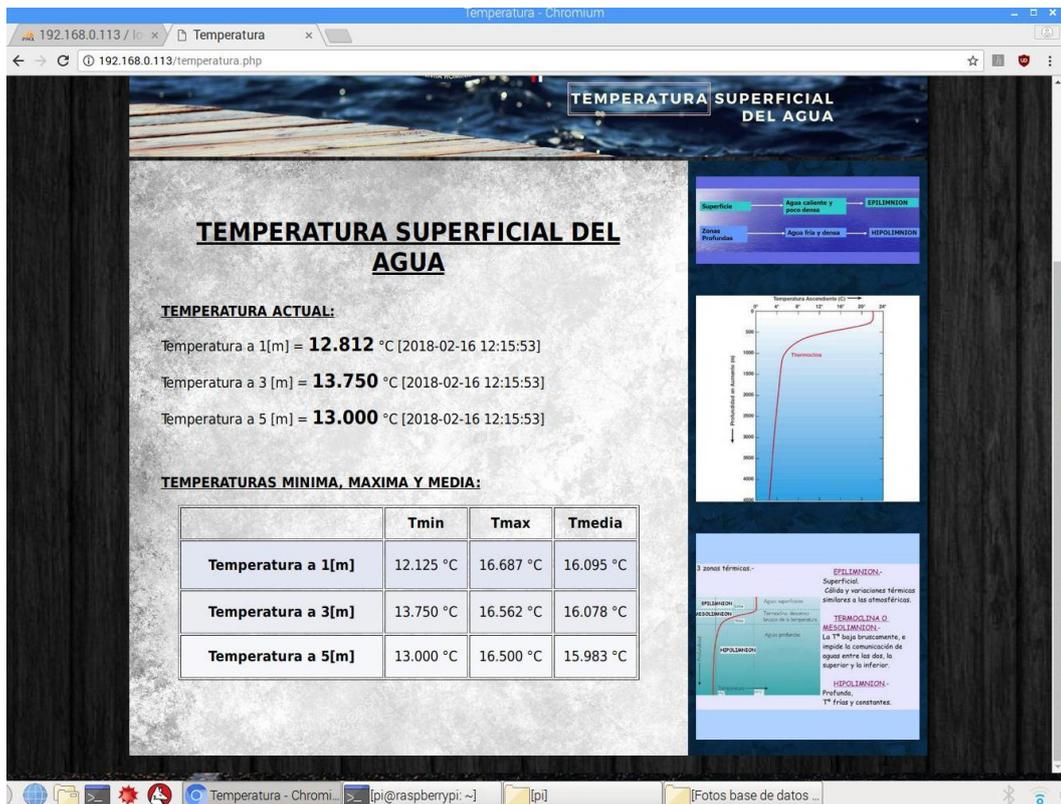


Figura 2.53. Página temperatura.php

Para observar todo el código desarrollado en php para la página “temperatura.php” revisar el Anexo II.

2.5.6.4 Desarrollo página web tendencia.php

La tercera página web mostrará las tendencias de temperatura de las tres profundidades donde se encuentran los sensores. Estas gráficas podrán ser descargadas en varios formatos para que el usuario pueda analizarlas en un futuro.

Al igual que la página web anterior “temperatura.php”, “tendencia.php” ya se encuentra creada y lista para ser modificada en el espacio de “contenido”. También se usará código PHP, ya que las gráficas que se mostrarán son de contenido dinámico, se irán actualizando cada vez que los sensores tomen datos.

Para iniciar, se escriben los títulos con el tipo y tamaño de letra que se desea; y se dejan los espacios donde se mostrarán las figuras, como se muestra en los recuadros de la Figura 2.54.



Figura 2.54. Diseño de la página web “tendencia.php”.

Primero, se creará la conexión con MySQL donde se encuentra la base de datos. Sin una conexión establecida no se podrá mostrar valores de la bases. Para establecer la conexión se debe configurar ciertos valores como usuario, *password*, servidor y a que base de datos se va a conectar como se observa en la Segmento de código 2.14.

Para mostrar una gráfica de forma dinámica con los datos de la base, se usa código PHP y la ayuda de la herramienta *HighCharts*.

Lo primero que se debe hacer es crear un fichero llamado *RandomClass.php*; en este fichero estarán dos funciones que se usarán para graficar las tendencias. Las funciones que se tendrá en el fichero son “function *getArraySQL(\$sql)*” y “function *getAllInfo()*”.

```

54 <p ><h1><strong><u><center>TENDENCIAS DE LA TEMPERATURA DEL AGUA</center></u></strong></h1></p>
55 <p class="titulo2"><?php
56     require_once("RandomClass.php");
57
58     $usuario = "sensores";
59     $password = "c2875";
60     $servidor = "localhost";
61     $basededatos = "monitoreo";
62     //variable que guarda la conexión de la base de datos
63     $conexion = mysqli_connect( $servidor, $usuario, $password, $basededatos ) or die ("No se ha podido conectar al servidor de Base de datos"
64 );
65     // Selección del a base de datos a utilizar
66     $db = mysqli_select_db( $conexion, $basededatos ) or die ( "Ups! Pues va a ser que no se ha podido conectar a la base de datos" );
67     //Comprobamos si la conexión ha tenido éxito
68     if(!$conexion){
69         echo 'Ha sucedido un error inesperado en la conexión de la base de datos<br>';
70     } ?></p>
71 </blockquote>

```

Segmento de código 2.14. Conexión con la base de datos.

Se puede observar el contenido de la función “getArraySQL(\$sql)” que se encuentra en el fichero “RandomClass.php” en el Segmento de código 2.15.

```

37 //Devuelve un array multidimensional con el resultado de la consulta
38 function getArraySQL($sql){
39     //Creamos la conexión
40     $conexion = $this->conectarBD();
41     //generamos la consulta
42     if(!$result = mysqli_query($conexion, $sql)) die();
43
44     $rawdata = array();
45     //guardamos en un array multidimensional todos los datos de la consulta
46     $i=0;
47     while($row = mysqli_fetch_array($result))
48     {
49         //guardamos en rawdata todos los vectores/filas que nos devuelve la consulta
50         $rawdata[$i] = $row;
51         $i++;
52     }
53     //Cerramos la base de datos
54     $this->desconectarBD($conexion);
55     //devolvemos rawdata
56     return $rawdata;
57 }
58
59

```

Segmento de código 2.15. Función getArraySQL(\$sql).

Además en el mismo fichero “RandomClass.php” se encuentra la función “getAllInfo()” con el código mostrado en el Segmento de código 2.16.

```

85
86 function getAllInfo(){
87     //Creamos la consulta
88     $sql = "SELECT * FROM random;";
89     //obtenemos el array con toda la información
90     return $this->getArraySQL($sql);
91 }
92 }
93
94 ?>

```

Segmento de código 2.16. Función getAllInfo ().

Una vez que ya se tiene el fichero con las funciones mencionadas, se crea el código en la página web “tendencia.php”. Aquí se hace la lectura de todos los datos de la base (con la función getAllInfo ()) y se dibuja la gráfica con Highcharts.

En primer lugar se incluye el fichero “RandomClass.php” en la página web “tendencia.php” como se observa el Segmento de código 2.17.

```
52 <strong>TENDENCIAS DE LA TEMPERATURA DEL AGUA</strong></p>
53 <strong><?php
54 require_once("RandomClass.php");
55
```

Segmento de código 2.17. Incluir el fichero “RandomClass.php”.

Después, se obtiene todos los datos de la tabla “temperatura” de la base de datos en un arreglo; dentro de la variable “rowdata[]” se almacenarán todos los datos de la base como se muestra en el Segmento de código 2.18.

```
71 <?php
72
73 $sql = "SELECT * FROM temperatura";
74 if(!$result = mysqli_query($conexion, $sql)) die("¡mal echa consulta!");
75
76 $rawdata = array();
77 //guardamos en un array multidimensional todos los datos de la consulta
78 $i=0;
79 while($row = mysqli_fetch_array($result))
80 {
81 //guardamos en rawdata todos los vectores/filas que nos devuelve la consulta
82 $rawdata[$i] = $row;
83 $i++;
84 }
```

Segmento de código 2.18. Se guarda en la variable “rawdata[]” todos los datos de la base.

A continuación, en la variable “rawdata[]” se tiene los datos de temperatura y la estampa de fecha y hora; para realizar la gráfica en uno de los ejes ira el tiempo (fecha y hora), es por esto que la estampa de fecha y hora de la variable hay que convertirla en un formato adecuado para pasarle a la herramienta Highcharts.

Para esto se usa la función “getTimestamp()”; en un bucle “for”, así se irá separando los valores de temperatura en la variable \$valoresArray y los valores de “fecha” en la variable \$timeArray, como se muestra en el Segmento de código 2.19.

```
85
86
87 //nos creamos dos arrays para almacenar el tiempo y el valor numérico
88 $valoresArray;
89 $timeArray;
90 //en un bucle for obtenemos en cada iteración el valor numérico y
91 //el TIMESTAMP del tiempo y lo almacenamos en los arrays
92 for($i = 0 ; $i < count($rawdata); $i++){
93 $valoresArray[$i] = $rawdata[$i][1]; //temperatura 1
94 //OBTENEMOS EL TIMESTAMP
95 $time = $rawdata[$i][0]; //tiempo
96 $date = new DateTime($time);
97 //ALMACENAMOS EL TIMESTAMP EN EL ARRAY
98 $timeArray[$i] = $date ->getTimestamp()*1000;
99 }
100
```

Segmento de código 2.19. Separación de los valores de temperatura y fecha en dos arreglos diferentes.

Una vez obtenido estos valores se dibujará la gráfica de tendencia con la librería Highcharts; primero se debe incluir las librerías en el archivo. Además se crea una división lógica con un id para identificarlo como se muestra en el Segmento de código 2.20.

```
103 <div id="contenedor1"></div>
104 <script src="https://code.jquery.com/jquery.js"></script>
105 <!-- Importo el archivo Javascript de Highcharts directamente desde su servidor -->
106 <script src="http://code.highcharts.com/stock/highstock.js"></script>
107 <script src="http://code.highcharts.com/modules/exporting.js"></script>
108 </script>
109
```

Segmento de código 2.20. División lógica y carga de scripts.

Finalmente se añade una función en JavaScript usando las funciones de la librería *Highcharts*, los datos que se inserta en la gráfica son los datos que se obtuvo de la base de datos en los pasos anteriores (“\$valoresArray” y “\$timeArray”); como se muestra en el Segmento de código 2.21.

Se puede observar que en la etiqueta series se tiene un campo llamado data, dentro de este campo se crea una función que se llame al inicio del script, esta función añadirá los parámetros a la gráfica, de tal forma que se añadirá tantos parámetros como datos tenga la tabla “temperatura” de la base de datos.

```
108 <script>
109
110 chartCPU = new Highcharts.StockChart({
111   chart: {
112     renderTo: 'contenedor1'
113     //defaultSeriesType: 'spline'
114   },
115   rangeSelector: {
116     enabled: false
117   },
118   title: {
119     text: 'TEMPERATURA DEL AGUA A 1[m]'
120   },
121   xAxis: {
122     type: 'datetime'
123     //tickPixelInterval: 150
124     //maxZoom: 20 * 1000
125   },
126   yAxis: {
127     minPadding: 0.2,
128     maxPadding: 0.2,
129     title: {
130       text: 'Temperatura',
131       margin: 10
132     }
133   }
134 },
135 series: [{
136   name: 'valor',
137   data: (function() {
138     // generate an array of random data
139     var data = [];
140     <?php
141     for($i = 0 ; $i < count($rawdata); $i++){
142     ?>
143     data.push([<?php echo $timeArray[$i];?>,<?php echo $valoresArray[$i];?>]);
144     <?php } ?>
145     return data;
146   })()
147   //return data;
148 }],
149 credits: {
150   enabled: false
151 }
152 });
153
154 </script></p>
```

Segmento de código 2.21. Gráfica de la tendencia a 1 metro gracias al script.

Cabe mencionar que se puede hacer una gran variedad de gráficas con la ayuda de Highcharts, para este proyectos se seleccionó una de las tantas variedades posibles para mostrar la gráfica de tendencias. A continuación se muestra la gráfica de tendencias generada por el script (Figura 2.55).



Figura 2.55. Gráfica de tendencia generada por el script realizado.

Así como se estableció la conexión con la base de datos, lo correcto es cerrar la misma al terminar de trabajar. Para este proceso se usa el Segmento de código 2.22.

La imagen muestra un editor de código con un archivo de código fuente que contiene un script PHP. El código visible es el siguiente:

```
324     });  
325     credits: {  
326         enabled: false  
327     }  
328 });  
329  
330     </script>  
331     </p>?php  
332     $close = mysqli_close($conexion);  
333     ?>  
334     </blockquote>
```

Segmento de código 2.22. Cierre de conexión con la base de datos.

Finalmente, el usuario podrá observar las tendencias de temperatura de los tres sensores como se muestra en la Figura 2.56

Para las otras dos gráficas el código es muy similar al mostrado, para observar todos los comandos para la creación de las gráficas de tendencia con la ayuda de *HighCharts*, dirigirse al Anexo III.



Figura 2.56. Página web “tendencia.php”.

2.5.7 Sistema fotovoltaico

El sistema fotovoltaico estará compuesto por los paneles solares y la batería; no se usará un controlador de carga ya que los paneles tienen integrado un sistema para cargar directamente los equipos, se asemeja a un controlador de carga. Como se menciona en secciones anteriores los paneles tienen puertos USB que dan un voltaje constante de 5V y máximo 2A, la corriente cambiará de acuerdo a la cantidad de sol que reciban los paneles. Además, los paneles tienen un puerto de salida de 18V constantes y la corriente variará de acuerdo a la radiación de sol recibida.

Surge un problema, ya que los paneles solares y la batería que se adquirió no tienen el mismo voltaje de salida y entrada respectivamente, como se muestra en el diagrama de la Figura 2.57.



Figura 2.57. Voltaje de salida y entrada de los paneles solares y la batería.

Para solucionar este problema, se usa un regulador de voltaje, que de 18V, dé una salida de 15V (regulador 1815). Con la corriente no hay problema, ya que a la salida de los paneles se tendrá una corriente máxima de 2A y el regulador 1815 permite un paso de corriente de máximo 3A, por tanto no habrá problema, como se ilustra en la Figura 2.58.



Figura 2.58. Diagrama de los paneles solares y la batería con el regulador 1815.

Se tiene otro problema igual al mencionado, si bien en la batería se tiene puertos USBs de salida que dan 5V y así se podría alimentar al hub USB que su entrada es de 5V, surge el problema con la corriente ya que estos puertos USB tienen una salida máxima de 2 A y para el proyecto la corriente pico sobrepasara este valor, así que por esta razón se usará la otra salida de voltaje de la batería que es de 12V y una corriente máxima de 15A. Con esto se asegura que se tiene la cantidad necesaria de corriente en todos los instrumentos en cualquier momento. Al usar esta salida de 12V de la batería se debe usar un regulador de voltaje de 12V a 5V, como se muestra en la Figura 2.59.



Figura 2.59. Diagrama de la batería y el HUB USB conectados por un regulador 1205.

Una vez solucionado los inconvenientes con los dos reguladores de voltaje, ya se puede montar el sistema fotovoltaico. Si bien la solución con reguladores funciona bien, hay que tener claro que no es algo óptimo en el tema energético.

Como se va usar dos reguladores de voltaje, y estos tienden a calentarse, por precaución se colocó un ventilador que disipe el calor de los dos circuitos. Finalmente se tendría un esquema semejante al mostrado en la Figura 2.60.

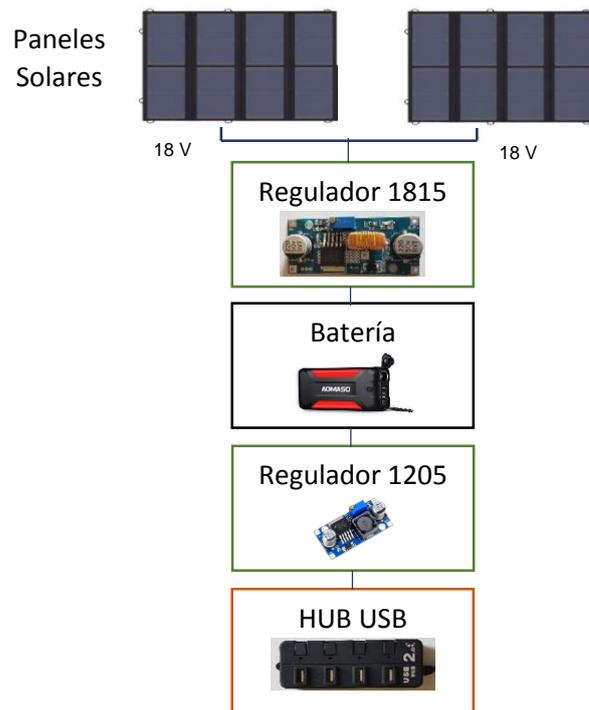


Figura 2.60. Diagrama del sistema Fotovoltaico.

2.5.8 Circuitos adicionales

Para facilitar la conexión de los elementos como los sensores y los paneles solares se usará circuitos para facilitar la conexión. Además se crea otro circuito para encender y apagar la Raspberry, que en secciones anteriores ya se creó el script para que este funcione.

2.5.8.1 Circuito para conectar los paneles solares

Para conectar los paneles solares en paralelo se crea un circuito que facilite la conexión con borneras para la fácil instalación de los cables. Además de un botón para apagar y prender los paneles, como se muestra en la Figura 2.61.



Figura 2.61. Circuito para la conexión de los paneles solares.

2.5.8.2 Circuito para conectar los sensores

También, para facilitar la conexión de los tres sensores con la Raspberry Pi se implementó una baquelita con borneras, además de conectar los sensores con la resistencia de *pull up*, la cual se explicó en la sección de sensores (Figura 2.62).

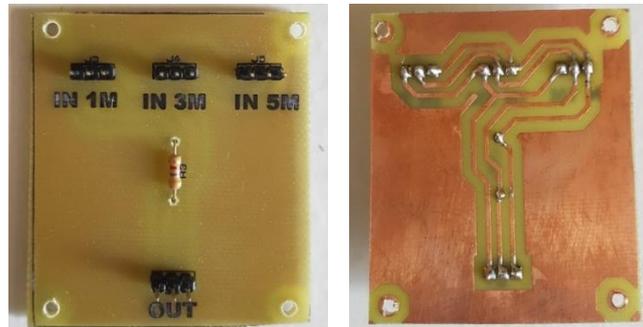


Figura 2.62. Circuito para conectar los sensores a la placa.

2.5.8.3 Circuito para encender/apagar la Raspberry Pi

En una sección anterior, se creó el script para encender y apagar la Raspberry a través de un botón y un indicador led, se implementa una baquelita con este botón y el led conectados a su respectiva resistencia de protección, además de borneras para facilitar la conexión de la baquelita con la Raspberry Pi, como se ilustra en la Figura 2.63.



Figura 2.63. Circuito para encender o apagar la Raspberry

2.5.9 Conexión de los equipos

Una vez que ya se tienen todos los equipos, y estos ya están configurados, se deben conectar sobre una superficie para que estos vayan dentro de la caja de comunicación. Se selecciona una tabla liviana para instalar y asegurar todos los equipos; como se muestra en la Figura 2.64.



Figura 2.64. Tabla sobre la cual se instalaran y conectaran todos los equipos.

Se busca la manera para que los equipos estén bien asegurados y dispuestos de una forma que facilite la conexión de todos ellos. (Figura 2.65 y Figura 2.66).



Figura 2.65. Instalación de los elementos en el tablero.

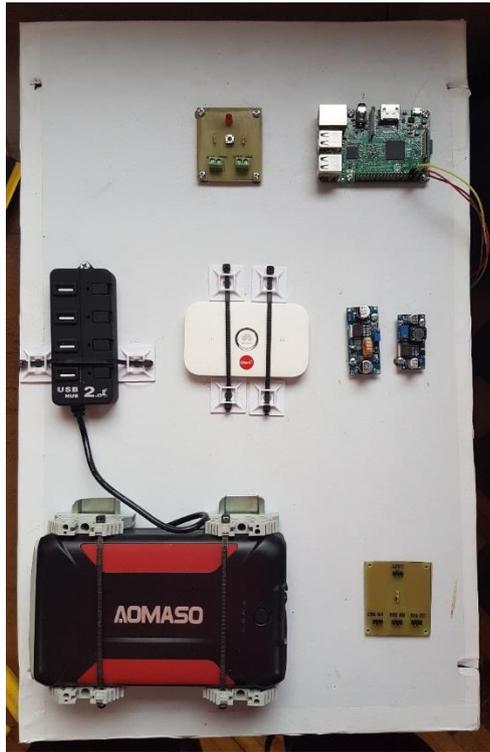


Figura 2.66. Instalación y sujeción de los equipos.

Una vez instalados los equipos, se procede a la conexión con cables de diferentes colores para tener una mejor referencia de donde están conectados, como se ilustra en la Figura 2.67.



Figura 2.67. Conexión de los equipos en el tablero.

2.5.10 Mecanismos para que el prototipo sea resistente al agua

Uno de los problemas es que el prototipo sea resistente al ambiente en el que va ser usado, es decir resistente al agua, para así poder preservar los componentes electrónicos y alargar la vida útil de los mismos. Hace falta pequeñas cantidades de agentes agresivos o de productos de corrosión para causar degradación de los componentes electrónicos, e incluso el fallo total del sistema. Cabe mencionar que el prototipo al estar una parte sobre la laguna y otra parte debajo de la misma, hace que se deba buscar mecanismos para aislar el armario de circuitos del agua.

Para proteger todos los equipos se los pondrá dentro de un armario eléctrico, para así protegerlos del ambiente externo, solo los paneles solares y los sensores de temperatura que son resistentes al agua, irán fuera de la caja de comunicaciones.

Además, hay que tener en cuenta que las conexiones con la caja estén protegidas, para así proteger el ingreso de agua al armario eléctrico, y el daño de los equipos.

Para facilitar la instalación de los sensores y para darles una protección extra a golpes y abolladuras, se instala dentro de una manguera, en donde irán los tres sensores (Figura 2.68.), así a la hora de instalarlos en el agua solo se bajará una manguera y se la estirará a la profundidad correcta, con una pesa colocada en la manguera.



Figura 2.68. Instalando los sensores en la manguera.

La caja de circuitos es el elemento principal para la protección contra el agua de los elementos a usar. Se ha escogido una caja resistente al agua y al polvo para proteger todos los equipos, si bien la caja no estará en contacto directo o sumergida al agua, tendrá ciertas salpicaduras, además de estar expuesta a las inclemencias del tiempo como la lluvia. La caja que se usará tendrá unas medidas de 20 cm de largo por 40 cm de ancho por 60 cm de alto como se ilustra en la Figura 2.69.

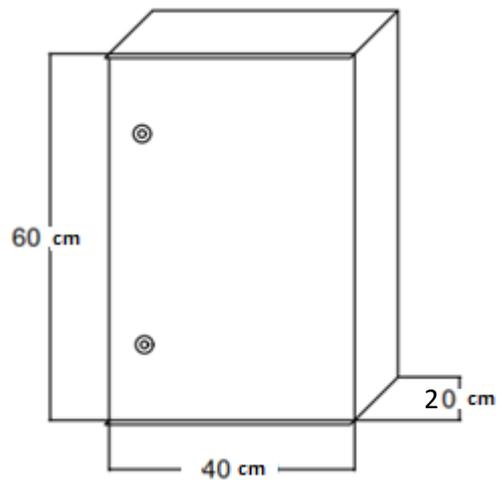


Figura 2.69. Medidas de la caja de protección.

En la Figura 2.70 se muestra las fotos de la caja de comunicación usada.



Figura 2.70. Fotos de la caja de comunicación.

Como se mencionó la caja tendrá protecciones contra el ingreso de agua y polvo. En la parte de la puerta se encuentra una junta tórica que impide el ingreso de agua cuando la caja se encuentre cerrada como se ilustra en la Figura 2.71. Además cuenta con unas pequeñas protecciones en la parte superior e inferior de la puerta para impedir el ingreso de agua.

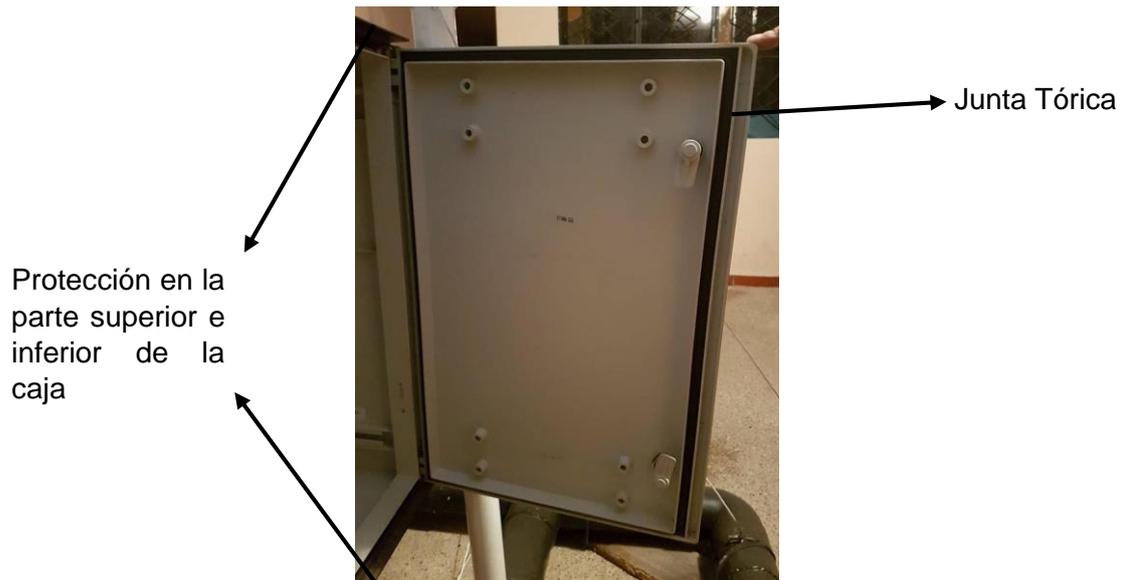


Figura 2.71. Protecciones de la caja de comunicación.

Además, para las conexiones exteriores de los cables de los paneles solares y los tres sensores, se usan conectores de aislamiento con rosca PG para impedir el ingreso de agua y tener conexiones con el exterior, como se muestra en las Figura 2.72 y en la Figura 2.73.



Figura 2.72. Conectores rosca PG.

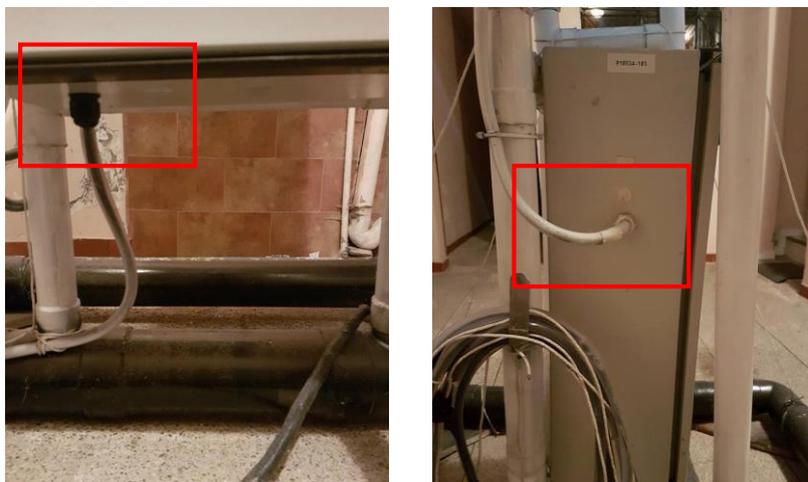


Figura 2.73. Conexiones del exterior a la caja.

Los paneles solares son resistentes al agua, están diseñados para usarse en este tipo de ambiente, como se mencionó en secciones anteriores. El material y los elementos con lo que los paneles están fabricados los hace perfectos para este proyecto, por el ambiente en el que se los va a usar.

Además, la batería es resistente al agua, si bien irá dentro de la caja de comunicaciones, no está demás tener una protección extra en caso de emergencia.

2.5.11 Estructura de soporte para el prototipo

Para mantener la caja de comunicaciones y los paneles solares sobre el agua, se crea una estructura que pueda sostener estos elementos de una manera adecuada y firme sobre la superficie del agua. Por tanto esta infraestructura deberá flotar y ser lo suficientemente robusta para sostener el prototipo.

El prototipo irá sobre una base cuadrada de 1.33 metros de largo y ancho, con refuerzos en la mitad, elaborada sobre tubo PVC de 4 pulgadas. Se escogió este material ya que permite construir la estructura de acuerdo al diseño, con implementos como codos y uniones. Además permite mantener a flote todo el prototipo y no es pesado. En la Figura 2.74 se muestran las medidas de la base, en la Figura 2.75 la implementación.

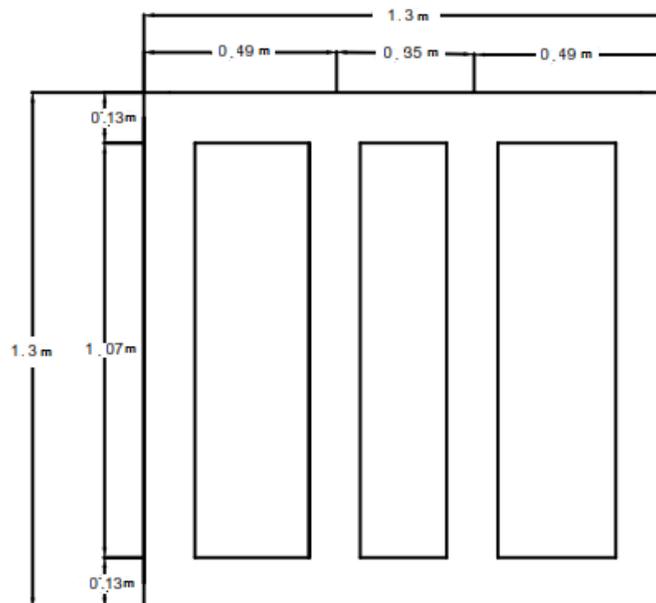


Figura 2.74. Base de la estructura con sus medidas, diseñada en AutoCAD.



Figura 2.75. Implementación de la base de la estructura.

El resto de la estructura se hará con tubo PVC de 2 pulgadas, más liviano para sostener la caja de comunicaciones y los paneles solares sobre la base. En la Figura 2.76 se muestra el diseño previo con las medidas deseadas.

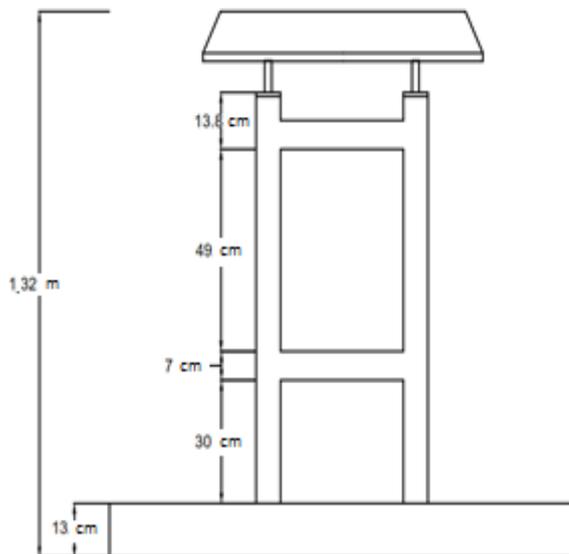


Figura 2.76. Diseño de Infraestructura.

Una vez hecha la estructura, se coloca la caja de comunicaciones con pernos tipo espárrago para sostenerla en él tubo PVC de 2 pulgadas, como se observa en la Figura 2.77.



Figura 2.77. Instalación de la caja de comunicaciones en la infraestructura.

El soporte superior, donde irán los paneles solares, se lo realizó en aluminio, ya que es un material liviano. Sobre este soporte se puso un techo de material muy liviano para soportar los paneles; este techo se instaló con una inclinación para que los paneles puedan recibir la mayor parte de radiación del sol, como se muestra en la Figura 2.78.



Figura 2.78. Infraestructura ya con el techo para los paneles solares.

Finalmente el prototipo está listo para ser probado y tomar la temperatura superficial del agua.

3. RESULTADOS Y DISCUSIÓN

Para realizar las pruebas de funcionamiento del prototipo se seleccionó la laguna de Colta, ya que presta todas las características óptimas para probar el prototipo. Además, que se brindó todas las facilidades por parte de la empresa COLMITUR S.A., encargada de la administración de la Laguna de Colta; para poder realizar las pruebas por el periodo de 5 días.

3.1 Laguna de Colta

La laguna es conocida en lengua Puruhá como “Kulta Kucha”, que significa “Laguna de Pato”. La laguna de Colta es única, desde Loja hasta Quito no existe otra de fácil acceso. Es un gran espejo de agua de forma alargada, rodeada de gran vegetación. Se la puede observar desde la carretera, adornada por el Chimborazo y por su naturaleza, con paisajes andinos únicos, como se ilustra en la Figura 3.1.



Figura 3.1. Laguna de Colta y el Chimborazo

La laguna de Colta permite realizar actividades como: caminatas, fotografía, observación de flora y fauna, observación del paisaje y aventura.

La laguna de Colta, se encuentra a 20 km. de la ciudad de Riobamba y a 3 km. de la cabecera cantonal Cajabamba (Villa La Unión), a una altitud de 3.300 m.s.n.m. con una variación de temperatura promedio de 12 a 15 grados centígrados; tiene 2800 m. de largo por 1000 m. de ancho, está rodeado de varias colinas, donde habitan varias comunidades indígenas y dueños de una cultura milenaria.

En esta zona existe abundante flora destacándose la totora que es utilizado para alimento de animales domésticos, así como también para elaborar los tradicionales “caballitos de totora” que sirven para navegar; en la fauna se puede observar peces de colores y los más importantes son los patos de patas largas, únicos en su especie y que son originarios de

este lugar, por este motivo a la laguna le bautizaron como la de patos o Colta. En la Tabla 3.1 se muestra el resumen de las características de la laguna de Colta.

Tabla 3.1. Características de la laguna de Colta.

Ubicación	La laguna de Colta se halla ubicada a 20 minutos al suroeste de la ciudad de Riobamba. Coordenadas: 1°44'17.3"S 78°45'19.0"W
Extensión	Largo: 2800 metros. Ancho: 1000 metros.
Clima	Clima templado, la temperatura se encuentra entre los 12 y 15 grados centígrados.
Flora	Crece la totora, utilizada para el tejido de esteras y variadas artesanías.
Fauna	Entre los principales están: patos emigrantes, garzas, ovejas, gran diversidad de aves, y peces de colores.

Para tener una mejor referencia de la forma y la ubicación de la laguna de Colta se muestra en la Figura 3.2, una captura de la laguna a través de Google Maps.

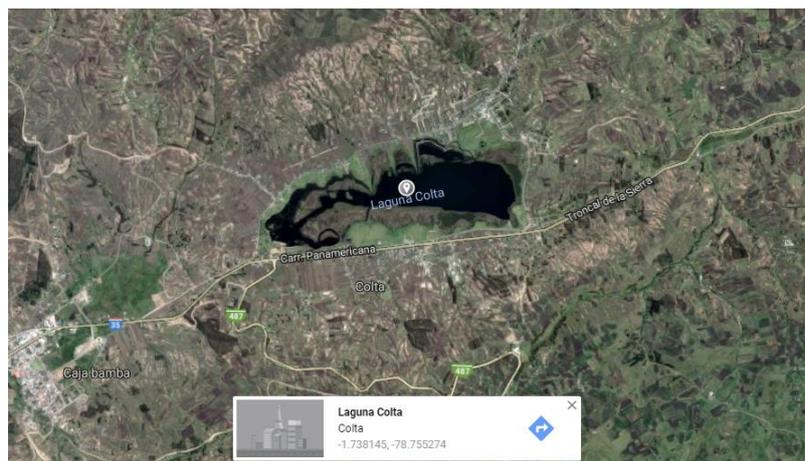


Figura 3.2. Laguna de Colta en Google Maps.

3.2 Pruebas de Funcionamiento

Se realizarán dos pruebas de funcionamiento en diferentes lugares. La primera se realizará en un tanque, en donde se podrá verificar que todos los equipos funcionen correctamente,

que la estructura diseñada sea resistente manteniendo la caja de comunicaciones y los paneles solares sobre el agua; finalmente se verificará si hay que hacer algún cambio o mejora al prototipo. La otra prueba se realizará en la laguna de Colta, que como ya se mencionó presenta un ambiente deseado para probar el prototipo ya con sus correcciones y mejoras, si hubiera el caso de hacerlas después de realizar las pruebas en el tanque.

3.2.1 Pruebas iniciales en un tanque

Una vez que el prototipo esté listo y configurado, se realizaron las pruebas en un tanque de agua de dimensiones 4m x 3m x 3m como se ilustra en la Figura 3.3.

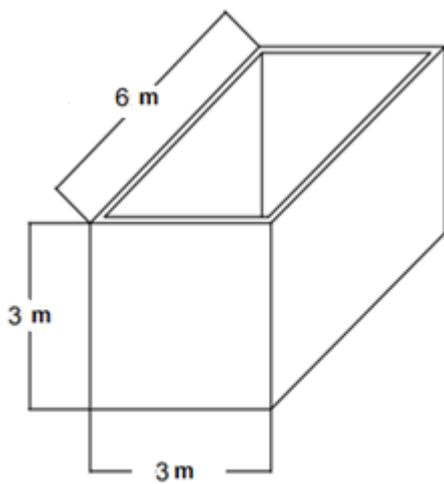


Figura 3.3. Tanque de agua donde se realizaron las pruebas iniciales.

Si bien la profundidad del tanque es 3 metros y el sensor de mayor distancia tiene 5 metros, la manguera se estirará a lo largo del tanque para poder tomar datos de los tres sensores y probar el funcionamiento de todos los equipos y sobre todo de la estructura.

Se hicieron las pruebas en los equipos para tomar la temperatura a las tres diferentes profundidades y poder visualizarlas en la página web. Además, se probó que la estructura mantenga a flote la caja de comunicaciones y los paneles solares; todo esto siendo resistente al agua, es decir, prohibido el ingreso de agua a la caja de comunicaciones.

Una vez en la ubicación del tanque de agua, se encendió el prototipo para ingresarlo al tanque, como me ilustra en la Figura 3.4.



Figura 3.4. Prototipo listo para ser probado en el tanque de agua.

Una vez preparado el prototipo, y cerrada la caja de comunicaciones, se lo puso sobre el agua con la ayuda de cuerdas. Lo primero que se hizo, fue verificar que la estructura se mantenga sobre la superficie del agua, es decir, que flote con todos los equipos y no se hunda, o se filtre agua. Al realizar las pruebas y verificar visualmente la estructura, esta no tuvo ningún problema y se mantuvo flotando con todos los equipos, como se muestra en la Figura 3.5.



Figura 3.5. Prototipo probado en el tanque de agua.

Una vez verificada toda la estructura y funcione de acuerdo a lo esperado, se acabó de llenar el tanque con agua (Figura 3.6). Así se quedó el prototipo en el tanque por el lapso de tres días, para verificar cualquier inconveniente durante este tiempo.



Figura 3.6. Prototipo funcionando sin problemas en el tanque de agua.

Una vez probada la estructura, lo siguiente fue probar la adquisición de datos de temperatura por los tres sensores, y que estos datos puedan ser observados en la página web diseñada; así que a través del celular se verificó que los datos se muestren en la página sin problemas, como se muestra en la Figura 3.7.



Figura 3.7. Página web mostrando los datos de temperatura superficial del agua del tanque.

También se verificó que la página web muestre las tendencias de temperatura; esto se lo hizo a través del celular, observando que las gráficas correspondan a los datos tomados, como se muestra en la Figura 3.8.

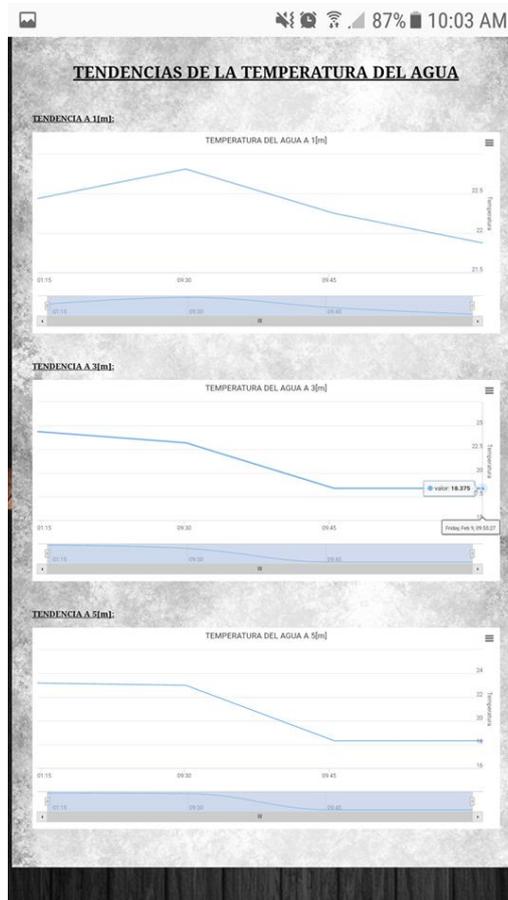


Figura 3.8. Tendencias de la temperatura del agua en el tanque.

En el sector donde se realizaron las pruebas llovió al segundo día de estar instalado el prototipo, esto permitió probar que todo el equipo sea resistente al agua y no haya filtraciones dentro de la caja de comunicaciones (Figura 3.9.). También se verificó que los paneles solares al ser expuestos al agua, funcionen correctamente sin ningún daño (Figura 3.10.).

Una vez probado el prototipo por tres días y verificando constantemente el funcionamiento, se comprobó que todo funciona correctamente, incluso el sistema fotovoltaico funciono adecuadamente dotando de energía a los equipos todo el tiempo y cargando la batería para más días de uso, aun habiendo un día completo de lluvia con escasa radiación solar.

Después de estas pruebas, se decidió pintar la base para que sea más firme y dar protección extra, ya que por el traslado del prototipo, y dado que la base está en continuo

contacto con el suelo, se requería protección extra. Así que a la base se aplicó pintura negra resistente al agua.



Figura 3.9. Caja de comunicaciones sin filtraciones de agua después de la lluvia.



Figura 3.10. Paneles solares sin ningún daño después de la lluvia.

3.2.2 Pruebas finales en la Laguna de Colta

Las pruebas de funcionamiento se realizaron en la laguna de Colta, las coordenadas tomadas con el GPS y mostradas en Google Maps se muestran en la Figura 3.11.

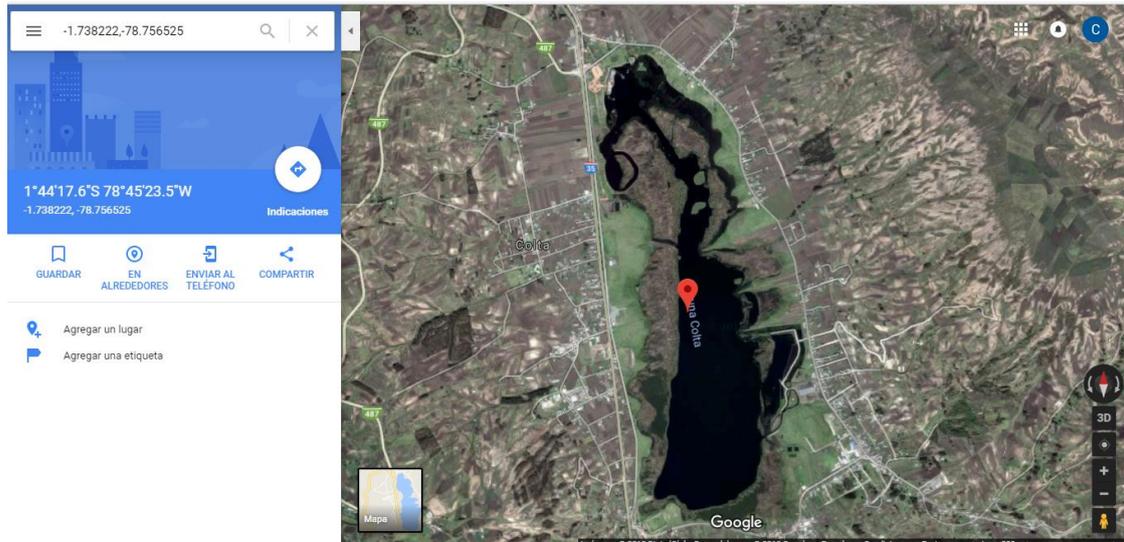


Figura 3.11. Ubicación en Google Maps donde se instaló el prototipo.

Las coordenadas donde se instaló el prototipo fueron:

1°44'17.6\"S 78°45'23.5\"W

El punto especificado se encuentra aproximadamente en la mitad de la laguna, donde está la mayor profundidad de 7 metros. Es importante instalar el prototipo en una profundidad mayor a 5 metros ya que el sensor más profundo tiene esta longitud.

Para trasladarse al punto se usó una embarcación provista por COLMITUR-S.A. empresa responsable del manejo de la laguna de Colta, quienes ayudaron con todas las facilidades para realizar las pruebas en la laguna.



Figura 3.12. Prototipo listo en la embarcación para ser trasladado al lugar de instalación.

Una vez en el punto establecido se instaló el equipo y se bajó la manguera de los sensores como se ilustra en la Figura 3.13. Además, se ancló el prototipo con la ayuda de unas

pesas para que por el viento y el movimiento del agua no se mueva del lugar instalado (ver Figura 3.14.). También se colocó otras pequeñas pesas en la manguera de los sensores (ver Figura 3.15.), para que se extienda en toda su longitud, y poder tomar las medidas de temperatura a las profundidades establecidas.



Figura 3.13. Instalación del prototipo en el lugar seleccionado.



Figura 3.14. Pesa para anclar el prototipo.



Figura 3.15. Pesas para la manguera de sensores.

Una vez instalado el prototipo, con los sensores en la profundidad establecida, se empezó a tomar datos de temperatura superficial del agua en la laguna de Colta, como se muestra en la Figura 3.16.



Figura 3.16. Prototipo funcionando en la laguna de Colta.

El prototipo al ser autónomo energéticamente gracias a los paneles solares y la batería, se quedó tomando los datos de temperatura por el lapso de 5 días ininterrumpidos, sin necesidad de proveer energía eléctrica adicional o hacer algún ajuste (Figura 3.17.).

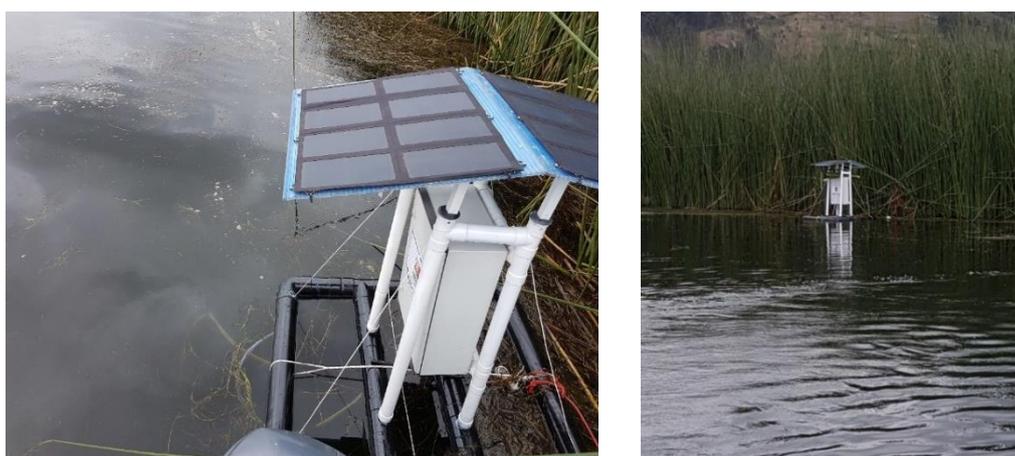


Figura 3.17. Prototipo instalado en el lugar seleccionado.

Se puede observar los videos de las pruebas realizadas en la laguna de Colta en el Anexo V.

3.3 Análisis de resultados.

Con las pruebas realizadas se logró comprobar que todos los equipos funcionen correctamente. Además la estructura no tuvo problemas para mantenerse en el agua con todos los equipos.

El prototipo tomo datos de temperatura del agua cada 10 minutos a tres diferentes profundidades, y envió estos datos de manera autónoma a través de enlace celular hacia la página web.

El sistema fotovoltaico para la alimentación autónoma del prototipo funcionó correctamente. Doto de energía a los equipos durante todas las pruebas, incluso en días nublados y de lluvia logró cargar la batería y que funcione según lo requerido.

Los equipos y mecanismos usados en el proyecto dotaron de protección contra el agua y el ambiente que fue probado. Al finalizar las pruebas se verificó que los equipos seguían protegidos y no tuvieron contacto con el agua, gracias a la caja de comunicaciones y los mecanismos como mangueras y conectores que se usaron para las conexiones externas; cumplieron el objetivo de no permitir el ingreso de agua. Además los paneles solares y los sensores que estuvieron en contacto con el agua funcionaron sin problema alguno, comprobando su resistencia al agua.

Se comprobó que el diseño de la estructura fue óptimo y funcionó de acuerdo a lo esperado, siempre se mantuvo a flote y tuvo la robustez necesaria para sostener los equipos y soportar el ambiente y las inclemencias del tiempo.

De los datos obtenidos, se presenta una tabla en el Anexo IV. En la tabla consta la temperatura de los sensores a 1 metro, a 3 metros y a 5 metros tomados cada 10 minutos.

El período donde se tomó los datos fue desde el miércoles 14 de febrero al domingo 18 de febrero del 2018. En la tabla mostrada en el Anexo IV, se muestra los datos obtenidos por los sensores con la respectiva fecha y hora de cada dato.

3.3.1 Gráficas de tendencias

A continuación, se presenta el gráfico de la tendencia de la temperatura del agua a 1 metro de profundidad en la laguna de Colta, Figura 3.18.

Como se observa, la temperatura a 1 metro fluctúa entre los 16 grados centígrados. Varía de acuerdo a la hora del día y por ende a la cantidad de sol que calienta la superficie del agua.

TEMPERATURA DEL AGUA A 1[m]

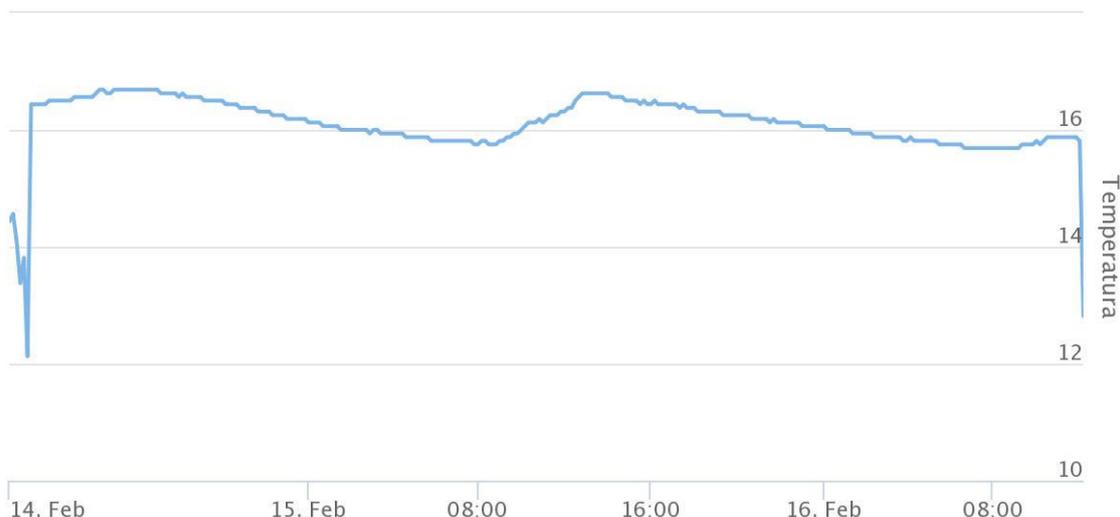


Figura 3.18. Gráfica de tendencia de temperatura a 1 metro.

Los primeros y los últimos datos no se toman en cuenta ya que se tomaron durante la instalación/desinstalación del prototipo en la laguna, así que estos no corresponden a la temperatura real del agua a 1 metro.

Además, la temperatura media calculada a 1 metro fue de 16.095 grados centígrados, y la temperatura máxima registrada fue 16.687 grados centígrados.

En la Figura 3.19 se muestra la tendencia de la temperatura superficial de la laguna de Colta a 3 metros de profundidad.

TEMPERATURA DEL AGUA A 3[m]

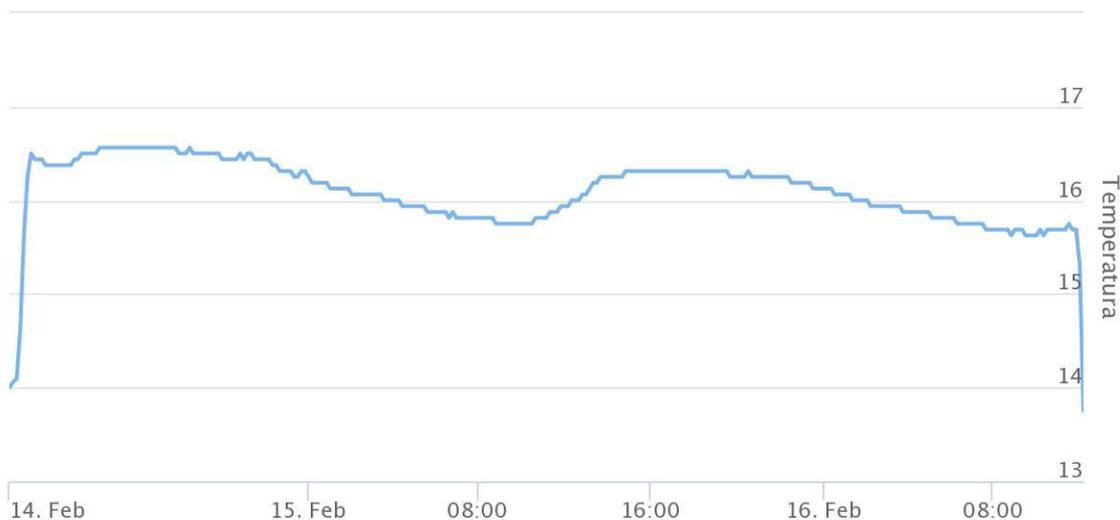


Figura 3.19. Grafica de tendencia de temperatura a 3 metros.

Como se observa la temperatura a 3 metros fluctúa entre los 16 grados centígrados al igual que a 1 metro. Varía de igual manera en el transcurso del día como vaya cambiando la temperatura superficial de acuerdo a la cantidad de sol.

También, los primeros y los últimos datos no se los toma en cuenta ya que se tomaron durante la instalación/desinstalación del prototipo en la laguna, así que estos no corresponden a la temperatura real del agua a 3 metros.

Además, la temperatura media calculada a 3 metros fue de 16.078 grados centígrados y la temperatura máxima registrada fue 16.562 grados centígrados.

Como se espera la temperatura media a 3 metros fue menor a la de un metro; al estar el sensor más profundo en el agua se esperaría tener una menor temperatura. Si bien no es mucha la diferencia de temperatura, la distancia entre los sensores también no lo es (2 metros entre el sensor de 1 metro y 3 metros).

En la Figura 3.20 se muestra la tendencia de la temperatura superficial de la laguna de Colta a 5 metros de profundidad.



Figura 3.20. Gráfica de tendencia de temperatura a 5 metros.

Como se observa la temperatura a 5 metros fluctúa entre los 16 grados centígrados al igual que a 1 y 3 metros. Varía de igual manera en el transcurso del día como vaya cambiando la temperatura superficial de acuerdo a la cantidad de sol que se presente ese día.

Además, la temperatura media calculada a 5 metros fue de 15.983 grados centígrados y la temperatura máxima registrada fue 16.500 grados centígrados.

Como se espera la temperatura media a 5 metros fue menor a la de 1 y 3 metros; al estar el sensor más profundo en el agua se tiene una menor temperatura.

A continuación se muestra la Tabla 3.2 que se puede observar en la página web mostrando los valores mínimos, máximos y medios del transcurso en que se probó el prototipo.

Tabla 3.2. Temperatura mínima, máxima y media.

	Tmin	Tmax	Tmedia
Temperatura a 1[m]	12.125 °C	16.687 °C	16.095 °C
Temperatura a 3[m]	13.750 °C	16.562 °C	16.078 °C
Temperatura a 5[m]	13.000 °C	16.500 °C	15.983 °C

Al examinar la tabla se puede destacar ciertos aspectos, el primero las temperaturas mínimas no son las correctas ya que el prototipo se encendió antes de ingresar en el agua y se apagó un tiempo después de ya estar fuera, así que tomo valores iniciales y finales que no corresponden a la temperatura real del agua, como ya se mencionó en las gráficas de las tendencias.

Los valores de temperatura máxima y media se muestran de igual manera en la tabla, y son como se esperaba, a 1 metro la temperatura media y máxima es mayor que a 3 y 5 metros. Los sensores al estar sumergidos más profundos en el agua tendrán una temperatura menor, ya que la luz y el sol a más profundidad llegan en menor cantidad.

En la Figura 3.21 se muestra la temperatura ambiental de la laguna de Colta del mes de febrero cuando se realizaron las pruebas. Estos datos de temperatura ambiental son sacados de la página web "*The weather channel*" que muestra la temperatura de cualquier sitio a nivel mundial (<https://www.weather.com>).

Como se resalta los días que se realizaron las pruebas la temperatura ambiental está acorde a la temperatura superficial del agua en la laguna de Colta. De aquí se puede deducir que la temperatura superficial de la laguna dependerá en gran medida a la temperatura ambiental de esta.

A continuación, en la Figura 3.22 se muestra un ejemplo de cómo varía la temperatura mínima y máxima a 1 metro de profundidad en un día.

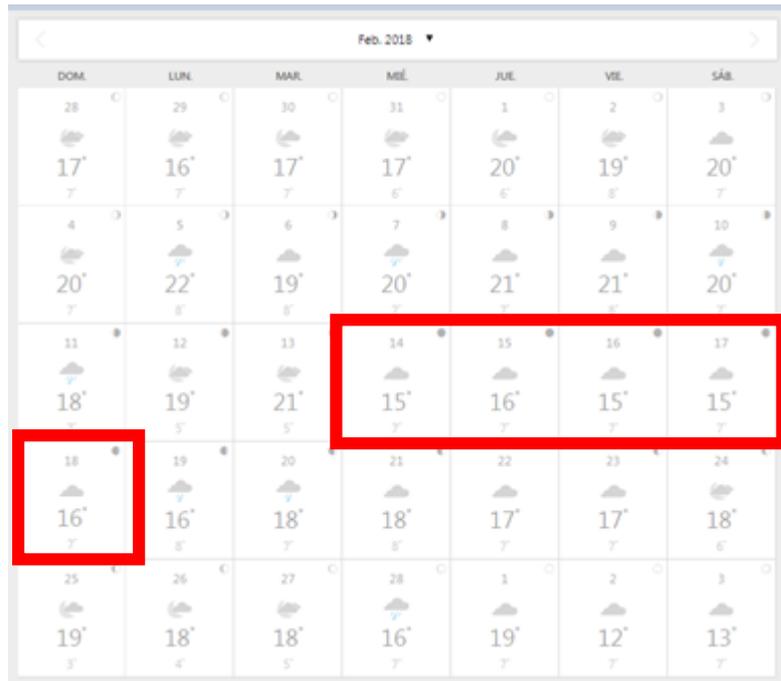


Figura 3.21. Temperatura ambiental de la laguna de Colta en el mes de Febrero.



Figura 3.22. Temperatura mínima mostrada en la tendencia a 1 metro.

La temperatura mínima en la tendencia es de 15.750 grados centígrados, y se la registro el día jueves 15 de febrero a las 08:30 de la mañana. Esta sería la verdadera temperatura mínima registrada y no la que se muestra en la Tabla 3.2.

La temperatura máxima registrada se puede observar en la Figura 3.23, tiene un valor de 16.687 grados centígrados y fue registrada el jueves 15 de febrero a las 13:03 de la tarde. Es lógico pensar que a esta hora hay mayor cantidad de sol y por ende la temperatura superficial será la máxima en este día.



Figura 3.23. Temperatura máxima mostrada en la tendencia a 1 metro.

4. CONCLUSIONES

4.1 Conclusiones

- La primera prueba del prototipo en el tanque, sirvió para ver que los equipos funcionen correctamente y si hay que hacer algún cambio hacerlo antes de la prueba final en la laguna de Colta. Además la prueba en el tanque ayudó a comprobar que el prototipo es totalmente resistente al agua, ya que después de los tres días de pruebas y con lluvia, dentro de la caja no había agua ni humedad.
- La estructura realizada funcionó correctamente durante todas las pruebas, se comprobó que es una estructura robusta y sobretodo capaz de mantener a flote todos los equipos. En ningún momento la estructura mostró fallas o inestabilidad, permitiendo a los equipos funcionar correctamente.
- La ubicación del prototipo dentro de la laguna fue aproximadamente en la mitad de la misma, ya que en este sector es donde está la profundidad máxima de 7 metros. Al ir hacia los extremos de la laguna, la profundidad va disminuyendo y al tener la longitud máxima de 5 metros en los sensores se debió buscar la máxima profundidad.
- El día en que se realizaron las pruebas había bastante viento y lluvia, esto permitió comprobar que el prototipo es resistente al agua, y que gracias a la pesa puesta para anclar el prototipo, este no se movió de su sitio.
- En los días de prueba el cielo estaba bastante nublado, por lo que no hubo mucha radiación que recibieran los paneles solares, así se pudo comprobar que el prototipo fue autónomo energéticamente, energizando a los equipos aún en condiciones no tan favorables de sol.
- La cantidad de luz que penetra en el agua se limita a las capas superficiales de agua, es de decir, en las capas más superficiales habrá mayor cantidad de luz y por ende estas tendrán mayor cantidad calor. Así mismo mientras más profundo menor cantidad de luz incidente y por tanto menor calor, es decir, menor temperatura en capas más profundas. Como se observa en los datos recolectado, en los primeros metros se tiene una temperatura que va disminuyendo mientras más profundo se tomen los datos.
- La temperatura superficial del agua de la laguna de Colta cambia tanto estacionalmente como con la profundidad. Al decir estacionalmente se refiere a que la temperatura del agua depende en gran medida a la temperatura ambiental de la

laguna de Colta. La temperatura ambiental dependerá a la época del año, hay meses más calurosas y otros más fríos.

En cada cambio estacional variará también la radiación solar y la temperatura del aire que calientan la superficie de agua.

- Se puede pensar que la temperatura superficial del agua no influye en aspectos biológicos de plantas y animales, pero contrario a esto, la temperatura es uno de los parámetros más importantes e influyente en un ecosistema de agua. Por ejemplo la cantidad de oxígeno y de saturación en el agua depende en gran medida de su temperatura, cuando la temperatura aumenta la cantidad de oxígeno disuelto disminuye.
- Si la temperatura superficial cambia drásticamente o por un tiempo prolongado esto afectara en gran medida a las especies de plantas y animales que viven en la laguna. Es por eso que es importante tener un monitoreo continuo de la temperatura superficial del agua, para así poder determinar los cambio y tendencias de la temperatura, más ahora que el cambio climático es muy drástico en el planeta.

4.2 Recomendaciones

- Una recomendación importante en el sistema fotovoltaico es respecto a los paneles solares, si bien estos funcionaron de acuerdo a lo requerido en las pruebas realizadas, lo óptimo en eficiencia energética fuera tener un solo panel de 40W y no dos de 21W. En el mercado existen paneles de 40W con las mismas características de resistencia al agua, resistencia al polvo, materiales, etc. de los usados en este proyecto; es más, la misma marca “Giaride” fabricante de los paneles usados, cuenta con paneles de 40W.
- Para mejorar la eficiencia energética en el sistema fotovoltaico, se debería tener elementos que eviten el uso de reguladores de voltaje; es decir, los paneles y la batería deberían ser compatibles en términos de voltaje y corriente para así poder conectarse directamente; ya que al usar reguladores va disminuyendo la eficiencia del sistema.
- Si bien en una laguna instalar un prototipo como el diseñado en este proyecto es de gran ayuda para recolectar información importante acerca del agua, ayudaría tener más prototipos de este tipo instalados, es decir, tomar medidas de

temperatura en diferentes lugares de la laguna para así tener información más precisa de toda la laguna y no solo de un punto.

Si bien en la laguna no cambia mucho los parámetros de temperatura de un lugar a otro, ya que el agua se encuentra estancada, en lugares como ríos o mares los parámetros de temperatura del agua si cambian significativamente de un lugar a otro.

- Si bien el prototipo está diseñado para ríos, lagos y lagunas, fuera de gran importancia instalarlo en las costas del país, ya que con la temperatura superficial del mar se puede hacer estudios de fenómenos climatológicos como el fenómeno del niño.

Para instalar el prototipo en el mar se debería hacer ciertos cambios, como hacerlo resistente a la salinidad y humedad del ambiente, ya que son los parámetro que afectarán en mayor medida al prototipo.

5. REFERENCIAS BIBLIOGRÁFICAS

- [1] Adafruit, «Waterproof DS18B20 Digital Temperature Sensor,» [En línea]. Available: <https://www.adafruit.com/product/381>. [Último acceso: Julio 2017].
- [2] C. Castillo y C. Espinoza, «Modelo de simulacion para la calidad de agua en un río,» *CI51D Contaminación de Recursos Hidricos*, pp. 1-18, 2005.
- [3] D. Sanchez, «Calidad del agua y su control,» Universidad de Castilla la Mancha, Castilla, Madrid, 2016.
- [4] S. Bruceet, «El aumento de las temperaturas está matando la biodiversidad de ríos y lagos,» 2016.
- [5] P. Loné, «Indicadores de calidad del agua,» Waterpeople, 2017.
- [6] C. Gonzales, «Monitoreo de la calidad del agua: La temperatura,» Servicio de Extension Agricola, 2011.
- [7] BVSDE, «Aspectos biologicos de la calidad del agua,» 2016.
- [8] IES, «Tecnologia de la comunicacion,» Departamento de Tecnologia, 2012.
- [9] C. Andalucia, «Emision y recepcion de radio,» *Revista digital para profesionales de la enseñanza*, 2010.
- [10] G. Contreras, «Tecnologías móviles,» 2011.
- [11] Electro Industria, «Uso de la tecnologia GSM en transmision de datos industriales,» 2015.
- [12] D. Rojas, «Telefonia movil 3G: Una tecnologia que avanza para quedarse atras,» Universidad Tecnica Federico Santa Maria, Quilpe, Chile, 2009.
- [13] G. Lopez, «Sistemas de 3G,» Universiad de las Americas Puebla, Mexico, 2012.
- [14] M. Chimbo, «Análisis de la propuesta de evolucion de redes 3G y su convergencia a la tecnología 4G para redes de telefonía móvil,» Universidad politecnica saleciana, Cuenca. Ecuador, 2012.
- [15] E. Palacios, Caracteristicas tecnologias de los sistemas móviles de 3G y 4G Lte: Situacion actual en el Ecuador, Guayaquil, Ecuador: Universidad Catolica de Santiago de Guayaquil, 2016.
- [16] Huawei, «HSPA & HSPA+ Introduction,» Huawei Technologies, 2014.
- [17] R. Pucha, «Diseño de un sistema prototipo para transicion de imagenes de estaciones meteorológicas a traves de la red celular para brindar soporte al personal tecnico de meteorología del INAMHI,» Escuela Politecnica Nacional, 2016.

- [18] C. Marcellini, «Tecnología 4G LTE,» Universidad tecnica Federico Santa María, 2013.
- [19] E. Inga, «La telefonía movil de cuarta generacion 4G y Long Term Evolution,» *Revista Bibliografica Ingenius*, pp. 1-10, 2016.
- [20] K. M., «LTE-Advanced Technology Introduction,» Rohde & Schwarz, 2012.
- [21] NETR, «LTE and LTE advanced,» University of Cyprus, 2014.
- [22] J. Wannstrom, «LTE-Advanced,» 3GPP, 2013.
- [23] A. Coronado, «Redes de Transmision Satelital,» Universidad Mariano Galvez, Guatemala, 2008.
- [24] Sistelmedia S.A., «Introduccion al Internet Satelital,» 2012.
- [25] Via satelital, «Block Up-Converter,» 2012.
- [26] Satelite clase A, «Low Noise Block,» 2012.
- [27] D. Aguirre, Diseño y construccion de un modeo de antena para recibir informacion de satelites meteorológicos de órbita polar en formato HRPT, en 1.7 GHz, Arequipa, Peru: Universidad Nacional de San Agustin, 2017.
- [28] M. Alonso, Sistemas Fotovoltaicos, Madrid, España: Centro de Investigaciones energeticas, medioambientales y tecnologicas, 2015.
- [29] M. Gutierrez, «Manual de instalacion y mantenimiento de sistemas solares fotovoltaicos,» Mexico, 2002.
- [30] W. Montilla, «Conceptos básicos para la creación de una pagina web.,» Clase W, 2010.
- [31] A. Gutierrez, Servidor Web, Coacalco: Tecnológico de estudios superiores de Coacalco, 2014.
- [32] Mozilla, «Tecnología web para desarrolladores: HTML,» 2016. [En línea]. Available: <https://developer.mozilla.org/es/docs/Web/HTML>. [Último acceso: 06 Febrero 2018].
- [33] . R. Alvarez y M. Alvarez, «Introducción a la programación en PHP,» 26 Octubre 2016. [En línea]. Available: <https://desarrolloweb.com/articulos/303.php>. [Último acceso: 06 Febrero 2018].
- [34] E. Gomez y P. Martinez, Bases de Datos 1, Universidad Politecnica Superior, 2008.
- [35] Oracle, MySQL 5.0 Reference Manual, México D.F: Oracle and/or its affiliates, 2014.
- [36] Python, «Python Beginners Guide,» Septiembre 2017. [En línea]. Available: <https://wiki.python.org/moin/BeginnersGuide>. [Último acceso: Enero 2018].
- [37] A. Delgado, «¿Qué es el hardware libre?,» 07 Noviembre 2007. [En línea]. Available: <http://www.consumer.es/web/es/tecnologia/hardware/2007/11/20/171514.php>. [Último acceso: 06 Febrero 2018].

- [38] A. Lazlde, «Hardware libre Recomendaciones para el fomento de la innovación ciudadana,» *Buen Conocer - FLOK Society*, vol. 1. 2.3, p. 3, 2015.
- [39] A. Lee, «CubieBoard6 is released to the overseas users,» Diciembre 2017. [En línea]. Available: <http://cubieboard.org/2017/12/27/cubieboard6-is-released-to-the-overseas-users/>. [Último acceso: Enero 2018].
- [40] Electronilab, «Cubieboard A20,» 2017. [En línea]. Available: <https://electronilab.co/tienda/cubieboard-a20/>. [Último acceso: 06 Febrero 2018].
- [41] Madness_Electronics, «CubieBoard 2 A20,» [En línea]. Available: <http://www.madnesselectronics.com/producto/cubieboard-2-a20/>. [Último acceso: 06 Febrero 2018].
- [42] M. López, «Galileo, interesante tarjeta de desarrollo de Intel,» 25 Febrero 2014. [En línea]. Available: <https://www.unocero.com/noticias/ciencia/galileo-interesante-tarjeta-de-desarrollo-de-intel/>. [Último acceso: 06 Febrero 2018].
- [43] TecBolivia, «Intel Galileo,» [En línea]. Available: <http://tecbolivia.com/index.php/venta-de-componentes-electronicos-11/arduino/intel-galileo-detail>. [Último acceso: 06 Febrero 2018].
- [44] Aprendiendo_Arduino, «Aprendiendo Arduino,» 2016. [En línea]. Available: <https://aprendiendoarduino.wordpress.com/2016/09/25/que-es-arduino/>. [Último acceso: 06 Febrero 2018].
- [45] MCI_electronics, «¿Qué es Arduino?,» 2017. [En línea]. Available: <http://arduino.cl/que-es-arduino/>. [Último acceso: Enero 2018].
- [46] Steren, «Placa con Sistema Arduino Uno,» 2015. [En línea]. Available: <http://www.steren.com.mx/tarjeta-programable-arduino-tipo-uno.html>. [Último acceso: 06 Febrero 2018].
- [47] A. García, «panamahitek.com,» 23 Enero 2013. [En línea]. Available: <http://panamahitek.com/arduino-mega-caracteristicas-capacidades-y-donde-conseguirlo-en-panama/>. [Último acceso: 06 Febrero 2018].
- [48] Culturacion, «Raspberry Pi: Qué es, características y precios,» [En línea]. Available: <http://culturacion.com/raspberry-pi-que-es-caracteristicas-y-precios/>. [Último acceso: 06 Febrero 2018].
- [49] Electronicos_Caldas, «Sistemas de desarrollo: Raspberry Pi,» 2016. [En línea]. Available: <http://www.electronicoscaldas.com/boards-raspberry-pi/507-raspberry-pi-modelo-a-plus.html>. [Último acceso: 06 Febrero 2018].
- [50] Pc_Componentes, «Raspberry Pi 3 Modelo B,» [En línea]. Available: <https://www.pccomponentes.com/raspberry-pi-3-modelo-b>. [Último acceso: 06 Febrero 2018].

- [51] Medir_Temperatura, «Sensor de temperatura,» [En línea]. Available: <http://medirtemperatura.com/sensor-temperatura.php>. [Último acceso: 06 Febrero 2018].
- [52] C. Mathas, «Conceptos Básicos sobre sensores de temperatura,» *Biblioteca de artículos*, p. 2, 2011.
- [53] Programar_Facil, «DS18B20 sensor de temperatura para líquidos con Arduino,» 2016. [En línea]. Available: <https://programarfacil.com/blog/arduino-blog/ds18b20-sensor-temperatura-arduino/>. [Último acceso: 06 Febrero 2018].
- [54] MaximIntegrated, Programmable Resolution 1-Wire Digital Thermometer, Maxim Integrated Products, Inc. , 2015 .
- [55] Electronrtools, «Sensor de Temperatura – DS1820,» 2017. [En línea]. Available: <http://www.electronrtools.com/Home/WP/2016/03/09/sensor-de-temperatura-ds1820/>. [Último acceso: 06 Febrero 2018].
- [56] Libelium, «4G + GPS Shield for Arduino and Raspberry Pi Tutorial (LTE / WCDMA / HSPA+ / 3G / GPRS),» 2017. [En línea]. Available: <https://www.cooking-hacks.com/documentation/tutorials/4g-gps-lte-wcdma-hspa-3g-gprs-shield-arduino-raspberry-pi-waspmote-tutorial/>. [Último acceso: Enero 2018].
- [57] Huawei, «Modem Huawei E5377,» 2017. [En línea]. Available: <https://consumer.huawei.com/en/mobile-broadband/e5377/specs/>. [Último acceso: 06 Febrero 2018].
- [58] CONELEC, Atlas solar del Ecuador con fines de generación eléctrica, Quito: Corporación para la Investigación Energética, 2008.
- [59] Amazon, «Bateria AOMASO,» [En línea]. Available: https://www.amazon.com/42000mAh-Portable-All-purpose-Aomaso-Emergency/dp/B01LWOW6V3/ref=sr_1_fkmr0_1?ie=UTF8&qid=1516077960&sr=8-1-fkmr0&keywords=aomaso+battery+bank+42000mAh. [Último acceso: 06 Febrero 2018].
- [60] Amazon, «Cargador solar Giaride plegable,» [En línea]. Available: https://www.amazon.com/GIARIDE-Foldable-Charger-Sunpower-Portable/dp/B074GQF3CW/ref=sr_1_9?ie=UTF8&qid=1513762577&sr=8-9&keywords=paneles%2Bsolares%2B40w&th=1. [Último acceso: Noviembre 2017].
- [61] Adobe_Systems, «Notas de Dreamweaver,» [En línea]. Available: <https://www.adobe.com/support/documentation/es/dreamweaver/mx2004/releasenotes.htm>. [Último acceso: 06 Febrero 2018].

6. ANEXOS

ANEXO I. Programa para la adquisición de datos de los sensores temperatura_agua.py

ANEXO II. Programa que permite visualizar los datos de temperatura en el sitio web.

ANEXO III. Programa que permite visualizar las tendencias de temperatura en el sitio web.

ANEXO IV. Tabla de valores obtenidos en la laguna de Colta.

ANEXO V. Videos de las pruebas del prototipo en la laguna de Colta.

ANEXO I

Programa para la adquisición de datos de temperatura de los sensores, y para almacenar estos datos en la base. Programa: temperatura_agua.py

```
# =====  
# Programa para leer la temperatura del agua a 1, 3 y 5 metros  
# =====  
# Se importa librerías necesarias  
import sys  
import os  
import curses  
import time  
import MySQLdb  
from datetime import datetime  
  
# Carga de modulos para sensor DS18B20 OneWire  
os.system("sudo modprobe wire")  
os.system("sudo modprobe w1-gpio")  
os.system("sudo modprobe w1-therm")  
  
# Función para la lectura de temperatura del sensor a 1 metro  
def dametemp1():  
    tfile = open("/sys/bus/w1/devices/28-0517004bfdff/w1_slave")  
    lectura = tfile.read() # Se almacena la información del fichero en la variable lectura.  
    tfile.close() # Se cierra el fichero.  
    linea1 = lectura.split("\n")[1] # Se busca dentro del fichero el valor de t.  
    temperaturaleida = linea1.split(" ")[9]  
    temperatura = float(temperaturaleida[2:])  
    temperatura = temperatura / 1000  
    rtemp = round (temperatura,3) # Se redondea el valor de temperatura  
    return rtemp # La función retorna la variable rtemp  
  
def dametemp2(): #Se lee la temperatura del sensor a 3 metros con su dirección única  
    tfile = open("/sys/bus/w1/devices/28-0417007627ff/w1_slave")
```

```

lectura = tfile.read()
tfile.close()
linea1 = lectura.split("\n")[1]
temperaturaleida = linea1.split(" ")[9]
temperatura = float(temperaturaleida[2:])
temperatura = temperatura / 1000
rtemp = round (temperatura,3)           #Se redondea a tres decimales
return rtemp

```

```

def dametemp3():    #Se lee la temperatura del sensor a 5 metros con su dirección única
    tfile = open("/sys/bus/w1/devices/28-0417000965ff/w1_slave")
    lectura = tfile.read()
    tfile.close()
    linea1 = lectura.split("\n")[1]
    temperaturaleida = linea1.split(" ")[9]
    temperatura = float(temperaturaleida[2:])
    temperatura = temperatura / 1000
    rtemp = round (temperatura,3)  #Se redondea a tres decimales
    return rtemp

```

Programa para tomar los datos de los sensores, ponerlos en la base de datos y mostrarlos en pantalla

Utiliza curses para visualizar salida a pantalla

```
stdscr = curses.initscr()
```

```
curses.cbreak()
```

```
curses.start_color()
```

Definición de paletas de colores y borde

```
curses.init_pair(1, curses.COLOR_RED, curses.COLOR_BLACK)
```

```
curses.init_pair(2, curses.COLOR_YELLOW, curses.COLOR_BLACK)
```

```
stdscr.border(0)
```

```
hsize = curses.COLS
```

```
vsize = curses.LINES
```

```

stdscr.keypad(1)
stdscr.nodelay(1)
try:
    curses.curs_set(0)

# Rutina principal del programa
while True:
    char = stdscr.getch()
    if (char == 113 or char == 81):
        break # q
    else:
        conexion = MySQLdb.connect('localhost','sensores','c2875','monitoreo')
        cursor = conexion.cursor()

# Se ejecuta las funciones de cada sensor para almacenar el valor en la columna de la
# base de datos temperatura1, temperatura2 y temperatura3 respectivamente.
valor1 = "INSERT INTO
temperatura(temperatura1,temperatura2,temperatura3)VALUES(%s,%s,%s)"
%(dametemp1(), dametemp2(),dametemp3())
        cursor.execute(valor1)
        conexion.commit()
        cursor.close()
        conexion.close

#Se muestra los valores de los sensores en pantalla con su respectiva estampa de tiempo.
stdscr.addstr(vsize/6, (hsize/2)-18, "Temperatura Superficial del
Agua",curses.color_pair(1))
stdscr.addstr((vsize/5)+2, (hsize/2)- 30,"Temperatura a 1[m]= ")
stdscr.addstr((vsize/5)+2, (hsize/2)- 10 ,str(dametemp1()),curses.color_pair(2))
stdscr.addstr((vsize/5)+2, (hsize/2) ,str(datetime.now()),curses.color_pair(2))

stdscr.addstr((vsize/4)+2, (hsize/2)- 30,"Temperatura a 3[m]= ")
stdscr.addstr((vsize/4)+2, (hsize/2)- 10 ,str(dametemp2()),curses.color_pair(2))
stdscr.addstr((vsize/4)+2, (hsize/2) ,str(datetime.now()),curses.color_pair(2))

```

```

stdscr.addstr((vsize/3)+2, (hsize/2)- 30,"Temperatura a 5[m]= ")
stdscr.addstr((vsize/3)+2, (hsize/2)- 10 ,str(dametemp3()),curses.color_pair(2))
stdscr.addstr((vsize/3)+2, (hsize/2) ,str(datetime.now()),curses.color_pair(2))

stdscr.addstr(vsize-4, (hsize/2)- 14,"Pulsa (Q) para salir")
stdscr.refresh()
time.sleep(600)                                # Introduce el tiempo de espera, en segundos

# Salir del programa
finally:
    curses.nocbreak()
    stdscr.keypad(0)
    curses.echo()
    curses.endwin()                            # Se cierran los cursores

```

ANEXO II

Programa que permite visualizar los datos de temperatura almacenados en la base a través del sitio web. Programa: temperatura.php

```
<?php                                     // Conexión con la base de datos
$usuario = "sensores";
$password = "c2875";
$servidor = "localhost";
$basededatos = "monitoreo";

$conexion = mysqli_connect( $servidor, $usuario, $password ) or die ("No se ha podido
conectar al servidor de Base de datos");

$db = mysqli_select_db( $conexion, $basededatos ) or die ( "Upps! Pues va a ser que no
se ha podido conectar a la base de datos" );

?>
```

```
<?php                                     // Mostrar temperatura a 1 metro.
$consulta = "SELECT * FROM temperatura ORDER BY fecha desc limit 1";
$resultado = mysqli_query( $conexion, $consulta ) or die ( "Algo ha ido mal en la consulta
a la base de datos");

if ($row = mysqli_fetch_row($resultado)) {
    $id = trim($row[1]);
}

$id1m = number_format($id,3);
echo $id1m;                               //Muestra último registro

?>
```

```
<?php                                     // Mostrar estampa de fecha y hora
$consulta = "SELECT * FROM temperatura ORDER BY fecha desc limit 1";
$resultado = mysqli_query( $conexion, $consulta ) or die ( "Algo ha ido mal en la consulta
a la base de datos");
```

```

if ($row = mysqli_fetch_row($resultado)) {
    $id = trim($row[0]
}
echo $id; //Muestra último registro
?>

```

```

<?php // Mostrar temperatura a 3 metros.
$consulta = "SELECT * FROM temperatura ORDER BY fecha desc limit 1";
$resultado = mysqli_query( $conexion, $consulta ) or die ( "Algo ha ido mal en la consulta
a la base de datos");

```

```

if ($row = mysqli_fetch_row($resultado)) {
    $id = trim($row[2]);
}
$id3m = number_format($id,3);
echo $id3m; //Muestra último registro
?>

```

```

< php // Muestra estampa de fecha y hora.
$consulta = "SELECT * FROM temperatura ORDER BY fecha desc limit 1";
$resultado = mysqli_query( $conexion, $consulta ) or die ( "Algo ha ido mal en la consulta
a la base de datos");

```

```

if ($row = mysqli_fetch_row($resultado)) {
    $id = trim($row[0]);
}
echo $id; //Muestra último registro
?>

```

```

<?php // Mostrar temperatura a 5 metros.
$consulta = "SELECT * FROM temperatura ORDER BY fecha desc limit 1";
$resultado = mysqli_query( $conexion, $consulta ) or die ( "Algo ha ido mal en la consulta
a la base de datos");

```

```

if ($row = mysqli_fetch_row($resultado)) {
    $id = trim($row[3]
}
$id5m = number_format($id,3);
echo $id5m; //Muestra último registro
?>

```

```

[<?php                                     // Muestra estampa de fecha y hora.
$consulta = "SELECT * FROM temperatura ORDER BY fecha desc limit 1";
$resultado = mysqli_query( $conexion, $consulta ) or die ( "Algo ha ido mal en la consulta
a la base de datos");

```

```

if ($row = mysqli_fetch_row($resultado)) {
    $id = trim($row[0]
}
echo $id; //Muestra último registro
?>

```

```

<?php                                     // Mostrar valor mínimo
$consulta2 = "SELECT min(temperatura1) as minimo FROM temperatura ";
$resultado2 = mysqli_query( $conexion, $consulta2 ) or die ( "Algo ha ido mal en la consulta
a la base de datos");
if ($row2 = mysqli_fetch_row($resultado2)) {
    $id2 = trim($row2[0]);
}
$idmin1 = number_format($id2,3);
echo $idmin1; //Valor mínimo de columna
?>

```

```

<?php                                     // Mostrar valor máximo
$consulta3 = "SELECT max(temperatura1) as maximo FROM temperatura ";
$resultado3 = mysqli_query( $conexion, $consulta3 ) or die ( "Algo ha ido mal en la consulta
a la base de datos");

```

```

if ($row3 = mysqli_fetch_row($resultado3)) {
    $id3 = trim($row3[0]);
}
$idmax1 = number_format($id3,3);
echo $idmax1;      //Valor maximo de columna
?>

```

```

<?php                // Mostrar valor promedio.
$consulta4 = "SELECT  AVG(temperatura1) as temperatura1 FROM temperatura ";
$resultado4 = mysqli_query( $conexion, $consulta4 ) or die ( "Algo ha ido mal en la consulta
a la base de datos");
if ($row4 = mysqli_fetch_row($resultado4)) {
    $id4 = trim($row4[0]);
}
$id44 = number_format($id4,3);
echo $id44;  //Valor promedio de columna
?>

```

```

<?php                // Cierre de conexión con la base de datos
mysqli_close( $conexion );
?>

```

ANEXO III

Programa que permite visualizar las tendencias de temperatura a través del sitio web.

Programa: tendencia.php

```
<?php
require_once("RandomClass.php");

$usuario = "sensores";
$password = "c2875";
$servidor = "localhost";
$basededatos = "monitoreo";

//Variable que guarda la conexión de la base de datos
$conexion = mysqli_connect( $servidor, $usuario, $password, $basededatos ) or die ("No
se ha podido conectar al servidor de Base de datos");

// Selección de la base de datos a utilizar
$db = mysqli_select_db( $conexion, $basededatos ) or die ( "Ups! Pues va a ser que no se
ha podido conectar a la base de datos" );

//Se comprueba si la conexión ha tenido éxito
if(!$conexion){
    echo 'Ha sucedido un error inesperado en la conexión de la base de datos<br>';
}
?>

<?php
$sql = "SELECT * FROM temperatura";
if(!$result = mysqli_query($conexion, $sql)) die("mal echa consulta");
$rowdata = array();

//Se guarda en un array multidimensional todos los datos de la consulta
$i=0;
while($row = mysqli_fetch_array($result)) {
```

```
//Se guarda en rawdata todos los vectores/filas que devuelve la consulta
```

```
    $rawdata[$i] = $row;
```

```
    $i++;
```

```
    }
```

```
//Se crea dos arrays para almacenar el tiempo y el valor numérico
```

```
$valoresArray;
```

```
$timeArray;
```

```
//en un bucle for se obtiene en cada iteración el valor numérico y
```

```
//el TIMESTAMP del tiempo y se lo almacena en los arrays
```

```
for($i = 0 ;$i<count($rawdata);$i++){
```

```
    $valoresArray[$i]= $rawdata[$i][1];    //temperatura 1
```

```
    //SE OBTENIENE EL TIMESTAMP
```

```
    $time_real= $rawdata[$i][0];          //tiempo
```

```
    $time_change = strtotime('-5 hour', strtotime($time_real)); //Se cambia la hora para  
que aparezca la adecuada en la grafica
```

```
    $time = date('Y-m-d H:i:s', $time_change); //Se pone en el formato correcto la fecha  
y hora
```

```
    $date = new DateTime($time);
```

```
    //SE ALMACENA EL TIMESTAMP EN EL ARRAY
```

```
    $timeArray[$i] = $date ->getTimestamp()*1000;
```

```
    }
```

```
?>
```

```
<div id="contenedor1"></div>
```

```
<script src="https://code.jquery.com/jquery.js"></script>
```

```
<!-- Importo el archivo Javascript de Highcharts directamente desde su servidor -->
```

```
<script src="http://code.highcharts.com/stock/highstock.js"></script>
```

```
<script src="http://code.highcharts.com/modules/exporting.js"></script>
```

```
<script>
```

```

chartCPU = new Highcharts.StockChart({
  chart: {
    renderTo: 'contenedor1'
    //defaultSeriesType: 'spline'
  },

  rangeSelector : {
    enabled: false
  },

  title: {
    text: 'TEMPERATURA DEL AGUA A 1[m]'
  },

  xAxis: {
    type: 'datetime'
    //tickPixelInterval: 150
    //maxZoom: 20 * 1000
  },

  yAxis: {
    minPadding: 0.2,
    maxPadding: 0.2,
    title: {
      text: 'Temperatura',
      margin: 10
    }
  },

  series: [{
    name: 'valor',
    data: (function() {
      // generate an array of random data

```

```

        var data = [];
        <?php
            for($i = 0 ;$i<count($rawdata);$i++){
?>

            data.push([<?php echo $timeArray[$i];?>,<?php echo $valoresArray[$i];?>]);
            <?php } ?>
            return data;
        })()
    }],
    credits: {
        enabled: false
    }
});

```

```
</script></p>
```

```
<p>&nbsp;</p>
```

```
<p class="subtitulo"><strong><u>TENDENCIA A 3[m]:</u></strong></p>
```

```
<p class="subtitulo"><?php
```

```
$sql = "SELECT * FROM temperatura";
```

```
if(!$result = mysqli_query($conexion, $sql)) die("mal echa consulta");
```

```
$rawdata = array();
```

```
$i=0;
```

```
while($row = mysqli_fetch_array($result)) {
```

```
    $rawdata[$i] = $row;
```

```
    $i++;    }
```

```
//se crea dos arrays para almacenar el tiempo y el valor numérico
```

```
$valoresArray;
```

```
$timeArray;
```

```
for($i = 0 ;$i<count($rawdata);$i++){
```

```

        $valoresArray[$i]= $rawdata[$i][2];    //temperatura 1
        $time_real= $rawdata[$i][0];          //tiempo
        $time_change = strtotime('-5 hour', strtotime($time_real)); //Se cambia la hora para
que aparezca la adecuada en la grafica
        $time = date('Y-m-d H:i:s', $time_change); //Se pone en el formato correcto la fecha
y hora
        $date = new DateTime($time);
        $timeArray[$i] = $date ->getTimestamp()*1000;
    }

```

?>

```

<div id="contenedor2"></div>
<p>
<script src="https://code.jquery.com/jquery.js"></script>
<!-- Importo el archivo Javascript de Highcharts directamente desde su servidor -->
<script src="http://code.highcharts.com/stock/highstock.js"></script>
<script src="http://code.highcharts.com/modules/exporting.js"></script>
<script>

```

```

chartCPU = new Highcharts.StockChart({
    chart: {
        renderTo: 'contenedor2'
        //defaultSeriesType: 'spline'
    },
    rangeSelector : {
        enabled: false
    },
    title: {
        text: 'TEMPERATURA DEL AGUA A 3[m]'
    },
    xAxis: {
        type: 'datetime'
        //tickPixelInterval: 150
        //maxZoom: 20 * 1000
    }

```

```

    },
    yAxis: {
        minPadding: 0.2,
        maxPadding: 0.2,
        title: {
            text: 'Temperatura',
            margin: 10
        }
    },
    series: [{
        name: 'valor',
        data: (function() {
            // generate an array of random data
            var data = [];
            <?php
                for($i = 0 ;$i<count($rawdata);$i++){
            ?>
            data.push([<?php echo $timeArray[$i];?>,<?php echo $valoresArray[$i];?>]);
            <?php } ?>
            return data;
        })()
    }],
    credits: {
        enabled: false
    }
});
<?php
$sql = "SELECT * FROM temperatura";
if(!$result = mysqli_query($conexion, $sql)) die("mal echa consulta");
    $rawdata = array();
    $i=0;
    while($row = mysqli_fetch_array($result))

```

```

        {
            $rawdata[$i] = $row;
            $i++;
        }
$valoresArray;
$timeArray;
for($i = 0 ;$i<count($rawdata);$i++){
    $valoresArray[$i]= $rawdata[$i][3];    //temperatura 3
    $time_real= $rawdata[$i][0];          //tiempo
    $time_change = strtotime('-5 hour', strtotime($time_real));
    $time = date('Y-m-d H:i:s', $time_change);
    $date = new DateTime($time);
    $timeArray[$i] = $date ->getTimestamp()*1000;
}
?>
<div id="contenedor3"></div>
<script src="https://code.jquery.com/jquery.js"></script>
<!-- Importo el archivo Javascript de Highcharts directamente desde su servidor -->
<script src="http://code.highcharts.com/stock/highstock.js"></script>
<script src="http://code.highcharts.com/modules/exporting.js"></script>
<script>
chartCPU = new Highcharts.StockChart({
    chart: {
        renderTo: 'contenedor3'
        //defaultSeriesType: 'spline'
    },
    rangeSelector : {
        enabled: false
    },
    title: {
        text: 'TEMPERATURA DEL AGUA A 5[m]'
    },
    xAxis: {

```

```

    type: 'datetime'
    //tickPixelInterval: 150
    //maxZoom: 20 * 1000
  },
  yAxis: {
    minPadding: 0.2,
    maxPadding: 0.2,
    title: {
      text: 'Temperatura',
      margin: 10
    }
  },
  series: [{
    name: 'valor',
    data: (function() {
      // generate an array of random data
      var data = [];
      <?php
        for($i = 0 ;$i<count($rawdata);$i++){
      ?>
      data.push([<?php echo $timeArray[$i];?>,<?php echo $valoresArray[$i];?>]);
      <?php } ?>
      return data;
    })()
  }],
  credits: {
    enabled: false
  }
});
<?php
$close = mysqli_close($conexion);
?>

```

ANEXO IV

Tabla de valores de temperatura obtenidos en la laguna de Colta.

Tabla IV.1. Valores obtenidos de las pruebas en la laguna de Colta.

Fecha / Hora	Sensor a 1m	Sensor a 3m	Sensor a 5m
2/14/2018 8:20	14.562	14.062	14.125
2/14/2018 8:31	14.812	14.375	14.375
2/14/2018 8:31	13.375	13.812	13.687
2/14/2018 10:38	13.375	14.625	14.250
2/14/2018 10:49	13.812	15.625	14.875
2/14/2018 10:59	12.125	16.250	14.937
2/14/2018 11:09	16.437	16.500	16.312
2/14/2018 11:19	16.437	16.437	16.312
2/14/2018 11:29	16.437	16.437	16.312
2/14/2018 11:39	16.437	16.437	16.312
2/14/2018 11:49	16.437	16.375	16.312
2/14/2018 11:59	16.500	16.375	16.312
2/14/2018 12:09	16.500	16.375	16.312
2/14/2018 12:19	16.500	16.375	16.375
2/14/2018 12:29	16.500	16.375	16.375
2/14/2018 12:40	16.500	16.375	16.312
2/14/2018 12:50	16.500	16.375	16.375
2/14/2018 13:00	16.500	16.375	16.312
2/14/2018 13:10	16.562	16.437	16.375
2/14/2018 13:20	16.562	16.437	16.375
2/14/2018 13:30	16.562	16.500	16.375
2/14/2018 13:40	16.562	16.500	16.375
2/14/2018 13:50	16.562	16.500	16.437
2/14/2018 14:00	16.562	16.500	16.375
2/14/2018 14:10	16.625	16.500	16.437
2/14/2018 14:20	16.687	16.562	16.437
2/14/2018 14:31	16.687	16.562	16.437
2/14/2018 14:41	16.625	16.562	16.437
2/14/2018 14:51	16.625	16.562	16.437
2/14/2018 15:01	16.687	16.562	16.437
2/14/2018 15:11	16.687	16.562	16.437
2/14/2018 15:21	16.687	16.562	16.437
2/14/2018 15:31	16.687	16.562	16.437
2/14/2018 15:41	16.687	16.562	16.500
2/14/2018 15:51	16.687	16.562	16.437
2/14/2018 16:01	16.687	16.562	16.437
2/14/2018 16:11	16.687	16.562	16.437
2/14/2018 16:22	16.687	16.562	16.437

2/14/2018 16:32	16.687	16.562	16.437
2/14/2018 16:42	16.687	16.562	16.437
2/14/2018 16:52	16.687	16.562	16.437
2/14/2018 17:02	16.687	16.562	16.437
2/14/2018 17:12	16.625	16.562	16.437
2/14/2018 17:22	16.625	16.562	16.437
2/14/2018 17:32	16.625	16.562	16.437
2/14/2018 17:42	16.625	16.562	16.437
2/14/2018 17:52	16.625	16.562	16.437
2/14/2018 18:02	16.562	16.500	16.437
2/14/2018 18:13	16.625	16.500	16.375
2/14/2018 18:23	16.562	16.500	16.375
2/14/2018 18:33	16.562	16.562	16.375
2/14/2018 18:43	16.562	16.500	16.375
2/14/2018 18:53	16.562	16.500	16.375
2/14/2018 19:03	16.562	16.500	16.437
2/14/2018 19:13	16.500	16.500	16.437
2/14/2018 19:23	16.500	16.500	16.375
2/14/2018 19:33	16.500	16.500	16.375
2/14/2018 19:44	16.500	16.500	16.375
2/14/2018 19:54	16.500	16.500	16.375
2/14/2018 20:04	16.500	16.437	16.375
2/14/2018 20:14	16.437	16.437	16.375
2/14/2018 20:24	16.437	16.437	16.375
2/14/2018 20:34	16.437	16.437	16.375
2/14/2018 20:44	16.437	16.437	16.375
2/14/2018 20:54	16.375	16.500	16.375
2/14/2018 21:04	16.375	16.437	16.375
2/14/2018 21:14	16.375	16.500	16.375
2/14/2018 21:24	16.375	16.500	16.375
2/14/2018 21:35	16.375	16.437	16.375
2/14/2018 21:45	16.312	16.437	16.375
2/14/2018 21:55	16.312	16.437	16.375
2/14/2018 22:05	16.312	16.437	16.375
2/14/2018 22:15	16.312	16.437	16.375
2/14/2018 22:25	16.250	16.375	16.375
2/14/2018 22:35	16.250	16.375	16.375
2/14/2018 22:45	16.250	16.312	16.312
2/14/2018 22:55	16.250	16.312	16.312
2/14/2018 23:05	16.187	16.312	16.375
2/14/2018 23:15	16.187	16.312	16.375
2/14/2018 23:25	16.187	16.250	16.375
2/14/2018 23:36	16.187	16.250	16.312
2/14/2018 23:46	16.187	16.312	16.312

2/14/2018 23:56	16.187	16.312	16.312
2/15/2018 0:06	16.125	16.250	16.250
2/15/2018 0:16	16.125	16.187	16.250
2/15/2018 0:26	16.125	16.187	16.250
2/15/2018 0:36	16.125	16.187	16.187
2/15/2018 0:46	16.062	16.187	16.125
2/15/2018 0:56	16.062	16.187	16.125
2/15/2018 1:06	16.062	16.125	16.125
2/15/2018 1:17	16.062	16.125	16.125
2/15/2018 1:27	16.062	16.125	16.062
2/15/2018 1:37	16.000	16.125	16.062
2/15/2018 1:47	16.000	16.125	16.062
2/15/2018 1:57	16.000	16.125	16.000
2/15/2018 2:07	16.000	16.062	16.000
2/15/2018 2:17	16.000	16.062	16.000
2/15/2018 2:27	16.000	16.062	16.000
2/15/2018 2:37	16.000	16.062	16.000
2/15/2018 2:47	16.000	16.062	16.000
2/15/2018 2:57	15.937	16.062	15.937
2/15/2018 3:08	16.000	16.062	15.937
2/15/2018 3:18	16.000	16.062	15.937
2/15/2018 3:28	15.937	16.062	15.937
2/15/2018 3:38	15.937	16.000	15.937
2/15/2018 3:48	15.937	16.000	15.937
2/15/2018 3:58	15.937	16.000	15.937
2/15/2018 4:08	15.937	16.000	15.937
2/15/2018 4:18	15.937	16.000	15.875
2/15/2018 4:28	15.937	15.937	15.875
2/15/2018 4:38	15.875	15.937	15.875
2/15/2018 4:48	15.875	15.937	15.875
2/15/2018 4:59	15.875	15.937	15.875
2/15/2018 5:09	15.875	15.937	15.875
2/15/2018 5:19	15.875	15.937	15.875
2/15/2018 5:29	15.875	15.937	15.812
2/15/2018 5:39	15.875	15.875	15.875
2/15/2018 5:49	15.812	15.875	15.875
2/15/2018 5:59	15.812	15.875	15.812
2/15/2018 6:09	15.812	15.875	15.812
2/15/2018 6:19	15.812	15.875	15.812
2/15/2018 6:29	15.812	15.875	15.812
2/15/2018 6:39	15.812	15.812	15.750
2/15/2018 6:50	15.812	15.875	15.750
2/15/2018 7:00	15.812	15.812	15.750
2/15/2018 7:10	15.812	15.812	15.750

2/15/2018 7:20	15.812	15.812	15.750
2/15/2018 7:30	15.812	15.812	15.750
2/15/2018 7:40	15.812	15.812	15.750
2/15/2018 7:50	15.750	15.812	15.750
2/15/2018 8:00	15.750	15.812	15.750
2/15/2018 8:10	15.812	15.812	15.750
2/15/2018 8:20	15.812	15.812	15.750
2/15/2018 8:30	15.750	15.812	15.750
2/15/2018 8:40	15.750	15.812	15.750
2/15/2018 8:51	15.750	15.750	15.687
2/15/2018 9:01	15.812	15.750	15.687
2/15/2018 9:11	15.812	15.750	15.687
2/15/2018 9:21	15.875	15.750	15.687
2/15/2018 9:31	15.875	15.750	15.687
2/15/2018 9:41	15.937	15.750	15.687
2/15/2018 9:51	15.937	15.750	15.687
2/15/2018 10:01	16.000	15.750	15.687
2/15/2018 10:11	16.062	15.750	15.687
2/15/2018 10:21	16.125	15.750	15.687
2/15/2018 10:31	16.125	15.750	15.687
2/15/2018 10:42	16.125	15.812	15.750
2/15/2018 10:52	16.187	15.812	15.687
2/15/2018 11:02	16.125	15.812	15.687
2/15/2018 11:12	16.187	15.812	15.687
2/15/2018 11:22	16.250	15.875	15.750
2/15/2018 11:32	16.250	15.875	15.750
2/15/2018 11:42	16.250	15.875	15.812
2/15/2018 11:52	16.312	15.937	15.812
2/15/2018 12:02	16.312	15.937	15.812
2/15/2018 12:12	16.375	15.937	15.812
2/15/2018 12:22	16.375	16.000	15.875
2/15/2018 12:33	16.500	16.000	15.875
2/15/2018 12:43	16.562	16.000	15.875
2/15/2018 12:53	16.625	16.062	15.937
2/15/2018 13:03	16.625	16.062	15.937
2/15/2018 13:13	16.625	16.125	15.937
2/15/2018 13:23	16.625	16.187	16.000
2/15/2018 13:33	16.625	16.187	16.000
2/15/2018 13:43	16.625	16.250	16.000
2/15/2018 13:53	16.625	16.250	16.000
2/15/2018 14:03	16.625	16.250	16.000
2/15/2018 14:13	16.562	16.250	16.000
2/15/2018 14:24	16.562	16.250	16.000
2/15/2018 14:34	16.562	16.250	16.062

2/15/2018 14:44	16.562	16.250	16.062
2/15/2018 14:54	16.500	16.312	16.062
2/15/2018 15:04	16.500	16.312	16.062
2/15/2018 15:14	16.500	16.312	16.062
2/15/2018 15:24	16.500	16.312	16.125
2/15/2018 15:34	16.437	16.312	16.125
2/15/2018 15:44	16.500	16.312	16.125
2/15/2018 15:54	16.437	16.312	16.125
2/15/2018 16:04	16.437	16.312	16.125
2/15/2018 16:15	16.500	16.312	16.125
2/15/2018 16:25	16.437	16.312	16.125
2/15/2018 16:35	16.437	16.312	16.125
2/15/2018 16:45	16.437	16.312	16.125
2/15/2018 16:55	16.437	16.312	16.125
2/15/2018 17:05	16.437	16.312	16.125
2/15/2018 17:15	16.437	16.312	16.125
2/15/2018 17:25	16.375	16.312	16.125
2/15/2018 17:35	16.437	16.312	16.125
2/15/2018 17:45	16.375	16.312	16.125
2/15/2018 17:55	16.375	16.312	16.125
2/15/2018 18:06	16.375	16.312	16.125
2/15/2018 18:16	16.312	16.312	16.187
2/15/2018 18:26	16.312	16.312	16.187
2/15/2018 18:36	16.312	16.312	16.125
2/15/2018 18:46	16.312	16.312	16.125
2/15/2018 18:56	16.312	16.312	16.125
2/15/2018 19:06	16.312	16.312	16.125
2/15/2018 19:16	16.312	16.312	16.125
2/15/2018 19:26	16.250	16.312	16.187
2/15/2018 19:36	16.250	16.312	16.125
2/15/2018 19:46	16.250	16.250	16.125
2/15/2018 19:57	16.250	16.250	16.125
2/15/2018 20:07	16.250	16.250	16.187
2/15/2018 20:17	16.250	16.250	16.187
2/15/2018 20:27	16.250	16.250	16.187
2/15/2018 20:37	16.250	16.312	16.187
2/15/2018 20:47	16.187	16.250	16.187
2/15/2018 20:57	16.187	16.250	16.187
2/15/2018 21:07	16.187	16.250	16.187
2/15/2018 21:17	16.187	16.250	16.187
2/15/2018 21:27	16.187	16.250	16.187
2/15/2018 21:37	16.125	16.250	16.187
2/15/2018 21:48	16.187	16.250	16.187
2/15/2018 21:58	16.125	16.250	16.187

2/15/2018 22:08	16.125	16.250	16.187
2/15/2018 22:18	16.125	16.250	16.187
2/15/2018 22:28	16.125	16.250	16.187
2/15/2018 22:38	16.125	16.187	16.187
2/15/2018 22:48	16.125	16.187	16.187
2/15/2018 22:58	16.125	16.187	16.125
2/15/2018 23:08	16.062	16.187	16.125
2/15/2018 23:18	16.062	16.187	16.125
2/15/2018 23:28	16.062	16.187	16.125
2/15/2018 23:39	16.062	16.125	16.125
2/15/2018 23:49	16.062	16.125	16.125
2/15/2018 23:59	16.062	16.125	16.187
2/16/2018 0:09	16.062	16.125	16.125
2/16/2018 0:19	16.000	16.125	16.125
2/16/2018 0:29	16.000	16.125	16.125
2/16/2018 0:39	16.000	16.062	16.125
2/16/2018 0:49	16.000	16.062	16.125
2/16/2018 0:59	16.000	16.062	16.062
2/16/2018 1:09	16.000	16.062	16.062
2/16/2018 1:19	16.000	16.062	16.062
2/16/2018 1:30	15.937	16.000	16.062
2/16/2018 1:40	15.937	16.000	16.000
2/16/2018 1:50	15.937	16.000	15.937
2/16/2018 2:00	15.937	16.000	15.937
2/16/2018 2:10	15.937	16.000	15.937
2/16/2018 2:20	15.937	15.937	15.875
2/16/2018 2:30	15.875	15.937	15.875
2/16/2018 2:40	15.875	15.937	15.875
2/16/2018 2:50	15.875	15.937	15.875
2/16/2018 3:00	15.875	15.937	15.875
2/16/2018 3:10	15.875	15.937	15.875
2/16/2018 3:21	15.875	15.937	15.812
2/16/2018 3:31	15.875	15.937	15.812
2/16/2018 3:41	15.875	15.937	15.875
2/16/2018 3:51	15.812	15.875	15.875
2/16/2018 4:01	15.812	15.875	15.812
2/16/2018 4:11	15.875	15.875	15.812
2/16/2018 4:21	15.812	15.875	15.750
2/16/2018 4:31	15.812	15.875	15.750
2/16/2018 4:41	15.812	15.875	15.750
2/16/2018 4:51	15.812	15.875	15.812
2/16/2018 5:01	15.812	15.875	15.750
2/16/2018 5:12	15.812	15.812	15.750
2/16/2018 5:22	15.812	15.812	15.750

2/16/2018 5:32	15.750	15.812	15.750
2/16/2018 5:42	15.750	15.812	15.687
2/16/2018 5:52	15.750	15.812	15.812
2/16/2018 6:02	15.750	15.812	15.750
2/16/2018 6:12	15.750	15.812	15.687
2/16/2018 6:22	15.750	15.750	15.687
2/16/2018 6:32	15.750	15.750	15.687
2/16/2018 6:42	15.687	15.750	15.625
2/16/2018 6:52	15.687	15.750	15.625
2/16/2018 7:03	15.687	15.750	15.625
2/16/2018 7:13	15.687	15.750	15.625
2/16/2018 7:23	15.687	15.750	15.625
2/16/2018 7:33	15.687	15.750	15.625
2/16/2018 7:43	15.687	15.687	15.625
2/16/2018 7:53	15.687	15.687	15.625
2/16/2018 8:03	15.687	15.687	15.625
2/16/2018 8:13	15.687	15.687	15.625
2/16/2018 8:23	15.687	15.687	15.625
2/16/2018 8:33	15.687	15.687	15.625
2/16/2018 8:43	15.687	15.687	15.625
2/16/2018 8:54	15.687	15.625	15.625
2/16/2018 9:04	15.687	15.687	15.625
2/16/2018 9:14	15.687	15.687	15.625
2/16/2018 9:24	15.750	15.687	15.625
2/16/2018 9:34	15.750	15.625	15.625
2/16/2018 9:44	15.750	15.625	15.625
2/16/2018 9:54	15.750	15.625	15.562
2/16/2018 10:04	15.812	15.625	15.562
2/16/2018 10:14	15.750	15.687	15.625
2/16/2018 10:24	15.812	15.625	15.625
2/16/2018 10:34	15.875	15.687	15.625
2/16/2018 10:45	15.875	15.687	15.625
2/16/2018 10:55	15.875	15.687	15.625
2/16/2018 11:05	15.875	15.687	15.625
2/16/2018 11:15	15.875	15.687	15.625
2/16/2018 11:25	15.875	15.687	15.687
2/16/2018 11:35	15.875	15.750	15.687
2/16/2018 11:45	15.875	15.687	15.687
2/16/2018 11:55	15.875	15.687	15.687
2/16/2018 12:05	15.812	15.312	14.875
2/16/2018 12:15	12.812	13.750	13.000
2/16/2018 12:25	16.375	16.000	15.875
2/16/2018 12:36	16.500	16.000	15.875
2/16/2018 12:46	16.562	16.000	15.875

2/16/2018 12:56	16.625	16.062	15.937
2/16/2018 13:06	16.625	16.062	15.937
2/16/2018 13:16	16.625	16.125	15.937
2/16/2018 13:26	16.625	16.187	16.000
2/16/2018 13:36	16.625	16.187	16.000
2/16/2018 13:46	16.625	16.250	16.000
2/16/2018 13:56	16.625	16.250	16.000
2/16/2018 14:06	16.625	16.250	16.000
2/16/2018 14:16	16.562	16.250	16.000
2/16/2018 14:27	16.562	16.250	16.000
2/16/2018 14:37	16.562	16.250	16.062
2/16/2018 14:47	16.562	16.250	16.062
2/16/2018 14:57	16.500	16.312	16.062
2/16/2018 15:07	16.500	16.312	16.062
2/16/2018 15:17	16.500	16.312	16.062
2/16/2018 15:27	16.500	16.312	16.125
2/16/2018 15:37	16.437	16.312	16.125
2/16/2018 15:47	16.500	16.312	16.125
2/16/2018 15:57	16.437	16.312	16.125
2/16/2018 16:07	16.437	16.312	16.125
2/16/2018 16:18	16.500	16.312	16.125
2/16/2018 16:28	16.437	16.312	16.125
2/16/2018 16:38	16.437	16.312	16.125
2/16/2018 16:48	16.437	16.312	16.125
2/16/2018 16:58	16.437	16.312	16.125
2/16/2018 17:08	16.437	16.312	16.125
2/16/2018 17:18	16.437	16.312	16.125
2/16/2018 17:28	16.375	16.312	16.125
2/16/2018 17:38	16.437	16.312	16.125
2/16/2018 17:48	16.375	16.312	16.125
2/16/2018 17:58	16.375	16.312	16.125
2/16/2018 18:09	16.375	16.312	16.125
2/16/2018 18:19	16.312	16.312	16.187
2/16/2018 18:29	16.312	16.312	16.187
2/16/2018 18:39	16.312	16.312	16.125
2/16/2018 18:49	16.312	16.312	16.125
2/16/2018 18:59	16.312	16.312	16.125
2/16/2018 19:09	16.312	16.312	16.125
2/16/2018 19:19	16.312	16.312	16.125
2/16/2018 19:29	16.250	16.312	16.187
2/16/2018 19:39	16.250	16.312	16.125
2/16/2018 19:49	16.250	16.250	16.125
2/16/2018 20:00	16.250	16.250	16.125
2/16/2018 20:10	16.250	16.250	16.187

2/16/2018 20:20	16.250	16.250	16.187
2/16/2018 20:30	16.250	16.250	16.187
2/16/2018 20:40	16.250	16.312	16.187
2/16/2018 20:50	16.187	16.250	16.187
2/16/2018 21:00	16.187	16.250	16.187
2/16/2018 21:10	16.187	16.250	16.187
2/16/2018 21:20	16.187	16.250	16.187
2/16/2018 21:30	16.187	16.250	16.187
2/16/2018 21:40	16.125	16.250	16.187
2/16/2018 21:51	16.187	16.250	16.187
2/16/2018 22:01	16.125	16.250	16.187
2/16/2018 22:11	16.125	16.250	16.187
2/16/2018 22:21	16.125	16.250	16.187
2/16/2018 22:31	16.125	16.250	16.187
2/16/2018 22:41	16.125	16.187	16.187
2/16/2018 22:51	16.125	16.187	16.187
2/16/2018 23:01	16.125	16.187	16.125
2/16/2018 23:11	16.062	16.187	16.125
2/16/2018 23:21	16.062	16.187	16.125
2/16/2018 23:31	16.062	16.187	16.125
2/16/2018 23:42	16.062	16.125	16.125
2/16/2018 23:52	16.062	16.125	16.125
2/17/2018 0:02	16.062	16.125	16.187
2/17/2018 0:12	16.062	16.125	16.125
2/17/2018 0:22	16.000	16.125	16.125
2/17/2018 0:32	16.000	16.125	16.125
2/17/2018 0:42	16.000	16.062	16.125
2/17/2018 0:52	16.000	16.062	16.125
2/17/2018 1:02	16.000	16.062	16.062
2/17/2018 1:12	16.000	16.062	16.062
2/17/2018 1:22	16.000	16.062	16.062
2/17/2018 1:33	15.937	16.000	16.062
2/17/2018 1:43	15.937	16.000	16.000
2/17/2018 1:53	15.937	16.000	15.937
2/17/2018 2:03	15.937	16.000	15.937
2/17/2018 2:13	15.937	16.000	15.937
2/17/2018 2:23	15.937	15.937	15.875
2/17/2018 2:33	15.875	15.937	15.875
2/17/2018 2:43	15.875	15.937	15.875
2/17/2018 2:53	15.875	15.937	15.875
2/17/2018 3:03	15.875	15.937	15.875
2/17/2018 3:13	15.875	15.937	15.875
2/17/2018 3:24	15.875	15.937	15.812
2/17/2018 3:34	15.875	15.937	15.812

2/17/2018 3:44	15.875	15.937	15.875
2/17/2018 3:54	15.812	15.875	15.875
2/17/2018 4:04	15.812	15.875	15.812
2/17/2018 4:14	15.875	15.875	15.812
2/17/2018 4:24	15.812	15.875	15.750
2/17/2018 4:34	15.812	15.875	15.750
2/17/2018 4:44	15.812	15.875	15.750
2/17/2018 4:54	15.812	15.875	15.812
2/17/2018 5:04	15.812	15.875	15.750
2/17/2018 5:15	15.812	15.812	15.750
2/17/2018 5:25	15.812	15.812	15.750
2/17/2018 5:35	15.750	15.812	15.750
2/17/2018 5:45	15.750	15.812	15.687
2/17/2018 5:55	15.750	15.812	15.812
2/17/2018 6:05	15.750	15.812	15.750
2/17/2018 6:15	15.750	15.812	15.687
2/17/2018 6:25	15.750	15.750	15.687
2/17/2018 6:35	15.750	15.750	15.687
2/17/2018 6:45	15.687	15.750	15.625
2/17/2018 6:55	15.687	15.750	15.625
2/17/2018 7:06	15.687	15.750	15.625
2/17/2018 7:16	15.687	15.750	15.625
2/17/2018 7:26	15.687	15.750	15.625
2/17/2018 7:36	15.687	15.750	15.625
2/17/2018 7:46	15.687	15.687	15.625
2/17/2018 7:56	15.687	15.687	15.625
2/17/2018 8:06	15.687	15.687	15.625
2/17/2018 8:16	15.687	15.687	15.625
2/17/2018 8:26	15.687	15.687	15.625
2/17/2018 8:36	15.687	15.687	15.625
2/17/2018 8:46	15.687	15.687	15.625
2/17/2018 8:57	15.687	15.625	15.625
2/17/2018 9:07	15.687	15.687	15.625
2/17/2018 9:17	15.687	15.687	15.625
2/17/2018 9:27	15.750	15.687	15.625
2/17/2018 9:37	15.750	15.625	15.625
2/17/2018 9:47	15.750	15.625	15.625
2/17/2018 9:57	15.750	15.625	15.562
2/17/2018 10:07	15.812	15.625	15.562
2/17/2018 10:17	15.750	15.687	15.625
2/17/2018 10:27	15.812	15.625	15.625
2/17/2018 10:37	15.875	15.687	15.625
2/17/2018 10:48	15.875	15.687	15.625
2/17/2018 10:58	15.875	15.687	15.625

2/17/2018 11:08	15.875	15.687	15.625
2/17/2018 11:18	15.875	15.687	15.625
2/17/2018 11:28	15.875	15.687	15.687
2/17/2018 11:38	15.875	15.750	15.687
2/17/2018 11:48	15.875	15.687	15.687
2/17/2018 11:58	15.875	15.687	15.687
2/17/2018 12:08	15.812	15.312	14.875
2/17/2018 12:18	12.812	13.750	13.000
2/17/2018 12:28	16.500	16.375	16.375
2/17/2018 12:39	16.500	16.375	16.312
2/17/2018 12:49	16.500	16.375	16.375
2/17/2018 12:59	16.500	16.375	16.312
2/17/2018 13:09	16.562	16.437	16.375
2/17/2018 13:19	16.562	16.437	16.375
2/17/2018 13:29	16.562	16.500	16.375
2/17/2018 13:39	16.562	16.500	16.375
2/17/2018 13:49	16.562	16.500	16.437
2/17/2018 13:59	16.562	16.500	16.375
2/17/2018 14:09	16.625	16.500	16.437
2/17/2018 14:19	16.687	16.562	16.437
2/17/2018 14:30	16.687	16.562	16.437
2/17/2018 14:40	16.625	16.562	16.437
2/17/2018 14:50	16.625	16.562	16.437
2/17/2018 15:00	16.687	16.562	16.437
2/17/2018 15:10	16.687	16.562	16.437
2/17/2018 15:20	16.687	16.562	16.437
2/17/2018 15:30	16.687	16.562	16.437
2/17/2018 15:40	16.687	16.562	16.500
2/17/2018 15:50	16.687	16.562	16.437
2/17/2018 16:00	16.687	16.562	16.437
2/17/2018 16:10	16.687	16.562	16.437
2/17/2018 16:21	16.687	16.562	16.437
2/17/2018 16:31	16.687	16.562	16.437
2/17/2018 16:41	16.687	16.562	16.437
2/17/2018 16:51	16.687	16.562	16.437
2/17/2018 17:01	16.687	16.562	16.437
2/17/2018 17:11	16.625	16.562	16.437
2/17/2018 17:21	16.625	16.562	16.437
2/17/2018 17:31	16.625	16.562	16.437
2/17/2018 17:41	16.625	16.562	16.437
2/17/2018 17:51	16.625	16.562	16.437
2/17/2018 18:01	16.562	16.500	16.437
2/17/2018 18:12	16.625	16.500	16.375
2/17/2018 18:22	16.562	16.500	16.375

2/17/2018 18:32	16.562	16.562	16.375
2/17/2018 18:42	16.562	16.500	16.375
2/17/2018 18:52	16.562	16.500	16.375
2/17/2018 19:02	16.562	16.500	16.437
2/17/2018 19:12	16.500	16.500	16.437
2/17/2018 19:22	16.500	16.500	16.375
2/17/2018 19:32	16.500	16.500	16.375
2/17/2018 19:42	16.500	16.500	16.375
2/17/2018 19:52	16.500	16.500	16.375
2/17/2018 20:03	16.500	16.437	16.375
2/17/2018 20:13	16.437	16.437	16.375
2/17/2018 20:23	16.437	16.437	16.375
2/17/2018 20:33	16.437	16.437	16.375
2/17/2018 20:43	16.437	16.437	16.375
2/17/2018 20:53	16.375	16.500	16.375
2/17/2018 21:03	16.375	16.437	16.375
2/17/2018 21:13	16.375	16.500	16.375
2/17/2018 21:23	16.375	16.500	16.375
2/17/2018 21:33	16.375	16.437	16.375
2/17/2018 21:43	16.312	16.437	16.375
2/17/2018 21:54	16.312	16.437	16.375
2/17/2018 22:04	16.312	16.437	16.375
2/17/2018 22:14	16.312	16.437	16.375
2/17/2018 22:24	16.250	16.375	16.375
2/17/2018 22:34	16.250	16.375	16.375
2/17/2018 22:44	16.250	16.312	16.312
2/17/2018 22:54	16.250	16.312	16.312
2/17/2018 23:04	16.187	16.312	16.375
2/17/2018 23:14	16.187	16.312	16.375
2/17/2018 23:24	16.187	16.250	16.375
2/17/2018 22:34	16.187	16.250	16.312
2/17/2018 22:45	16.187	16.312	16.312
2/17/2018 22:55	16.187	16.312	16.312
2/17/2018 23:05	16.125	16.250	16.250
2/17/2018 23:15	16.125	16.187	16.250
2/17/2018 23:25	16.125	16.187	16.250
2/17/2018 23:35	16.125	16.187	16.187
2/17/2018 23:45	16.062	16.187	16.125
2/17/2018 23:55	16.062	16.187	16.125
2/18/2018 0:05	16.062	16.125	16.125
2/18/2018 0:15	16.062	16.125	16.125
2/18/2018 0:25	16.062	16.125	16.062
2/18/2018 0:36	16.000	16.125	16.062
2/18/2018 0:46	16.000	16.125	16.062

2/18/2018 0:56	16.000	16.125	16.000
2/18/2018 1:06	16.000	16.062	16.000
2/18/2018 1:16	16.000	16.062	16.000
2/18/2018 1:26	16.000	16.062	16.000
2/18/2018 1:36	16.000	16.062	16.000
2/18/2018 1:46	16.000	16.062	16.000
2/18/2018 1:56	15.937	16.062	15.937
2/18/2018 2:06	16.000	16.062	15.937
2/18/2018 2:16	16.000	16.062	15.937
2/18/2018 2:27	15.937	16.062	15.937
2/18/2018 2:37	15.937	16.000	15.937
2/18/2018 2:47	15.937	16.000	15.937
2/18/2018 2:57	15.937	16.000	15.937
2/18/2018 3:07	15.937	16.000	15.937
2/18/2018 3:17	15.937	16.000	15.875
2/18/2018 3:27	15.937	15.937	15.875
2/18/2018 3:37	15.875	15.937	15.875
2/18/2018 3:47	15.875	15.937	15.875
2/18/2018 3:57	15.875	15.937	15.875
2/18/2018 4:07	15.875	15.937	15.875
2/18/2018 4:18	15.875	15.937	15.875
2/18/2018 4:28	15.875	15.937	15.812
2/18/2018 4:38	15.875	15.875	15.875
2/18/2018 4:48	15.812	15.875	15.875
2/18/2018 4:58	15.812	15.875	15.812
2/18/2018 5:08	15.812	15.875	15.812
2/18/2018 5:18	15.812	15.812	15.750
2/18/2018 5:28	15.750	15.812	15.750
2/18/2018 5:38	15.750	15.812	15.687
2/18/2018 5:48	15.750	15.812	15.812
2/18/2018 5:58	15.750	15.812	15.750
2/18/2018 6:09	15.750	15.812	15.687
2/18/2018 6:19	15.750	15.750	15.687
2/18/2018 6:29	15.750	15.750	15.687
2/18/2018 6:39	15.687	15.750	15.625
2/18/2018 6:49	15.687	15.750	15.625
2/18/2018 6:59	15.687	15.750	15.625
2/18/2018 7:09	15.687	15.750	15.625
2/18/2018 7:19	15.687	15.750	15.625
2/18/2018 7:29	15.687	15.750	15.625
2/18/2018 7:39	15.687	15.687	15.625
2/18/2018 7:49	15.687	15.687	15.625
2/18/2018 8:00	15.687	15.687	15.625
2/18/2018 8:10	15.687	15.687	15.625

2/18/2018 8:20	15.687	15.687	15.625
2/18/2018 8:30	15.687	15.687	15.625
2/18/2018 8:40	15.687	15.687	15.625
2/18/2018 8:50	15.687	15.625	15.625
2/18/2018 9:00	15.687	15.687	15.625
2/18/2018 9:10	15.687	15.687	15.625
2/18/2018 9:20	15.750	15.687	15.625
2/18/2018 9:30	15.750	15.625	15.625
2/18/2018 9:40	15.750	15.625	15.625
2/18/2018 9:51	15.750	15.625	15.562
2/18/2018 10:01	15.812	15.625	15.562
2/18/2018 10:11	15.750	15.687	15.625
2/18/2018 10:21	15.812	15.625	15.625
2/18/2018 10:31	15.875	15.687	15.625
2/18/2018 10:41	15.875	15.687	15.625
2/18/2018 10:51	15.875	15.687	15.625
2/18/2018 11:01	15.875	15.687	15.625
2/18/2018 11:11	15.875	15.687	15.625
2/18/2018 11:21	15.875	15.687	15.687
2/18/2018 11:31	15.875	15.750	15.687
2/18/2018 11:42	15.875	15.687	15.687
2/18/2018 11:52	15.875	15.687	15.687
2/18/2018 12:02	15.812	15.312	14.875
2/18/2018 12:12	12.812	13.750	13.000
2/18/2018 12:22	16.375	16.000	15.875
2/18/2018 12:32	16.500	16.000	15.875
2/18/2018 12:42	16.562	16.000	15.875
2/18/2018 12:52	16.625	16.062	15.937
2/18/2018 13:02	16.625	16.062	15.937
2/18/2018 13:12	16.625	16.125	15.937
2/18/2018 13:22	16.625	16.187	16.000

ANEXO V

Videos cuando se probó el prototipo en el tanque

<https://drive.google.com/open?id=1vrXO4tn7qnjroS7O5rz5X1jMJCqv9dgx>

<https://drive.google.com/open?id=1EdSxh6fE8vRbNN4OUWU46uADALicZ2Ky>

Videos cuando se probó el prototipo en la laguna de Colta

https://drive.google.com/open?id=1gxn4pIJyt0cTz_FEVTVf1kxqVd598Trp

<https://drive.google.com/open?id=1nTNIsB465aWP7VINpQibHzEMJfibsYch>

<https://drive.google.com/open?id=1CDicQd2nDLJXyzK8R5XlzPa8HV0x3bWw>

<https://drive.google.com/open?id=1AxL7pJnBEOsyL2WvNXwXpzn1Kg2Veh3f>

https://drive.google.com/open?id=1scex_Jg0TLeJMG48ZNXyEsurNfNlq9ud

<https://drive.google.com/open?id=1QYSrYm2QKyNqvgtvPW3fWEnrN5YsJ3ac>

ORDEN DE EMPASTADO