



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

“DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO DE
TELECONTROL DE SISTEMA DE CULTIVO HIDROPÓNICO
CON INTERFAZ WEB MEDIANTE USO DE HARDWARE Y
SOFTWARE LIBRE”

INFORME DE PROYECTO INTEGRADOR

Previa a la obtención del Título de:

INGENIERO EN TELEMÁTICA

JOSEPH JACKSON REYES MOREIRA

GUAYAQUIL – ECUADOR

AÑO: 2015

AGRADECIMIENTOS

Mis más sinceros agradecimientos a mis padres por todo el apoyo y afecto que he recibido a lo largo de mi vida, también a mis hermanos, mis sobrinos y familia más cercana de quienes siempre he recibido aliento y esperanza.

A mis amigos quienes su fortaleza en espíritu me han llenado de luz y verdad para poder dedicar todos mis conocimientos y experiencia al servicio de mi patria, de mi comunidad y de los más necesitados.

Un agradecimiento especial a los Padres Alberto Raad y César Piechestein quienes son pilares fundamentales de mi fe cristiana, de quienes sus enseñanzas y amistad sincera, han forjado en mí, un corazón más limpio y puro con el cual puedo llegar a jóvenes y niños con un mensaje de amor y esperanza.

DEDICATORIA

Laudato Si, mi Signore.

Por el amor grande y la belleza de su ser mi corazón palpita ante su altar, ante su creación y ante la bondad de la gente humilde que le alaba con su corazón.

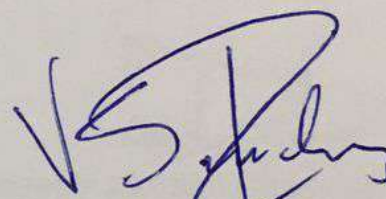
Muéveme, en fin, tu amor, y en tal manera, que aunque no hubiera cielo, yo te amara, y aunque no hubiera infierno te temiera. No me tienes que dar porque te quiera, pues aunque lo que espero no esperara, lo mismo que te quiero te quisiera.

TRIBUNAL DE EVALUACIÓN



Marcos Millán T., M.Sc.

PROFESOR EVALUADOR

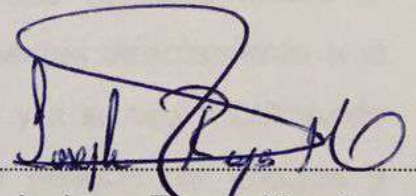


Vladimir Sánchez P., M.Sc.

PROFESOR EVALUADOR

DECLARACIÓN EXPRESA

"La responsabilidad y la autoría del contenido de este Trabajo de Titulación, me corresponde exclusivamente; y doy mi consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual"



Joseph Jackson Reyes Moreira

RESUMEN

Este proyecto se presenta como una solución sostenible para el desarrollo de la agricultura en zonas rurales y urbanas, donde se requiere de alimentos en mejor estado, que aporten más nutrientes para el consumo humano, los tiempos de cosecha sean menores y no se dependa de la tierra ni el clima como factores que pueden influir en el valor nutricional de estas plantas y a la producción.

Debido a la gran cantidad de químicos en nuestras tierras, el uso de plaguicidas o los cambios bruscos del clima, es necesario utilizar nuevas técnicas y tecnologías sostenibles como es el caso de la hidroponía, una técnica utilizada desde la antigüedad, que hace uso del agua como portador de nutrientes directamente a la raíz, adaptando fácilmente las necesidades de cada planta y a su vez reutilizando eficientemente el flujo del agua, además existen menos riesgos de hongos y parásitos que afectan a nuestros cultivos, puedes cultivar tus propios alimentos en tu hogar, corregir fácilmente la falta o el exceso de los valores nutricionales, obtener una mayor velocidad de crecimiento, reducir costos de producción y se puede realizar en zonas con cualquier tipo de clima.

Se diseñó e implemento un sistema automatizado de hidroponía, cuya estructura está basada en diseño NFT que nos permite cultivos horizontales y verticales con recirculación continua del flujo de agua.

El uso de hardware y software libre nos reduce considerablemente los costos de producción, el controlador BeagleBone Black es eficiente para el manejo de procesos y servicios necesarios en nuestro proyecto, ya que su reducido tamaño y potente procesador lo hacen destacar como una opción primaria de proyectos de automatización.

El sistema SMARSiS un sistema informático especialmente diseñado para obtener resultados basados en un análisis estadístico de los diferentes parámetros a medir, permitirá tomar decisiones y medidas correctivas en el menor tiempo posible traduciendo esto en reducción de costos, mantenimiento y optimización del producto.

La Hidroponía propone un cambio de mentalidad más humano, pues debemos de considerar que el planeta está constantemente cambiando y hay zonas donde llueve más y zonas con sequías, por lo que nosotros debemos de cuidar nuestros recursos naturales promoviendo proyectos ecológicos, sostenibles y sustentables, para dejar un mejor planeta para los que nos precederán a futuro.

ÍNDICE GENERAL

AGRADECIMIENTOS.....	ii
DEDICATORIA	iii
TRIBUNAL DE EVALUACIÓN	iv
DECLARACIÓN EXPRESA.....	v
RESUMEN	vi
ÍNDICE GENERAL.....	viii
CAPÍTULO 1	1
1. DESCRIPCIÓN DEL PROBLEMA.....	1
1.1 Objetivos	1
1.2 Antecedentes	1
1.3 Justificación	3
CAPÍTULO 2.....	5
2. DISEÑO DEL SISTEMA HIDROPÓNICO	5
2.1 Modelos del sistema	5
2.2 Descripción de procesos	9
2.3 Controlador, sensores y actuadores	11
2.4 Plataforma del sistema hidropónico	16
CAPÍTULO 3.....	22
3. IMPLEMENTACIÓN DEL SISTEMA HIDROPÓNICO	22
3.1 Prototipo del sistema hidropónico NFT	22
3.2 Procesos y algoritmos	32
3.3 Sistemas de monitoreo, análisis y control	37
CAPÍTULO 4.....	49
4. PRUEBAS DEL SISTEMA Y ANÁLISIS DE RESULTADOS	50
4.1 Toma de lecturas en tiempo real	50
4.2 Control de actuadores y configuración remota	55
4.3 Análisis estadístico de los parámetros medibles obtenidos	56
4.4 Resultados de rendimiento del controlador	69

CONCLUSIONES Y RECOMENDACIONES	77
BIBLIOGRAFÍA	79
ANEXOS	80

CAPÍTULO 1

1. DESCRIPCIÓN DEL PROBLEMA

1.1 Objetivos

El proyecto “diseño e implementación de un prototipo de telecontrol de sistema de cultivo hidropónico con interfaz web mediante uso de hardware y software libre”, tiene como objetivo beneficiar a las zonas agrícolas y zonas urbanas ecológicas, de manera que las personas puedan cultivar alimentos sin depender de las condiciones de la tierra y el clima, por consecuente obtener productos más saludables y en menor tiempo. Al ser un producto cultivado en condiciones controladas donde solo se hace uso del agua como recurso de nutrientes para las plantas, estas se verán menos afectadas por enfermedades o plagas, brindando un producto de mayor calidad para el consumo humano y ayudando a combatir la desnutrición.

Este sistema hace uso eficiente del recurso del agua de manera que se puede reutilizar en ciclos cerrados y volviéndola a purificar a través de filtros naturales.

1.2 Antecedentes

Actualmente el uso de plaguicidas y otros químicos afectan los cultivos, el suelo, el agua y la salud de las personas que consumimos estos productos. Las condiciones climáticas y otros fenómenos naturales no permiten el correcto desarrollo de los cultivos produciendo pérdidas económicas y en la población una mayor desnutrición por falta de estos productos primarios.

Es por eso que ahora se requiere de técnicas más sustentables y ecológicas que no solo nos provea de un producto de mayor calidad sino que sea amigable con el ambiente.

La hidroponía también conocida como el arte de cultivar sin tierra es una técnica milenaria que está tomando mucha fuerza en estos tiempos, cuando esta se combina con las tecnologías de la información de manera que se tenga un control automatizado permitiéndonos ahorrar recursos y tiempo. Se cree que los jardines colgantes de Babilonia pudieron utilizar esta técnica hidropónica.

El siguiente gran desarrollo del cultivo hidropónico se da por el siglo XVII en Europa con el uso de invernaderos para cultivar plantas de menta sin necesidad de suelo, fueron los alemanes quienes desarrollaron las técnicas de laboratorio [1].

Los EE.UU empezaron a investigar y experimentar aplicaciones hidropónicas durante la II Guerra Mundial, para proveer de suministros alimenticios a los militares en las islas del Pacífico donde por su zona geográfica y suelo no era posible cultivar. Los sistemas hidropónicos al inicio eran demasiado costosos de implementar, sin embargo a finales de los años 60, se recobró la atención en esta técnica de nuevo al verse reducido los costos de implementación gracias a los avances del plástico.

Hoy en día, el uso de la hidroponía se ha popularizado no solo como una forma de producir alimentos en mayor escala, sino alimentos más saludables y sabrosos [1]. También ayuda a que en las zonas urbanas se promueva los huertos ecológicos ya sea como pasatiempo, compartir experiencias en familia o incluso para ayudar centros de protección infantil y adultos mayores brindándoles un alimento con todos los nutrientes necesarios.

La electrónica y la computación han evolucionado también de manera que las válvulas de vacío y los viejos transistores son reemplazados por pequeños y más poderosos micro procesadores, esto es debido a la micro miniaturización de los componentes y la investigación de nuevos materiales.

Así mismo en la computación veíamos enormes y pesados equipos con poco poder de procesamiento y de enorme costos que solo se encontraban en centros de investigación, ahora es tan simple como ver dispositivos móviles inteligentes o las SBC que en los últimos años están ganando mucho terreno debido a su alto poder de procesamiento y bajo costo para aplicaciones sencillas.

Para que un controlador cumpla con el formato SBC necesita básicamente que todos sus componentes estén integrados en la placa base y que solo requiera de ellos salvo alguna excepción, solo los periféricos de entrada y salida, a la vez que pueda caber en la palma de la mano.

Muchos de estos dispositivos se pueden configurar y programar en diferentes lenguajes de programación tanto de bajo nivel como (ASM, VHDL y MIPS) como de alto nivel (C/C++/C#, Java, Python, Ruby, PHP, JavaScript, Node.js, Octave, etc.). Muchos de estos dispositivos no están adecuados para entornos industriales duros por las condiciones ambientales como el ruido de los equipos, pero a pesar de ello son muy utilizadas para automatizar diversas aplicaciones de forma profesional y a bajo costo.

Las comunicaciones y los sistemas de información se han integrados para crear servicios en la nube que pueden almacenar nuestros datos de manera segura y accesible en cualquier lugar del mundo. Entre estas tecnologías encontramos las redes de sensores inalámbricas, que han evolucionado de las redes convencionales pero con la diferencia de que cada nodo actuará de forma autónoma según el protocolo que implemente, otra característica importante es que su reducido tamaño y costo de manera que se puede fácilmente tener una red con cientos de nodos disponibles para aplicaciones de telemetría y telecontrol.

Por último, hace unas décadas atrás las aplicaciones y servicios web tenían contenido estático y estaban desarrollados en un modelo de una sola capa, lo que hacía del aplicativo mono funcional y de difícil mantenimiento, otra desventaja que se veía era la incompatibilidad entre sistemas debido a la falta de estándares. Ahora los sistemas se integran para formar aplicaciones de n-capas, portables y flexibles, con la aparición de nuevos lenguajes de programación y entornos de desarrollos web, los tiempos de desarrollo disminuyen, a la vez que se crean nuevos servicios web como es el caso del internet de las cosas, que integra todos los dispositivos del hogar en una sola red brindando numerosos beneficios al usuario.

1.3 Justificación

A nivel global, se está viviendo un cambio brusco en las condiciones climáticas debido a la influencia del hombre en las últimas décadas produciendo el efecto invernadero. Esto causa que en algunas zonas del mundo llueva más de lo normal o se produzcan severas sequías. Es una cuestión crítica a considerar en los años venideros pues el alimento podrá escasear y debido al mal uso de

nuestros suelos y el desgaste del agua, nos veremos obligados a consumir productos de mala calidad y de bajo contenido nutricional.

Debemos tomar conciencia que hay que tener un ambiente más verde, que es bueno tener edificios pero, no hay que destruir algo tan bello que es la naturaleza para construir otra belleza que la suplante como un edificio, hay técnica y técnicas que pueden utilizarse en conjunto de manera que ambas puedan convivir y así dar un mayor bienestar a la humanidad [2].

El cuidado de la tierra es de responsabilidad de cada uno de nosotros, no solo por el hecho de que en ella encontramos refugio, alimentos y grandes bellezas, sino por el futuro de nuestras generaciones que sufrirán el dolor de las consecuencias y malas decisiones que tomamos hoy.

Para solventar este problema se plantea una posible solución en el cual se diseñará e implementará un controlador que será el encargado de coordinar los diversos procesos del sistema, para esto se hará uso de tecnología de bajo consumo energético y costo.

El sistema cuenta con un reservorio de agua en el cual se medirán los parámetros de la calidad del agua, estos datos serán administrados por un servicio web desarrollado en tecnología java y software de código libre que permitirá visualizar y analizar en tiempo real la información. En el futuro, la tendencia sugiere que la mayoría de infraestructuras de TI empezarán a migrar a la nube.

El sistema hidropónico hará uso de materiales reciclables y de bajo costo para la construcción de la plataforma del prototipo, de manera que el costo final sea compensado por el tiempo de retorno de la inversión inicial para el montaje del proyecto.

Toda la información recopilada por el servicio web servirá para que el agricultor o analista pueda tomar medidas correctivas eficientemente y así obtener productos de alta calidad y disminuir costes en la producción de los alimentos.

CAPÍTULO 2

2. DISEÑO DEL SISTEMA HIDROPÓNICO

2.1 Modelos del sistema

El sistema está diseñado de tal manera que permite adaptarse fácilmente a las necesidades del usuario, para lo cual se propone el siguiente modelo del controlador como podemos ver en la Figura 2.1.

Este modelo cuenta con los siguientes bloques que serán los responsables de realizar las tareas o procesos que requiera el sistema para su correcto funcionamiento.

Inicialización del sistema: permite inicializar las variables de entorno del sistema, ejecutar tareas, ejecutar servicios y sincronizarse con la red IP.

Programador de tareas: permite crear un servicio de programación de tareas que servirán para la ejecución de eventos, alarmas, periodos de toma de medición, etc.

Servidor web: permite cargar un sitio o aplicativo web que servirá como interfaz hombre máquina con el controlador. Este servidor deberá escuchar peticiones a un puerto común, brindar las debidas seguridades e integridad de la información y manejar correctamente los recursos de conexión a la base de datos.

Base de datos: permite manejar y almacenar la información en un esquema de datos relacionados. Esta base de datos deberá estar normalizada y relacionada correctamente para que la lógica de negocios pueda ser flexible e independiente.

Servidor NTP: permite la sincronización del tiempo del controlador a través de un servicio que se conecta a internet.

Selector de tareas: permite ejecutar una o varias tareas dependiendo de las condiciones y eventos que ocurran en el sistema. Estas tareas pueden ser ejecutadas manualmente o automáticamente.

Sensores: permiten la toma de lecturas de parámetros físicos o químicos a través de transductores, esta información luego será amplificada, acondicionada y digitalizada para su posterior almacenamiento.

Actuadores: permiten ejecutar dispositivos electromecánicos o electrónicos a través de señales eléctricas como motores, relés, válvulas eléctricas, etc.

Requerimiento web: permite realizar peticiones desde un cliente web remoto al controlador de manera que este pueda proveer de la información necesaria o ejecutar una tarea específica.

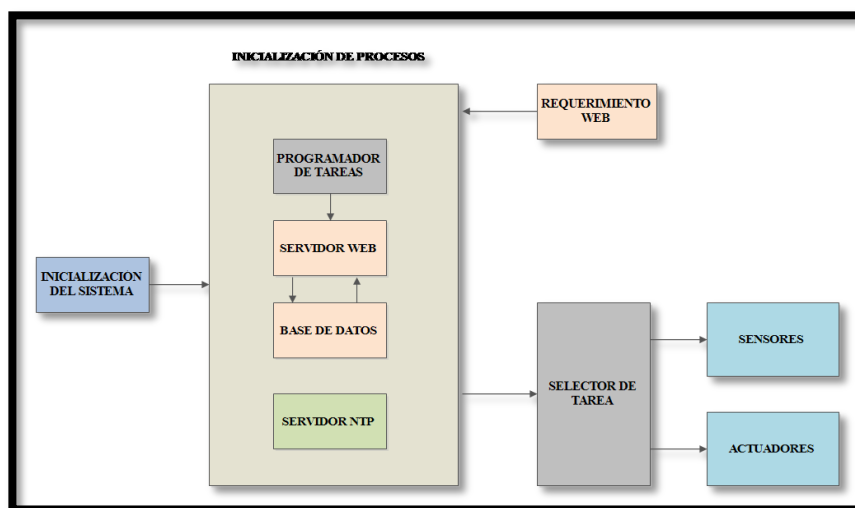


Figura 2.1: Modelo del controlador.

El esquema de la base de datos está diseñado de tal manera que el sistema pueda ser lo más dinámico posible, permitiendo crear módulos, nodos, sensores, actuadores, usuarios y roles. Como podemos ver en la Figura 2.2. Este esquema deberá estar normalizado y relacionado correctamente, contar con las debidas seguridades y ser independiente del lenguaje de programación para que cualquier aplicativo pueda consumir o realizar transacciones con la información.

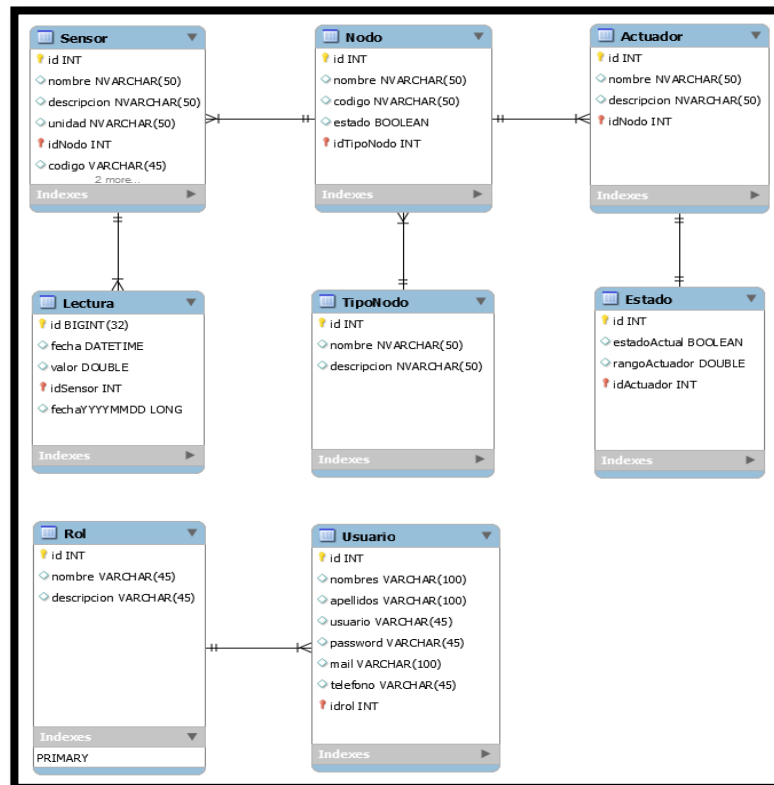


Figura 2.2: Modelo de la base de datos.

Se diseñará un sistema informático que permita monitorear en tiempo real, analizar datos mediante estadística, controlar dispositivos remotamente y administrar usuarios. Para lo cual este sistema deberá estar implementado bajo la arquitectura MVC de manera que todas las capas sean independientes entre sí. Ver Figura 2.3.

Se crearán cuatro módulos que están asociados a la capa de presentación que son:

Módulo de resumen: este módulo permitirá visualizar las lecturas tomadas de los sensores en tiempo real presentando la información básica de cada parámetro.

Módulo de análisis: este módulo permitirá analizar estadísticamente la información de manera descriptiva e inferencial para que el analista pueda tomar decisiones oportunamente.

Módulo de control: permitirá que el usuario pueda interactuar remotamente con algún dispositivo electromecánico o electrónico, deberá almacenar el último estado del dispositivo.

En la lógica de negocios se implementará dos módulos importantes que servirán de interfaz entre el sistema SMARTSiS y el controlador los cuales son:

Módulo de servicios web: este módulo permitirá crear servicios para que cualquier dispositivo pueda ejecutar un procedimiento remotamente, en este caso almacenamiento de información en la base de datos. Deberá contar con las debidas seguridades y realizar las peticiones mediante POST.

Modulo middleware: es una aplicación que sirve como intermediario entra dos sistemas, para lo cual se implementará como interfaz de ejecución de comandos shell script remotamente. Se deberá brindar los debidos permisos de ejecución, lectura y escritura a los archivos con los que el usuario interactuará remotamente.

Por último se tiene el módulo de base de datos el cual está relacionado a la capa de datos que ejecutará todas las transacciones y lecturas según los requerimientos del usuario. El sistema se conectará a la base de datos mediante datasources para poder manejar de manera eficiente todas las conexiones a la base de datos y así no saturar al sistema.

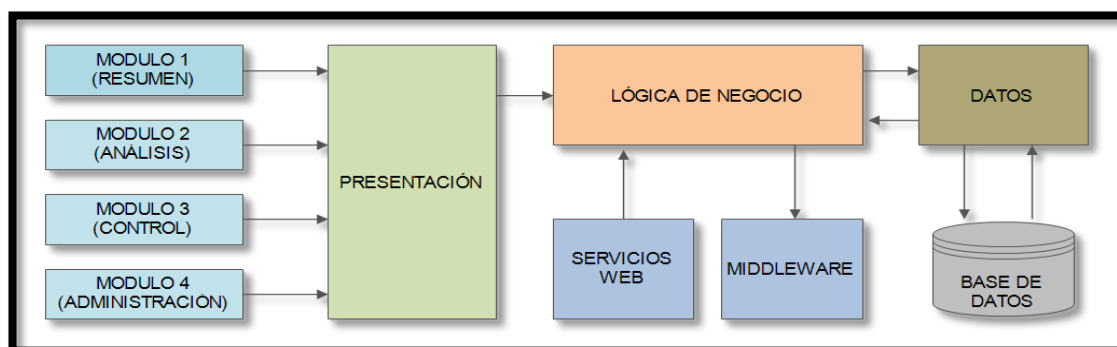


Figura 2.3: Modelo del sistema de SMARTSiS.

2.2 Descripción de procesos

El sistema hidropónico cuenta con un controlador principal el cual deberá coordinar todos los procesos y tareas que se requieran. Para esto se han dividido los procesos en las siguientes categorías:

Proceso de adquisición de datos: proceso por el cual el controlador tomará lecturas de los dispositivos de medición ya sea a través de I2C, UART, SPI, ADC, OneWire, etc. Este proceso deberá hacer abstracción de la capa de hardware de manera que pueda ser parametrizado para cualquier modo de comunicación antes mencionado. Ver características de estos protocolos en la Tabla 1.

Comunicación	Tasa	Distancia	Hardware	Escalabilidad
UART	20kbps	15m	2 pines	baja
SPI	25Mbps	0.1m	4+ pines	media
I2C	1Mbps	0.5m	2 pines	alta
OneWire	110kbps	5m	1 pin	alta

Tabla 1: Características principales de protocolos de comunicación para periféricos

Proceso para actuadores: proceso por el cual el controlador activa o desactiva un dispositivo electromecánico o electrónico ya sea manual o automáticamente para esto hace uso de una capa de abstracción de hardware, en el cual le indicamos si un pin deberá estar encendido o apagado.

Proceso de programación de tareas: proceso que sirve para automatizar tareas, de manera que puedan ejecutarse en periodos programados de tiempo, bajo eventos u alarmas. Este proceso deberá poder administrar en cualquier momento todas las tareas relacionadas a un usuario del sistema.

Proceso inicializador del sistema: proceso que permitirá pre configurar el sistema levantando servicios, sincronizando módulos, configurando variables de entorno, etc. En la Figura 2.4 podemos observar el bloque de procesos que administra el controlador. Este proceso deberá validar todos los periféricos de entradas y salidas que estén correctamente funcionales y listo para ser utilizados.



Figura 2.4: Proceso inicializador del sistema.

Proceso servicios web: proceso que integra el sistema SMARTSiS para brindar facilidad de interactuar con el controlador independientemente del lenguaje de programación que se use. Estos servicios deberán brindar toda la seguridad respectiva con clave de seguridad, códigos y tiempos máximo de respuesta. Ver Figura 2.5 el diagrama de bloques del servicio RESTFul implementado.

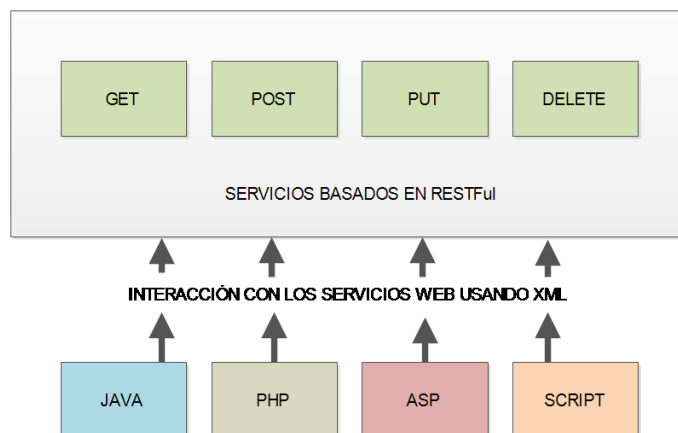


Figura 2.5: Servicios web RESTful.

2.3 Controlador, sensores y actuadores

Como controlador principal del sistema se utilizará la tarjeta BeagleBone Black que es una SBC, el diseño de esta tarjeta se centra en un microprocesador, memorias RAM, periféricos de entradas y salidas, módulos de comunicación, etc. todo esto integrado en la misma placa electrónica. Ver Figura 2.6.

El sistema operativo que corre es Linux, una versión especialmente diseñada para este tipo de microcomputador en el cual podremos programar todos los procesos, tareas y servicios que se necesiten.

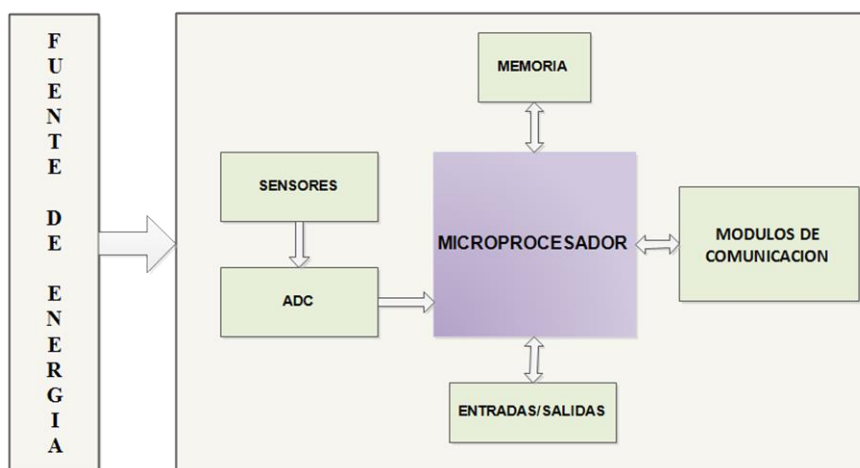


Figura 2.4: Arquitectura general del controlador.

En la Tabla 2 podremos observar las características de este micro computador el cual brinda numerosas aplicaciones que requieran de hardware de bajo costo y consumo de energía.

Características		
Procesador	Sitara AM3358BZCZ100 1Ghz, 2000 MIPS	
Motor Gráfico	SGX530 3D 1280x1024 máximo HDMI, 1024x768 a 1920x1080@24Hz	
Memoria SDRAM	512MB DDR3L 800Mhz	
EMMC	4Gb, 8bit embebido	
Fuente de alimentación	Mini USB o DC Jack	5VDC pines de expansión
PCB	3.4" x 2.1"	6 capas
Puertos Seriales	6 puertos UART0 a UART5	
Puerto Ethernet	10/100 RJ45	
SD/MMC conector	microSD 3.3V	
Pulsadores	Botón para encender, reiniciar e inicializar arranque	
Puertos	SPI, I2C, GPIO(69 pines), LCD, GPMC, MMC1, MMC2, 7 entradas análogas, 4 temporizadores, CAN	
Peso	39.68 gramos	
Audio	Interface Via HDMI, Stereo	

Tabla 2: Característica de la BeagleBone Black Rev. C.

El sistema hidropónico deberá contar con los siguientes sensores para poder funcionar correctamente, los dividiremos en dos grupos: sensores para la calidad del agua y sensores ambientales.

Sensores de calidad del agua

PH: es una medida de acidez o alcalinidad de una disolución, que nos indica la concentración de iones de hidrogeno (H^+) tiene un rango entre 0 y 14, cuando más bajo el PH es una sustancia acida, cuando más alto el pH es una sustancia básica [3]. Ver en la Figura 2.5 la escala del PH.



Figura 2.5: Escala medible del pH.

Conductividad eléctrica: la conductividad eléctrica es una medida que permite estimar la concentración de sales disueltas en el agua, lo que permite conducir corriente a través del agua [3]. Cuanto más nutrientes contiene la solución, más conduce la electricidad, esto puede provocar que las plantas se deshidraten por exceso de sales. Se mide en microSiemens por centímetro ($\mu S/cm$).

Sólidos totales disueltos: es una medida de cuantificar los sólidos totales disueltos en una solución cloruro sódica $NaCl$ o sal común, donde su factor de conversión es $1 \text{ microS/cm} = 0.5 \text{ ppm}$, es decir 500 ppm por cada 1 mS/cm . Ver en la Figura 2.6 como TDS está relacionado a la conductividad eléctrica.

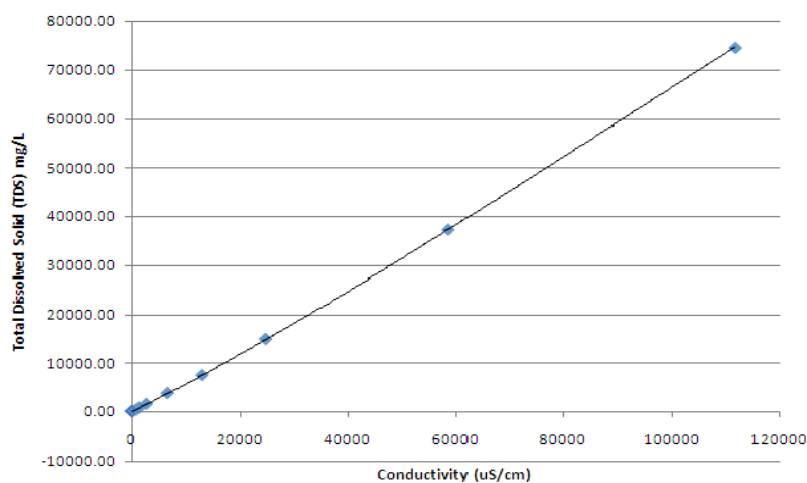


Figura 2.6: Escala medible TDS vs EC

Oxígeno disuelto: Es la cantidad de oxígeno que esta disuelta en el agua, nos permite medir el nivel de contaminación del agua. A un nivel más alto de oxígeno disuelto el agua es de mejor calidad, si los niveles de oxígeno disuelto son muy bajos los peces y otros organismos no podrán sobrevivir [3]. Parte del oxígeno disuelto en el agua es el resultado de la fotosíntesis de las plantas acuáticas. La conductividad eléctrica, la temperatura y la altitud pueden afectar la medición del oxígeno disuelto tal como vemos en la Figura 2.7. Se expresa en porcentaje de saturación y pueden variar entre 7 a 12 ppm o mg/l. la mínima cantidad entre 4 a 5 ppm.

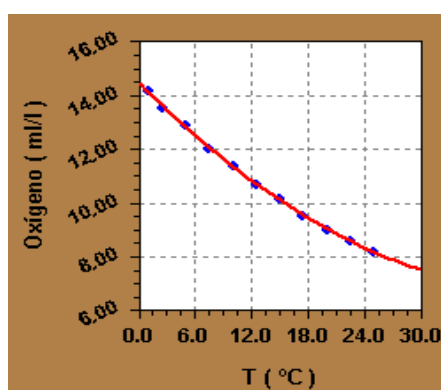


Figura 2.7: Escala medible D.O vs Temperatura.

ORP: potencial de reducción u oxidación, también conocido como REDOX, nos permite medir el potencial eléctrico en mV del agua, lo que nos indica la presencia de agentes de oxidación o reducción de su carga eléctrica específica [4]. En la Figura 2.8 observamos como el ORP varía en función del PH.

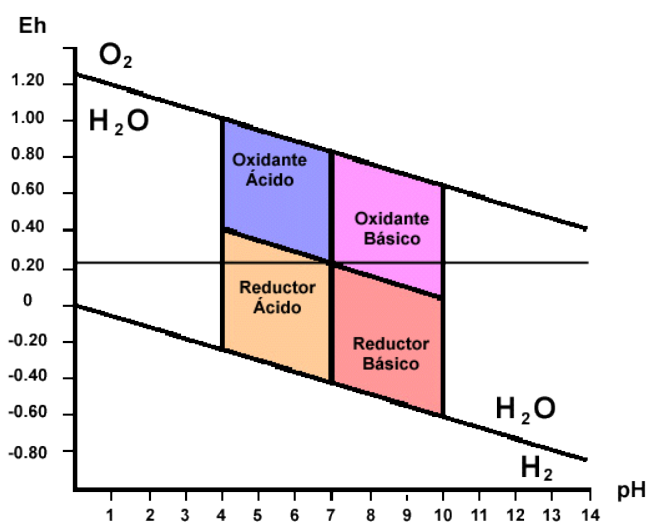


Figura 2.8: Escala medible ORP vs PH.

Temperatura del agua: este parámetro nos permite mantener el agua en un estado donde la vida orgánica sea sustentable lo mediremos en grados Celsius $^{\circ}C$.

Sensores ambientales

Intensidad solar relativa: este parámetro nos permite medir la cantidad de luz que incidirá sobre la vida orgánica. La mediremos en porcentaje del 0 a 100%, es decir de la ausencia de luz a la máxima intensidad que soporta el transductor.

Temperatura ambiental: este parámetro nos permitirá medir la temperatura ambiente, la mediremos en grados Celsius $^{\circ}C$.

Humedad relativa: este parámetro nos estimar en porcentaje la cantidad de vapor de agua o algún líquido presente en el medio ambiente.

Para poder controlar los diferentes dispositivos del sistema utilizaremos los actuadores, estos nos permitirán encender o apagar, controlar velocidades de motores, intensidad de luz, fuerza aplicada, etc.

Tipo de actuador	Ventajas	Desventajas
Neumático	<ul style="list-style-type: none"> • Bajo costo • Rapidez • Sencillos • Robustos 	<ul style="list-style-type: none"> • Requieren de instalaciones especiales • Ruidosos
Hidráulico	<ul style="list-style-type: none"> • Rápidos • Alta capacidad de carga • Presentan estabilidad frente a cargas estáticas 	<ul style="list-style-type: none"> • Requieren instalaciones especiales • Son de difícil mantenimiento • Resultan poco económicos
Eléctrico	<ul style="list-style-type: none"> • Precisos y fiables • Silenciosos • Su control es sencillo • Son de una fácil instalación 	<ul style="list-style-type: none"> • Potencia limitada

Tabla 3: Tabla de ventajas y desventajas de los actuadores

2.4 Plataforma del sistema hidropónico

Existen muchos tipos de sistemas hidropónicos para este proyecto se utilizará el sistema NFT, este sistema permite llevar la solución nutritiva a través de

tuberías, donde las plantas estarán colocadas en pequeñas perforaciones y separadas entre sí a una cierta distancia para que la raíz pueda mantenerse siempre húmeda y oxigenada en un flujo constante y re circulatorio de agua. Este sistema requiere de una bomba y permite cultivar de manera vertical y horizontal. En la Figura 2.9 podemos ver el modelo 3D de la plataforma del sistema hidropónico.

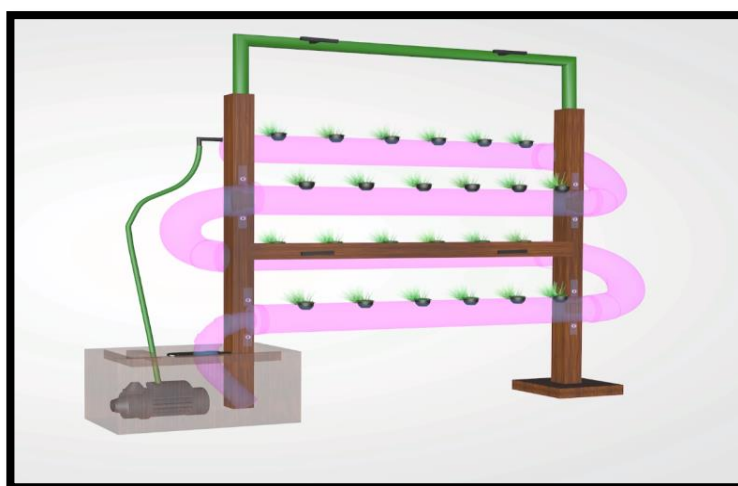


Figura 2.9: Modelo 3D del sistema hidropónico NFT.

Para el diseño de este sistema se deberá tomar en cuentas las siguientes características:

Para los marcos que sostendrán la tubería deberán estar separados al menos un metro, tener una base firme en cada lado para evitar que las vibraciones o el peso puedan hacer que se balancee y dañe la estructura y las plantas.

La tubería que utilizaremos es de PVC de las cuales cortaremos cuatro pedazos de un metro cada uno en cada esquina, para unir a los tubos se utilizará codos de PVC los cuales deberán estar correctamente sellado para evitar fuga de agua.

En cada tubería haremos huecos circulares de diez centímetros a una distancia de quince centímetros entre sí, que servirán de apoyo para las plántulas que cultivaremos de manera que tengan su espacio suficiente para desarrollarse [5].

Se requerirán de una bomba de agua que succionara el agua de unos de los reservorios hasta la parte superior de la estructura, donde el agua circulará entre las tuberías hasta llegar hasta otro reservorio donde ser filtrará y pasará al primer reservorio donde el ciclo comenzara de nuevo.

Este sistema deberá de contar con aspersores para mantener húmeda las hojas de las plántulas cada cierto tiempo dependiendo de la especie [6].

Todo este sistema será controlado por la BeagleBone Black que mantendrá la calidad del agua de manera que sustente la vida y los actuadores que deberán controlar la bomba de agua y el sistema de riego.

En el cultivo hidropónico intervienen varios factores que pueden influir en el correcto funcionamiento del sistema de los cuales tenemos los siguientes:

La luz es fundamental para producir clorofila e hidratos en las plantas, en ausencia de la luz el follaje se vuelve amarillento y la planta muere. Hay que tener en consideración que la planta debe recibir todo el espectro de colores de la luz, para que sus procesos biológicos funcionen correctamente [7].

Por lo general una planta debe estar sometida a la luz por al menos 12 horas, dependiendo de la planta a cultivar. Para obtener un mayor periodo de crecimiento se deberá de proveer luz a la planta por un mayor tiempo.

Si se utiliza sistemas de iluminación artificial hay que tener en consideración la distancia conveniente a cada planta, pues si no llega luz suficiente estas crecerán en la dirección donde puedan obtener una mayor cantidad de luz en vez de crecer de forma homogénea, y si está demasiado cerca puede quemar las partes más cercanas de la planta.

La luz está relacionada directamente con la temperatura del medio ambiente, al tener más horas de luz la temperatura aumentará provocando deshidratación severa o quemaduras en las plantas, también acelerara los procesos biológicos disminuyendo la energía disponible de la planta lo cual implicara frutos más pequeños. Caso contrario al tener menos luz la temperatura disminuirá produciendo que el agua se congele y no lleguen los suficientes nutrientes para

el correcto desarrollo, también los tejidos celulares de las plantas se cristalizan afectando y dañando a las raíces, hojas, tronco o frutos [8].

La temperatura óptima para el desarrollo de las plantas durante el día está entre 22 a 24 ° C, y en las noches entre 15 a 18° C. Otro aspecto importante es el control de la temperatura de la solución nutritiva que debe mantenerse entre 13 y 15°C, ya que al verse afectada por la temperatura se congelará cerca de los 0°C y a temperaturas mayores a 30°C la solubilidad del oxígeno se reduce.

La humedad es un parámetro importante que ayuda a compensar los cambios bruscos de temperatura, por lo general se recomienda mantener al sistema entre 50 y 70% de humedad relativa. Otro punto, sea cualquier tipo de sistema hidropónico a implementar la humedad de la raíz debe mantenerse siempre húmeda para que pueda absorber adecuadamente los nutrientes que contiene el agua [9].

La calidad del agua es un recurso muy importante que debemos mantener con los parámetros estándares de calidad que los sistemas hidropónicos requieren, ya que son los encargados de llevar todos los nutrientes a las raíces de las plantas.

La solución nutritiva debe aportar todos los nutrientes necesarios para la germinación, desarrollo, floración y fructificación de la planta. Para que las raíces puedan absorber correctamente los nutrientes, estos deben estar disueltos en el agua, es por eso que se debe mantener los parámetros de pH, conductividad eléctrica, oxígeno disuelto, temperatura y ORP en los rangos indicados para cada sistema hidropónico.

Los componentes de los que está formado la solución nutritiva y que son necesarios para un correcto desarrollo de las plantas son: Azufre, Boro, Calcio, Carbono, Fósforo, Hidrógeno, Hierro, Magnesio, Manganeso, Nitrógeno, Oxígeno, Potasio y Zinc. En la Tabla 4 podemos observar los componentes principales de una solución nutritiva para hidroponía.

Elemento	Formas Iónicas Absorbidas por la planta	Rango Común (ppm = mg/l)
Nitrógeno	Nitrato (NO ₃ -), Amonio (NH ₄ +)	100-250
Fósforo	H ₂ PO ₄ -, PO ₄ ³⁻ , HPO ₄ ²⁻	30-50
Potasio	Potasio (K ⁺)	100-300
Calcio	Calcium (Ca ²⁺)	80-140
Magnesio	Magnesio (Mg ²⁺)	30-70
Azufre	Sulfato (SO ₄ ²⁻)	50-120
Hierro	Fe ²⁺ , Fe ³⁺	1.0-3.0
Cobre	Cobre (Cu ²⁺)	0.08-0.2
Manganeso	Manganeso (Mn ²⁺)	0.5-1.0
Zinc	Zinc (Zn ²⁺)	0.3-0.6
Molibdeno	Molybdato (MoO ₄ ²⁻)	0.04-0.08
Boro	BO ₃ ²⁻ , B ₄ O ₇ ²⁻	0.2-0.5
Cloruro	Cloro (Cl ⁻)	<75
Sodio		<50

Tabla 4: Tabla de los componentes de la solución nutritiva.

El agua es el recurso principal en el sistema hidropónico en el cual estarán disueltos los nutrientes que absorberá la raíz. Si no se tiene un adecuado control de las concentraciones se verán afectados los parámetros del agua como son la conductividad eléctrica, el pH, los sólidos totales disueltos, ORP y el oxígeno disuelto.

El PH es un parámetro importante que se debe medir constantemente ya que se ve afectado por las raíces que crecen en el medio y que alteran a la solución nutritiva, se debe mantener entre 5.5 y 7.0 en un sistema hidropónico.

Un pH demasiado bajo puede ocasionar riego de toxicidad en los micronutrientes, así como problemas de disponibilidad de calcio y magnesio. Un pH demasiado alto puede ser un indicador de carencia de hierro y disponibilidad de nutrientes ya que son más difícil de disolver por la planta [8].

La conductividad eléctrica de la solución varía con la temperatura por lo que se debe de compensar al momento de tomar una medición. Para mantener la disponibilidad de los nutrientes se recomienda mantenerla en el rango de 1.5 a 3 mS/cm o 750 a 1500 ppm. Los cultivos hidropónicos emplean soluciones entre 500 a 2000 ppm, siendo los valores más seguros entre 800 a 1200 ppm.

La cantidad de oxígeno que esta disuelta en el agua, nos permite medir el nivel de contaminación del agua. A un nivel más alto de oxígeno disuelto el agua es de mejor calidad, si los niveles de oxígeno disuelto son muy bajos los peces y otros organismos no podrán sobrevivir.

Parte del oxígeno disuelto en el agua es el resultado de la fotosíntesis de las plantas acuáticas. La conductividad eléctrica, la temperatura y la altitud pueden afectar la medición del oxígeno disuelto. Se expresa en porcentaje de saturación y pueden variar entre 7 a 12 ppm o mg/l. la mínima cantidad entre 4 a 5 ppm [9].

El agua con un nivel alto de pH tiene más agentes que reducen y el agua con niveles bajos en PH tiene agentes que oxidan. El agua idealmente inactiva viralmente esta entre 650 a 750mV lo que nos indica que es acta para el consumo humano. Con un valor adecuado de ORP algunas bacterias como la E. Coli, la salmonela, la Listeria o el virus de la polio y otros organismo patógenos pueden vivir tan solo unos 30 segundos, por lo que es importante monitorizar el nivel de ORP.

CAPÍTULO 3

3. IMPLEMENTACIÓN DEL SISTEMA HIDROPÓNICO

3.1 Prototipo del sistema hidropónico NFT

El prototipo que se desarrolló está basado en diseño hidropónico NFT el cual está compuesto de dos reservorios de plástico con agua, el primer reservorio tendrá peces que servirán como productores de nutrientes para las plantas a través de sus desechos. El segundo reservorio servirá como depósito de sólidos suspendidos de gran tamaño, el agua pasará posteriormente al primer reservorio para cerrar el ciclo continuo del agua. En la Figura 3.1 podemos observar el prototipo NFT hidropónico implementado con un acho de 1 metro y de largo 1.5 metros.



Figura 3.1: Plataforma del sistema hidropónico.

Se cuenta con una bomba de agua de $\frac{1}{2}$ HP que trabaja a 110V, consume una corriente de 3.5 A y tiene un caudal de 35 litros por minuto. Esta bomba

permitirá succionar el agua del segundo reservorio a través de la tubería de PVC de $\frac{3}{4}$ de pulgada para que pueda ingresar al sistema por la parte superior. Ver Figura 3.2



Figura 3.2: Bomba de agua de $\frac{1}{2}$ HP.

Las tuberías deberán estar inclinadas en un ángulo entre 5 a 10 grados con respecto a la horizontal de manera que por acción de gravedad el agua circule a través de ellos hasta llegar al segundo reservorio.

Los orificios de las tuberías tienen un diámetro de 8 cm se los puede realizar con herramientas circulares que se adaptan al taladro de mano, se debe procurar dejar una distancia entre orificios de 10 a 15 cm para que las raíces no vayan a estar en contacto una con otra.

Para el sistema de riego se usó una electroválvula de $\frac{1}{2}$ de pulgada a 12V DC con una presión máxima de 116 psi y hasta 100 °C de temperatura que permitirá rociar agua a través de los aspersores de $\frac{1}{4}$ de pulgada a las hojas de las plántulas, para que estas no se sequen por el calor, el tiempo de riego depende de la especie que se cultivara pudiendo ser una sola vez al día o dos veces por semana. En la Figura 3.3 podemos ver el sistema de riego que utilizaremos.



Figura 3.3: Electroválvula de ½ pulgada, manguera plástica de ½ de pulgada y aspersores de ¼ de pulgada.

Para el controlador se usó la tarjeta BeagleBone Black Rev. C con sistema operativo Linux distribución debían. Ver el controlador en la Figura 3.4.

Esta placa es un microcomputador que cuenta con todas las características necesarias básicas para poder ejecutar nuestro proyecto correctamente. Para lo cual haremos uso de los puertos seriales, puertos GPIO, puertos ADC, Ethernet y el conector de alimentación de 9V DC.

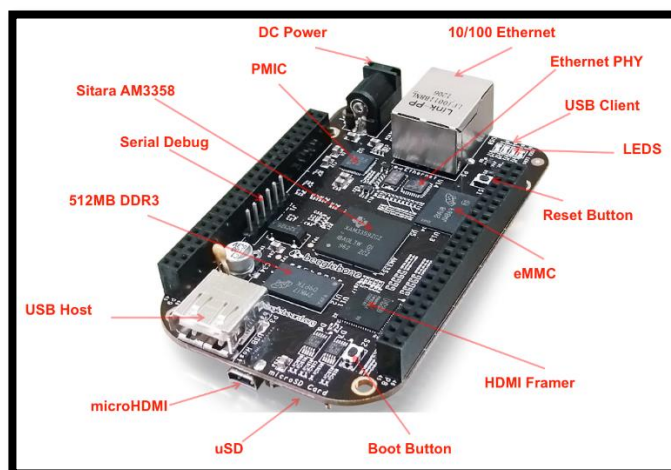


Figura 3.4: BeagleBone Black Rev. C

Para lo cual se debe seguir las siguientes instrucciones para poder instalar los servicios, servidores y controladores necesarios que se utilizarán para este sistema hidropónico.

Actualizar el kernel, las dependencias y paquetes

```
# sudo apt-get update  
# sudo apt-get upgrade
```

Instalar el editor de texto nano

```
# sudo apt-get install nano
```

Configurar la red con una IP estática 192.168.1.24/24 y puerta de enlace 192.168.1.1

```
#nano /etc/network/interfaces  
  
# The loopback network interface  
auto lo  
iface lo inet loopback  
auto eth0  
iface eth0 inet static  
address 192.168.1.24  
netmask 255.255.255.0  
network 192.168.1.0  
broadcast 192.168.1.255  
gateway 192.168.1.1
```

Instalar Python y librerías de Adafruit

```
# apt-get install python3  
# apt-get install build-essential python-dev python-setuptools python-pip python-  
smbus -y  
#pip install Adafruit_BBIO
```

Instalar Java versión 7

```
# nano /etc/apt/sources.list

deb http://ppa.launchpad.net/webupd8team/java/ubuntu precise main
deb-src http://ppa.launchpad.net/webupd8team/java/ubuntu precise main

# apt-key adv --keyserver keyserver.ubuntu.com --recv-keys EEA14886
# apt-get install oracle-java7-installer
```

Instalar base de datos MySQL

```
# apt-get install mysql-server
# /sbin/chkconfig --level 2345 mysql on
# /usr/bin/mysql -u root -p
mysql > GRANT ALL PRIVILEGES ON *.* TO 'root'@'%'
mysql > SELECT User, Host, Password FROM mysql.user;
mysql > CREATE DATABASE SMARTSiS;
mysql > INSERT INTO mysql.user (User,Host,Password)
VALUES('jossjack','%',PASSWORD('Pa$$w0rd01'));
mysql > FLUSH PRIVILEGES;

# /usr/sbin/update-rc.d mysql defaults
```

Instalar servidor de aplicaciones Wildfly versión 9.1

```
# /root/home/instalarWildfly.sh
# chkconfig --add wildfly
# chkconfig wildfly on

# wget http://central.maven.org/maven2/mysql/mysql-connector-
java/5.1.33/mysql-connector-java-5.1.33.jar
```

```

# cd /opt/wildfly/module/system/layers/base/com
# mkdir mysql
# mkdir driver
# mkdir main

# touch module.xml

<module xmlns="urn:jboss:module:1.3" name="com.mysql.driver">
  <resources>
    <resource-root path="mysql-connector-java-5.1.33.jar" />
  </resources>
  <dependencies>
    <module name="javax.api"/>
    <module name="javax.transaction.api"/>
  </dependencies>
</module>

```

Archivo standalone.xml que se encuentra en el directorio del servidor wildfly.

```

#nano /opt/wildfly/standalone/configuration/standalone.xml
...
<driver name="mysql" module="com.mysql.driver">
  <driver-class>com.mysql.jdbc.Driver</driver-class>
</driver>

```

Instalar servicio NTP

```

# apt-get install ntp
# nano /etc/ntp.conf

server 0.south-america.pool.ntp.org
server 1.south-america.pool.ntp.org

```

Agregar reglas al cortafuego

```
# nano /etc/iptables.rules
```

```
*filter
```

```
# Permitir todo el tráfico de loopback (lo0) y denegar todo el tráfico de 127/8 que  
no utiliza lo0
```

```
-A INPUT -i lo -j ACCEPT
```

```
-A INPUT ! -i lo -d 127.0.0.0/8 -j REJECT
```

```
# Aceptar todas las conexiones entrantes previamente establecidas
```

```
-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
```

```
# Aceptar todo el tráfico saliente
```

```
# Puede modificar esto para aceptar cierto tráfico únicamente
```

```
-A OUTPUT -j ACCEPT
```

```
# Permitir HTTP y HTTPS desde cualquier lugar
```

```
-A INPUT -p tcp --dport 8080 -j ACCEPT
```

```
-A INPUT -p tcp --dport 443 -j ACCEPT
```

```
-I INPUT -p tcp --dport 3306 -m state --state NEW,ESTABLISHED -j ACCEPT
```

```
-I OUTPUT -p tcp --sport 3306 -m state --state ESTABLISHED -j ACCEPT
```

```
-I INPUT -m state --state NEW -m tcp -p tcp --dport 9990 -j ACCEPT
```

```
# Permitir las conexiones de SSH
```

```
# EL NUMERO -dport ES EL MISMO QUE CONFIGURÓ EN EL ARCHIVO  
SSHD_CONFIG
```

```
-A INPUT -p tcp -m state --state NEW --dport 22 -j ACCEPT
```

```
# Permitir ping
```

```
-A INPUT -p icmp -m icmp --icmp-type 8 -j ACCEPT
```

```
# Registrar todas las llamadas denegadas por iptables (accesar vía el comando  
'dmesg')
```

```
-A INPUT -m limit --limit 5/min -j LOG --log-prefix "iptables denied: " --log-level 7
```

```
# Rechazar todo el otro tráfico de entrada - de manera predeterminada a menos  
de que exista una política que lo permita:
```

```
-A INPUT -j REJECT
```

```
-A FORWARD -j REJECT
```

```
COMMIT
```

```
# iptables-restore < /etc/iptables.rules
```

Para los sensores de PH, conductividad eléctrica, oxígeno disuelto, potencial de reducción u oxidación, se utilizará un selector de canal para lo cual se hará uso de los pines P9.41 y P9.42. En la Tabla 5 podemos observar cómo se mapea el selector de sensores que utilizaremos. Los datos se recibirán a través del puerto UART 1. En la Figura 3.5 podemos ver los sensores de calidad de agua que utilizaremos para este proyecto.



Figura 3.5: Sensores para medir la calidad del agua

S1	S0	SENSOR
0	0	PH
0	1	ORP
1	0	D.O.
1	1	---

Tabla 5: Selector de canal para los sensores asociados a UART1

Para el sensor de conductividad eléctrica se utilizará el puerto serial UART4, esto es debido a que este sensor al tener un oscilador interno puede afectar las medidas de los otros sensores.

Para el sensor de temperatura usaremos el DS18B20 un sensor digital con rango de medición entre -55 a 125 °C, 12 bits de resolución y un margen de error de lectura de ± 0.5 °C. Ver el sensor DS18B20 en la Figura 3.6.

Este sensor usa el protocolo de comunicación 1-wire por lo que solo necesitaremos un pin para los datos el cual estará conectado al pin P9.12.



Figura 3.6: Sensor de temperatura DS18B20.

Para medir la intensidad solar relativa usaremos una LDR que varía con respecto a todo el espectro de la luz, es decir a menos luz una resistencia casi cero y más luz una resistencia alta, esto conectado a un ADC se puede medir relativamente la proporción de luz en ese espacio donde se encuentra el

sensor. Ver el sensor de intensidad solar relativa en la Figura 3.7. Este sensor estará conectado al pin P9.39 un pin de entrada analógica.

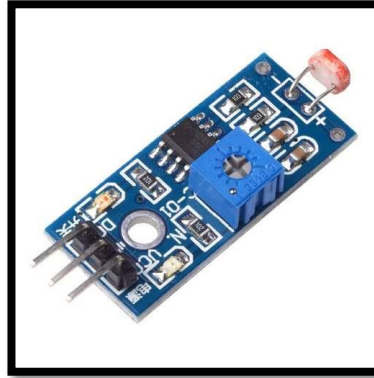


Figura 3.7: Sensor de intensidad solar relativa.

El sensor de humedad relativa y temperatura ambiental que utilizaremos es el DHT11, un sensor digital con un rango de operación de temperatura entre 0 a 55°C, con una resolución de 8 bit para la temperatura y 8 bits para la humedad relativa y un margen de error de 5% para la humedad relativa y $\pm 2^{\circ}\text{C}$ de temperatura. Ver el sensor DHT11 en la Figura 3.8. Este sensor estará conectado al pin P8.11.

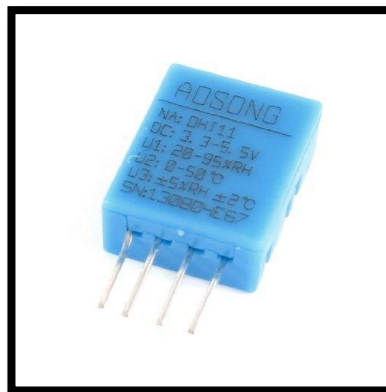


Figura 3.8: Sensor de humedad relativa y temperatura DHT11

Para los actuadores dispondremos de dos pines P8.3 para la bomba de agua y el P8.4 para el sistema de riego el cual enviara una señal digital en alto para encender o una señal digital en bajo para apagar.

Para poder hacer uso de estas señales dispondremos de una placa con relés que servirán de conmutadores con los equipos que trabajan con la red eléctrica de 110V AC. En la Figura 3.9 podemos observar el circuito de conmutación del cual disponemos para este proyecto.

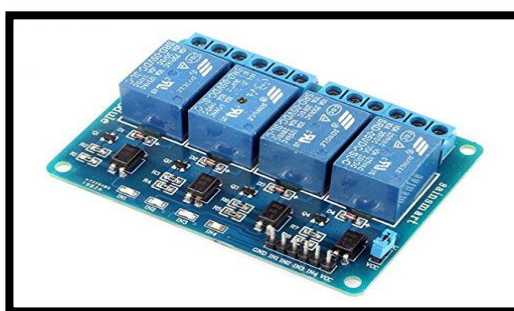


Figura 3.9: Placa de 4 relés activados a 5V DC.

Se debe tener en cuenta que alguno de estos puertos o pines deben ser inicializados antes de ser usados por lo que se deberá crear una tarea de inicialización de parámetros al arrancar el sistema.

3.2 Procesos y algoritmos

Para el correcto funcionamiento del sistema de hidroponía se detallan a continuación los procesos y algoritmos más importantes.

Proceso de inicialización de parámetros: en este proceso se cargan todos los módulos necesarios, servicios y variables de entorno.

```
# nano /etc/rc.local
```

```
#!/bin/sh -e
```

```
python /root/smartsis/lcd.py "BEAGLEBONE BLACK" "INICIANDO"
```

```
iptables-restore < /etc/iptables.rules
```

```

echo 44 > /sys/class/gpio/export
echo out > /sys/class/gpio/gpio44/direction
chmod 777 /sys/class/gpio/gpio44/value
echo 26 > /sys/class/gpio/export
echo out > /sys/class/gpio/gpio26/direction
chmod 777 /sys/class/gpio/gpio26/value
echo 1 > /sys/class/gpio/gpio44/value
echo 1 > /sys/class/gpio/gpio26/value
sleep 30
python /root/smartsis/lcd.py "BEAGLEBONE BLACK" "WILDFLY INICIANDO"
/etc/init.d/wildfly start
echo "WILFLY START"
python /root/smartsis/lcd.py "BEAGLEBONE BLACK" "WILDLFY UP"
echo BB-W1:00A0 > /sys/devices/bone_capemgr.9/slots
exit 0

```

Proceso de sensores de agua: este proceso se lo ha subdivido en varias tareas de manera que puedan ser en cualquier momento modificados o eliminados, a continuación veremos en la Figura 3.10. Como están formado este proceso.

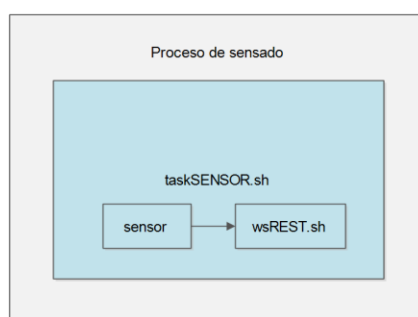


Figura 3.10: Estructura del proceso de sensado.

En el proceso sensorAgua.sh vemos como están agrupadas las siguientes tareas:

```
# cat sensorAgua.sh
```

```
#!/bin/bash
/root/smartsis/CE/taskCE.sh
/root/smartsis/PH/taskPH.sh
/root/smartsis/DO/taskDO.sh
/root/smartsis/ORP/taskORP.sh
/root/smartsis/DS18B20/taskDS18B20.sh
```

Ahora analizaremos a continuación cada una de las tareas.

```
# cat taskPH.sh
```

```
#!/bin/bash
python /root/smartsis/PH/uart.py | xargs /root/smartsis/wsREST.sh
# cat taskCE.sh
```

```
#!/bin/bash
python /root/smartsis/CE/uart.py | xargs /root/smartsis/wsREST.sh
```

```
# cat taskORP.sh
#!/bin/bash
python /root/smartsis/ORP/uart.py | xargs /root/smartsis/wsREST.sh
```

```
# cat taskDO.sh
#!/bin/bash
python /root/smartsis/DO/uart.py | xargs /root/smartsis/wsREST.sh
```

```
# cat taskDS18B20.sh
#!/bin/bash
echo "ds18b20"
echo "1616"
```

```
echo "C1S5"
cat /sys/bus/w1/devices/28-001414ab79ff/w1_slave | grep -o "t=.*" | egrep -o '\-
?[0-9]*\.[0-9]*'
```

En el proceso sensorAmbiente.sh vemos como están agrupadas las siguientes tareas:

```
# cat sensorAgua.sh

#!/bin/bash
/root/smartsis/LDR/taskLDR.sh
/root/smartsis/DHT11/taskDHT11.sh
```

A continuación detallamos cada una de estas tareas.

```
# cat taskLDR.sh
```

```
#!/bin/bash
python /root/smartsis/LDR/ldr.py | xargs /root/smartsis/wsREST.sh
```

```
# cat taskDHT11.sh
```

```
#!/bin/bash
python /root/smartsis/DHT11/dht11.py 11 P8_11 | xargs
/root/smartsis/wsREST.sh
```

Para poder activar los actuadores tenemos los siguientes script activar.sh y desactivar.sh tanto para la bomba de agua como para el sistema de riego.

```
#cat activar.sh
```

```
#!/bin/sh
```

```
echo 0 > /sys/class/gpio/gpio44/value
python /root/smartsis/lcd.py "BEAGLEBONE BLACK" "ACTIVANDO AHORA"
"BOMBA DE AGUA" "ENCENDIDA"
```

```
# cat desactivar.sh
```

```
#!/bin/sh
echo 1 > /sys/class/gpio/gpio44/value
python /root/smartsis/lcd.py "BEAGLEBONE BLACK" "ACTIVANDO AHORA"
"BOMBA DE AGUA" "APAGADA"
```

```
# cat activar.sh
```

```
#!/bin/sh
echo 0 > /sys/class/gpio/gpio26/value
python /root/smartsis/lcd.py "BEAGLEBONE BLACK" "ACTIVANDO AHORA"
"RIEGO DE AGUA" "ENCENDIDO"
```

```
#cat desactivar.sh
```

```
#!/bin/bash
echo 1 > /sys/class/gpio/gpio26/value
python /root/smartsis/lcd.py "BEAGLEBONE BLACK" "ACTIVANDO AHORA"
"RIEGO DE AGUA" "APAGADO"
```

Para la automatización de las tareas usaremos el comando `crontab -e` que nos permite programar tareas según el siguiente patrón:

```
minutos|horas|días|mes|dia de la semana|usuario|comando a ejecutar
```

```
* todos
```

```
# crontab -e
# m h dom mon dow  command
*/10 * * * * /root/smartsis/sensorAgua.sh
*/10 * * * * /root/smartsis/sensorAmbiente.sh
```

3.3 Sistemas de monitoreo, análisis y control

El sistema SMARTSiS es un sistema especialmente diseñado para el monitoreo, análisis de datos y control remoto. Su arquitectura está basada en el patrón MVC, está programada en java JSF 2.0 y para las vistas el framework PrimeFaces 5.2.

Este sistema hace uso de la persistencia de datos JPA 2.0 de manera que permite relacionar fácilmente las tablas con los modelos relacionales del sistema y así poder manipular los datos de manera segura y más flexible.

En la Figura 3.11 podemos observar el archivo persistence.xml en el cual configuramos el JPA con el datasources y los modelos de datos. Para conectarnos a la base de datos, haremos uso de un datasources que nos proveerá el servidor de aplicaciones wildfly el cual está configurado de la siguiente manera:

Nombre: dsMySQL

JNDI: java:jboss/dsMySQL

URL: jdbc:mysql://192.168.1.24:3306/SMARTSiS



```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <persistence version="2.0" xmlns="http://java.sun.com/xml/ns/persistence"
3     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4     xsi:schemaLocation="http://java.sun.com/xml/ns/persistence http://java.sun.com/xml/ns/persistence/persistence_2_0.xsd">
5     <persistence-unit name="SMARTSiSPU" transaction-type="RESOURCE_LOCAL">
6         <non-jta-data-source>java:jboss/dsMySQL</non-jta-data-source>
7         <class>com.smart.model.Rol</class>
8         <class>com.smart.model.Usuario</class>
9         <class>com.smart.model.TipoNodo</class>
10        <class>com.smart.model.Nodo</class>
11        <class>com.smart.model.Lectura</class>
12        <class>com.smart.model.Sensor</class>
13        <class>com.smart.model.ListaBean</class>
14        <class>com.smart.model.Actuador</class>
15        <class>com.smart.model.Estado</class>
16        <exclude-unlisted-classes>true</exclude-unlisted-classes>
17        <validation-mode>NONE</validation-mode>
18        <properties>
19        </properties>
20    </persistence-unit>
21 </persistence>
```

Figura 3.11: Archivo de persistencia del sistema SMARTSiS.

El sistema cuenta con un método de cifrado de contraseña de manera que si son capturadas no puedan ser descifradas. Para esto hace uso del algoritmo SHA256 que permite cifrar la contraseña en un hash de 256 bits de longitud.

La codificación hash de una cadena vacía sería:

SHA1("") = da39a3ee5e6b4b0d3255bfef95601890afd80709

También cuenta con un módulo de autenticación basado en filtros de sesiones de manera que solo los usuarios registrados puedan acceder a las páginas u opciones según su rol asignado. Ver en la Figura 3.12 la página de autenticación del sistema.

Figura 3.12: Página de autenticación del sistema SMARTSiS.

Método de autenticación de usuario

```
Usuario usuario = autenticacionService.obtenerUsuario(username, password);
```

Filtro de sesión

```
LoginBean loginBean = (LoginBean)
req.getSession().getAttribute("userLoginView");
```

Si este objeto loginBean es nulo automáticamente se re direccionara a la página de sesión para que el usuario se autentique, caso contrario le permitirá acceder a todos los módulos según su rol asignado.

Así mismo el sistema cuenta con un botón para cerrar sesión que básicamente lo que hace es invalidar la sesión, es decir limpiar todas las variables y objetos almacenados y obligarlo a re direccionar a la página de autenticación.

Al iniciar sesión correctamente veremos la pantalla de bienvenida del sistema en cual podemos observar el título del proyecto y una breve descripción. En la Figura 3.13 una vez autenticados vemos la página de bienvenida del sistema SMARTSiS. Podemos observar en el extremo izquierdo se carga los módulos del sistema según el rol del usuario.



Figura 3.13: Página de autenticación del sistema SMARTSiS.

Tenemos dos roles principales:

Usuario Administrador: permisos totales y puede ver todos los módulos existentes.

Usuario Consultor: permite cargar solo los módulos de consulta.

Módulo Consultar: este módulo nos permite visualizar los datos en tiempo real, podemos observar en la Figura 3.14. Que tenemos unos gráficos indicadores con rango según el sensor activo con un mínimo y un máximo. Esta barras se actualizarán automáticamente cada 30 segundo consultando la última lectura almacenada en la base de datos.

También podemos ver un resumen semanal de parámetros en el cual podemos rápidamente observar detalles como la media, valor mínimo, valor máximo y numero de muestras en la semana.

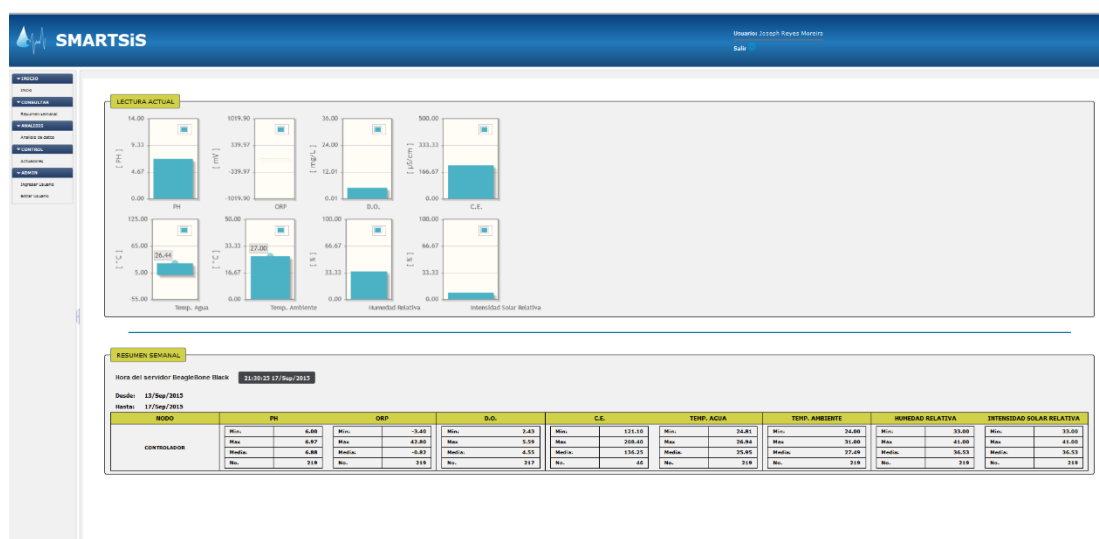


Figura 3.14: Módulo consultar del sistema SMARTSiS

Métodos o procedimientos principales de este módulo:

```
public void verResumen();
```

```
private void createBarModels();
```

```
private void createBarModel();
```

```
public void verBarras();
```

```
public void calcularResumenSemanal();
```

Estos métodos nos permitirán crear los gráficos, obtener la información más actual y calcular el resumen semanal de manera tabulada.

Módulo de análisis: este módulo nos permite realizar el análisis estadístico y descriptivo de una muestra o población de datos, que le servirá al analista ambiental para tomar decisiones en tiempo menor y prevenir cualquier inconveniente en sistema controlado. Ver en la Figura 3.15 el módulo de análisis de datos del cual podemos obtener mucha información valiosa para el analista ambiental.

Este sistema permite graficar lo siguiente:

Gráficos de barras que permiten visualizar los datos agrupados según su frecuencia acumulada por rango de valores del sensor.

Gráfico de diagrama de caja que permite visualizar rápidamente toda la estadística descriptiva, media, varianza, desviación estándar, moda, mediana, primer cuartil, segundo cuartil, tercer cuartil, datos atípicos, valor mínimo, valor máximo y número de muestra.

Gráfico de serie temporal que permite visualizar a lo largo del tiempo como se comportan estos datos.

Gráfico de distribución que permite visualizar la estadística inferencial a través del modelo normal estándar, para esto no basamos en el teorema del límite central que nos dice que un número de muestras independientes entre sí, mayor o igual a 30 y que siguen un mismo modelo tienden a una distribución normal.

Gráfico de registros de datos que permite visualizar y exportar las lecturas del sensor en una hoja de Excel.

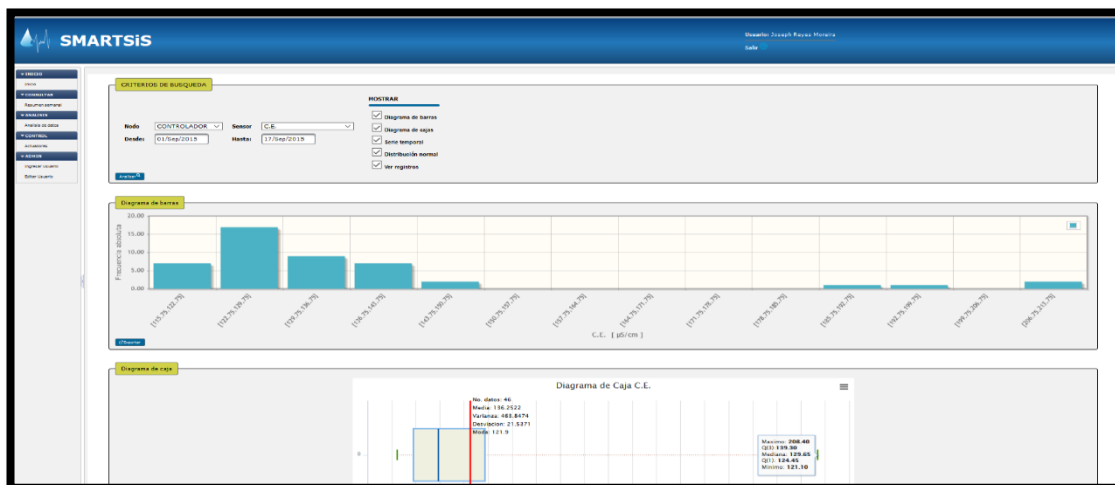


Figura 3.15: Modulo de análisis del sistema SMARTSiS

Métodos o procedimientos principales de este módulo:

```
public void verGraficas();
```

```
private BarChartModel initBarModel(String simbolo, Estadistica est);
```

```
public void verBarras(List<Lectura> Lectura, Sensor S, FacesContext context);
```

```
public void verDiagramaSerie(List<Lectura> Lectura, Sensor S, FacesContext context);
```

```
public void verDistribucionNormal(List<Lectura> Lectura, Sensor S, FacesContext context);
```

```
public void verDiagramaCaja(List<Lectura> Lectura, Sensor S, FacesContext context);
```

```
public void verSensores();
```

Estos métodos o procedimientos permiten crear los diferentes tipos de gráficas, calcular la estadística en base a los datos obtenidos en un periodo de tiempo definido por el usuario.

Módulo de control: permite visualizar todos los actuadores relacionados a un nodo de manera que pueda interactuar remotamente con diferentes dispositivos. Como podemos observar en la Figura 3.16. Vemos dos actuadores, una bomba de agua y una electroválvula que activará los aspersores.

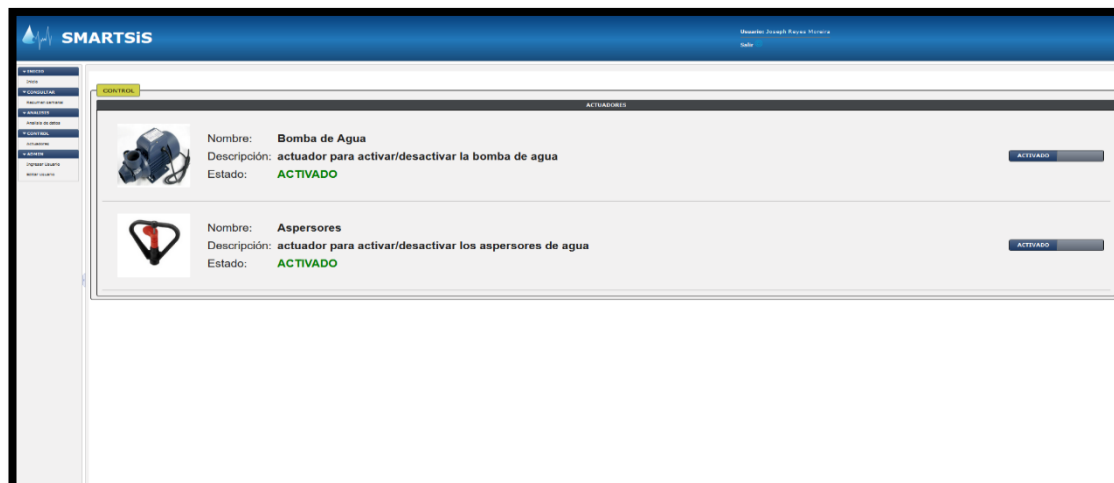


Figura 3.16: Modulo de control del sistema SMARTSiS

Métodos o procedimientos principales de este módulo:

```
public void obtenerEstados();
```

```
public void cambiarEstado(Estado estado, Actuador actuador);
```

```
public String executeCommand(String command);
```

Estos métodos o procedimientos permiten consultar el último estado de un actuador, cambiar el estado y ejecutar un script remotamente.

Módulo de administración: permite crear, modificar o eliminar un usuario del sistema. Este módulo permite asignar al usuario un rol específico que estará relacionado con los modulos a los cuales tendrá permiso. Ver en la Figura 3.17 el módulo de administración del sistema.

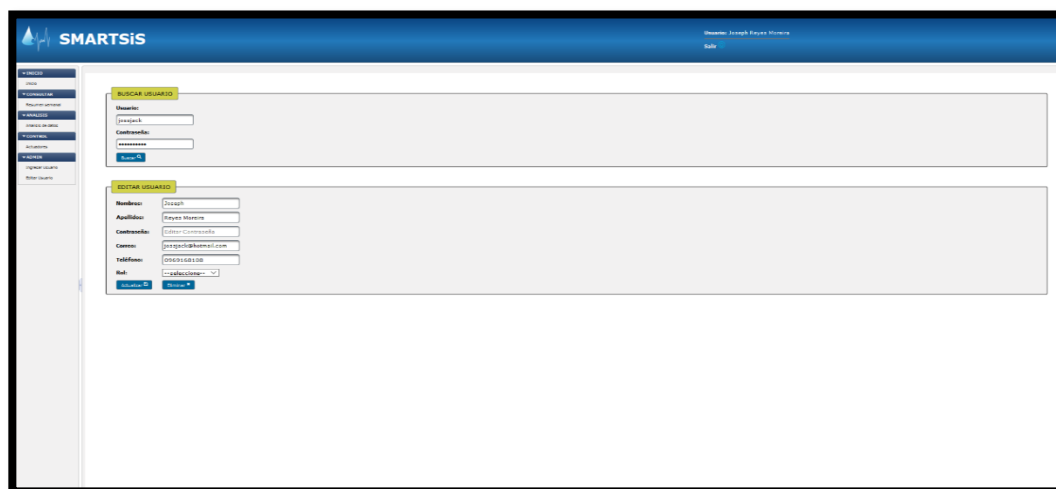


Figura 3.17: Modulo de administración del sistema SMARTSiS

Métodos o procedimientos principales de este módulo:

```
public void buscarUsuario();
```

```
public void insertarUsuario();
```

```
public void eliminarUsuario();
```

```
public void actualizarUsuario();
```

Estos métodos o procedimientos tienen la metodología CRUD que permite crear, consultar, actualizar y eliminar.

Por último detallaremos el funcionamiento del servicio web que implementa el sistema SMARTSiS con el cual los procesos de la BeagleBone Black se conectan para realizar las transacciones correspondientes.

Este servicio web utiliza la metodología REST que es una arquitectura de software para sistemas distribuidos tales como la web que funciona mediante requerimientos de GET, POST, PUT y DELETE, por lo que no requiere de implementaciones especiales y el cliente es independiente del lenguaje en que

se encuentre programado. Puede utilizar tecnologías como XML o JSON como contenedores de información.

Url del servicio web implementado:

<http://smartsis.ddns.net:2525/SMARTSiS/rest/servicioREST/>

Métodos remotos del servicio web:

Método que permite obtener la lectura de un sensor DHT11 y almacenar su información en la base de datos.

@POST

```
@Path("/dth11/{key}/{codigo1}/{codigo2}/{valorTemp}/{valorHum}")
```

```
@Produces(MediaType.TEXT_PLAIN)
```

```
public String obtenerTemAmbHumRel(
```

```
@PathParam("key") String key,
```

```
@PathParam("codigo1") String codigo1,
```

```
@PathParam("codigo2") String codigo2,
```

```
@PathParam("valorTemp") Double valorTemp,
```

```
@PathParam("valorHum") Double valorHum);
```

Método que permite obtener la lectura de un sensor DS18B20 y almacenar su información en la base de datos.

@POST

```
@Path("/ds18b20/{key}/{codigo}/{valor}")
```

```
@Produces(MediaType.TEXT_PLAIN)
```

```
public String obtenerTemAgua(
```

```

@PathParam("key") String key,
@PathParam("codigo") String codigo,
@PathParam("valor") Double valor);

```

Método que permite obtener la lectura de una LDR y almacenar su información en la base de datos.

```

@POST
    @Path("/ldr/{key}/{codigo}/{valor}")
    @Produces(MediaType.TEXT_PLAIN)
    public String obtenerISR(
@PathParam("key") String key,
@PathParam("codigo") String codigo,
@PathParam("valor") Double valor)

```

Método que permite obtener la lectura del PH y almacenar su información en la base de datos.

```

@POST
    @Path("/ph/{key}/{codigo}/{valor}")
    @Produces(MediaType.TEXT_PLAIN)
    public String obtenerCE(
@PathParam("key") String key,
@PathParam("codigo") String codigo,
@PathParam("valor") String valor);

```

Método que permite obtener la lectura de la conductividad eléctrica y almacenar su información en la base de datos.

```
@POST
    @Path("/ce/{key}/{codigo}/{valor}")
    @Produces(MediaType.TEXT_PLAIN)
    public String obtenerCE(
        @PathParam("key") String key,
        @PathParam("codigo") String codigo,
        @PathParam("valor") String valor);
```

Método que permite obtener la lectura del oxígeno disuelto en el agua y almacenar su información en la base de datos.

```
@POST
    @Path("/do/{key}/{codigo}/{valor}")
    @Produces(MediaType.TEXT_PLAIN)
    public String obtenerCE(
        @PathParam("key") String key,
        @PathParam("codigo") String codigo,
        @PathParam("valor") String valor);
```

Método que permite obtener la lectura del ORP y almacenar su información en la base de datos.

```
@POST
    @Path("/orp/{key}/{codigo}/{valor}")
    @Produces(MediaType.TEXT_PLAIN)
```



```

public String obtenerCE(
@PathParam("key") String key,
@PathParam("codigo") String codigo,
@PathParam("valor") String valor);

```

Por último se configuró un servicio DDNS que permitirá re direccionar una IP pública a una dirección privada a través de un puerto específico. Para esto utilizaremos el servicio gratuito básico de la web <http://www.noip.com/>. Ver en la Figura 3.18 la página que ofrece el servicio gratuito de ddns.

Primero tendremos que registrarnos, una vez realizado esto configuraremos la dirección ir pública que nos asigna nuestro proveedor de internet.

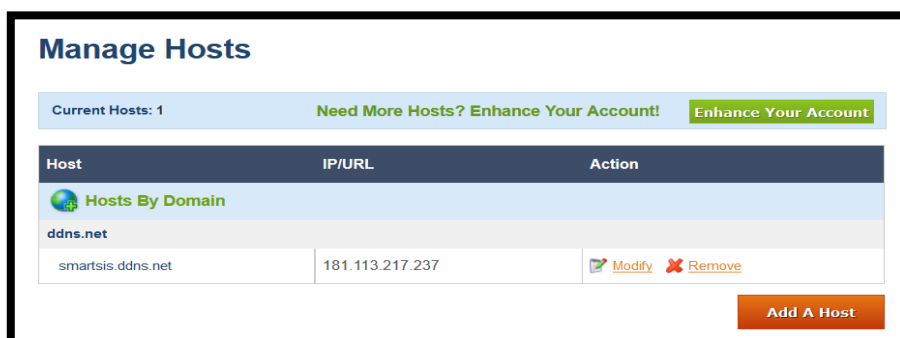


Figura 3.18: Administración de hosts de la página web www.noip.com

Posteriormente en nuestro enrutador configuraremos en la opción DNS donde colocaremos los datos principales como nombre del proveedor de servicios, host, dominio, dirección IP del servidor, puerto del servidor, etc. Ver en la Figura 3.19 la pantalla de configuración de servicio ddns de nuestro enrutador.

DDNS								New	Remove	Help
Service Provider	WAN Connection	Enable	Protocol	Domain	User Name	Password	Remove			
WILDFLY	INTERNET_TR069_R_0_35	Enable	GNUDip http	smartsis ddns.net	16121987	*****	<input type="checkbox"/>			

Settings	
Service provider: <input type="text" value="Others"/>	WAN connection: <input type="text" value="INTERNET_TR069_R_0_35"/>
Host: <input type="text" value="smartsis"/>	Domain: <input type="text" value="ddns.net"/>
User name: <input type="text" value="16121987"/>	Password: <input type="text" value="*****"/>
Server address: <input type="text" value="192.168.1.24"/>	Server port: <input type="text" value="8080"/>
Protocol: <input type="text" value="GNUDip http"/>	Service name: <input type="text" value="WILDFLY"/>
<input type="button" value="Submit"/>	

Figura 3.19: Página de configuración DDNS de nuestro enrutador

CAPÍTULO 4

4. PRUEBAS DEL SISTEMA Y ANÁLISIS DE RESULTADOS

En este capítulo se revisará las pruebas y análisis de resultados realizados al controlador y al ambiente de prueba detallando cómo interactúan los parámetros entre sí, y como deben responder los actuadores ante el cambio brusco de uno de los parámetros. Otro punto de análisis será la prueba de estrés realizada al controlado para el evaluar su rendimiento y corroborar su elección para este proyecto.

4.1 Toma de lecturas en tiempo real

El controlador está programado para tomar lecturas cada 10 minutos estos parámetros se almacenarán en la base de datos para posteriormente ser visualizada. En la Figura 4.1 podemos observar el cajetín con los componentes electrónicos, mecanismos de relés, sistema de sensores y el controlador BeagleBone Black Rev. C.

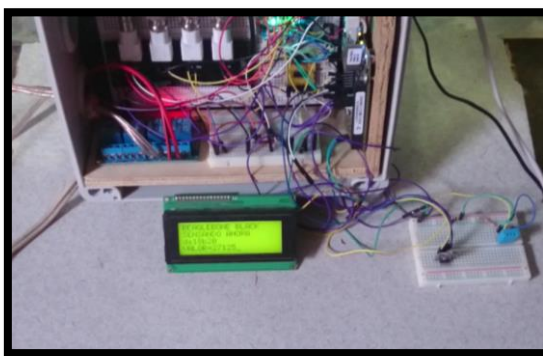


Figura 4.1: Cajetín que integra todos los componentes electrónicos y el controlador BeagleBone Black.

En la Figura 4.1 podemos observar los valores actuales de los parámetros de calidad del agua y ambientales.

El PH se mide en un rango de 0 a 14, la barra nos muestra una lectura de 6.8 la cual podemos visualizar si ponemos el puntero del ratón encima de ella. Esta lectura nos indica que el agua entra dentro del rango adecuado de PH que es de 6.5 a 8.0.

El ORP se mide en un rango de -1019.90 a 1019.90 mV, la barra nos muestra una lectura de 42mV, esto nos quiere decir que el agua tiene material contaminante presente.

Esto se puede explicar de la siguiente manera, a más contaminantes en el agua da como resultado menos oxígeno disuelto, debido a que este lo consumen los microorganismos dando un bajo nivel de ORP. Un alto ORP es un indicativo de que las bacterias y otros organismos han sido destruidos. Ver en la Tabla 6 los valores aceptables de ORP en las aplicaciones más importantes.

Nivel de ORP	Aplicaciones
0-150	Sin uso práctico
150-250	Acuicultura
250-350	Torres de enfriamiento
400-475	Piscinas
450-600	Piscinas de hidromasaje
600	Desinfección del agua
800	Esterilización del agua

Tabla 6: Niveles de ORP aceptables por aplicación.

El oxígeno disuelto se mide en un rango de 0 a 36 mg/L, la barra nos muestra una lectura de 6mg/L, lo cual nos indica que existe material contaminante el agua o ausencia de microorganismos que consumen el oxígeno presente. Existen

especies de agua cálida y fría que dependiendo del oxígeno disuelto en el agua tendrán un correcto desarrollo tal como vemos en la Tabla 7.

ESPECIES AGUA CÁLIDA		ESPECIES AGUA FRÍA	
0	Letal si está expuesto por un periodo de tiempo muy largo	0	Letal
1		1	
2	Crecimiento Lento	2	
3		3	
4	Rango óptimo	4	Crecimiento lento
5		5	
6		6	Rango óptimo
7		7	
		8	

Tabla 7: Estudio de niveles de oxígeno disuelto críticos entre especies de agua cálida y fría por Stanistaus Sonnenholzner 11 abril 2014

La conductividad eléctrica se mide en un rango de 0 a 500000 uS, la barra nos muestra una lectura de 208400 uS que nos indica que el nivel de salinidad o sólidos totales suspendidos está dentro del rango bajo. Existe una relación que permite estimar los sólidos totales disueltos tal como se muestra a continuación.

Los tomates toleran elevados valores de salinidad en comparación con las demás plantas, es por eso que no es recomendable cultivar tomates con otros tipos de plantas. Ver en la Tabla 8 los niveles de TDS y C.E críticos para la calidad del agua.

$$TDS(ppm) = 0.64 \times EC \left(\frac{\mu S}{cm} \right) = 640 \times EC(dS/m)$$

TDS ppm o mg/L	C.E. ds/m	Riesgo
< 500	< 0.8	Bajo
500 - 1000	0.8 – 1.6	Medio
1000 - 2000	1.6 – 3	Alto
> 2000	> 3	Muy alto

Tabla 8: Niveles de TDS y C.E. críticos en la calidad del agua

La temperatura del agua se mide en un rango de -55 a 125 °C, la barra nos muestra una lectura de 26.44 °C, que está dentro del rango adecuado para los peces 18 a 30 °C.

La temperatura del Ambiente se mide en un rango de 0 a 55°C, la barra nos muestra una lectura de 29.8°C, que está dentro del rango adecuado para las plántulas de tomate que es de 25 a 30°C, siendo el mínimo 12°C y el máximo 35°C. Ver en la Tabla 9 las temperaturas críticas para el cultivo de tomate [9].

Se congela la planta	- 2° C
Se detiene el desarrollo de la planta	10 a 12° C
Mayor desarrollo de la planta	20 a 24° C
Germinación mínima	15° C
Germinación óptima	25 a 30° C
Germinación máxima	35° C
Nascencia	18° C
Primeras hojas	12° C
Desarrollo de las flores de noche	13 a 17° C
Desarrollo de las flores de día	20 a 23° C
Cuaje de noche	14 a 17° C
Cuaje de día	20 a 25° C
Maduración del fruto rojo	15 a 22° C
Maduración del fruto amarillo	Más de 30° C
Temperatura del suelo mínima	12° C
Temperatura del suelo óptima	20 a 24° C
Temperatura del suelo máxima	34° C

Tabla 9: Temperaturas críticas para el cultivo del tomate

La Humedad relativa del Ambiente se mide en un rango de 0 a 100%, la barra nos muestra una lectura de 34.7%, esto nos indica que el lugar en donde estamos realizando las pruebas es muy seco, los rangos adecuados para el cultivo de plántulas de tomates se encuentran entre 50 a 60% de humedad.

Se deberá proveer de riego con abundante agua una o dos horas después de la salida del sol y antes de su puesta, con aspersores de $\frac{1}{4}$ de pulgada para que la gota no dañe a la plántula.

La intensidad solar relativa se mide en un rango de 0 a 100%, la barra nos muestra una lectura de 7.5% que está de acuerdo a la cantidad de luz presente en el escenario de la prueba tomada de noche.

Las plántulas de tomate deberán de contar con suficiente luz solar para que pueda realizar correctamente sus procesos biológicos [8].

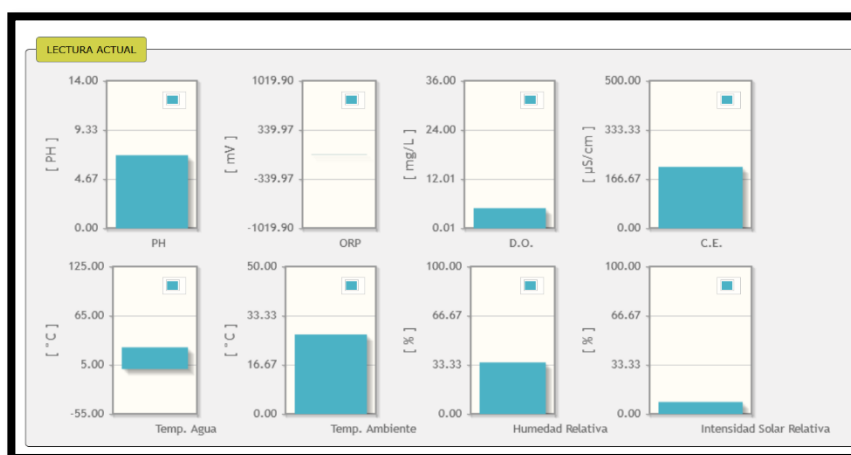


Figura 4.2: Lectura actual de parámetros del sistema SMARTSiS.

Esta sección nos muestra el resumen semanal de los parámetros de la calidad del agua y ambientales, visualizando valores mínimos y máximos, media y numero de muestras. Dando una ayuda visual rápida al analista ambiental, podemos observar en la Figura 4.3.

Todos estos valores tienen alguna relación entre sí, siendo que algunos aumentarían de valor mientras otros disminuyen y viceversa.

También se puede evidenciar que ciertos parámetros ambientales afectan directa o indirectamente la calidad del agua. Este es el caso de la temperatura ambiental, si esta disminuye también disminuye la temperatura del agua y aumenta la cantidad de oxígeno disuelto y viceversa.

El oxígeno disuelto en el agua también está relacionado al ORP esto quiere decir si el oxígeno disuelto aumenta el ORP disminuye y viceversa.

La exposición del agua a la luz solar durante un tiempo prolongado, ayuda eliminar microorganismos patógenos presentes, esto es debido a que los rayos ultravioleta sirven como purificador de agua, deteniendo el mecanismo reproductor de estos microorganismos y modificando su material genético haciendo que mueran.

RESUMEN SEMANAL																
Hora del servidor BeagleBone Black 22:00:33 17/Sep/2015																
Desde: 13/Sep/2015																
Hasta: 17/Sep/2015																
NODO	PH	ORP	D.O.	C.E.	TEMP. AGUA	TEMP. AMBIENTE	HUMEDAD RELATIVA	INTENSIDAD SOLAR RELATIVA								
CONTROLADOR	Min:	6.00	Min:	-3.40	Min:	2.43	Min:	121.10	Min:	24.81	Min:	24.00	Min:	33.00	Min:	33.00
	Max:	6.97	Max:	42.80	Max:	5.59	Max:	208.40	Max:	26.94	Max:	31.00	Max:	41.00	Max:	41.00
	Media:	6.88	Media:	-0.82	Media:	4.55	Media:	136.25	Media:	25.95	Media:	27.49	Media:	36.53	Media:	36.53
	No.	219	No.	219	No.	217	No.	46	No.	219	No.	219	No.	219	No.	219

Figura 4.3: Resumen semanal de parámetros del sistema SMARTSiS

4.2 Control de actuadores y configuración remota

Este módulo ofrece la capacidad de que el usuario pueda interactuar directamente con algún dispositivo remotamente, como es el caso de la bomba de agua y el sistema de riego.

Para lo cual el sistema muestra todos los actuadores relacionados a un nodo y botón de activación y desactivación, en el cual visualizaremos el último estado presente del dispositivo.

En la Figura 4.4 podemos observar el sistema riego acoplado a la estructura del sistema hidropónico, este sistema se puede activar remotamente mediante el módulo de control, bastará con dar clic en el botón activar para encender el sistema riego. Ver Figura 4.5. Una vez activado cambiara el texto a color verde mostrando la palabra activado, como podemos ver en el caso de la bomba de agua. El tiempo de retardo depende de la conexión y el ancho de banda del proveedor externo siendo aproximadamente 1.5 segundos.



Figura 4.4: Sistema de riego acoplado a la estructura del sistema hidropónico



Figura 4.5: Estado actual del sistema de riego desactivado.

4.3 Análisis estadístico de los parámetros medibles obtenidos

Estos resultados obtenidos que detallaremos a continuación, nos muestra como el sistema responde ante los cambios ambientales y factores eventuales internos

a lo largo del tiempo, como es el caso de los desechos orgánicos de los peces afectan parámetros como el ORP y el oxígeno disuelto.

Estos datos estadísticos nos servirán de mucha ayuda al momento de corregir deficiencias o excesos de nutrientes, materia orgánica, tiempos de producción estimados bajo ciertas condiciones parametrizadas.

Estas pruebas fueron realizadas con datos reales a lo largo de dos semanas con un periodo de tiempo de medición de 10 minutos las 24 horas del día los 7 días de la semana, en un ambiente cerrado con poca humedad y acceso regular de luz solar, durante la noche ausencia completa de luz. Se usó para la prueba un flujo continuo y circulante del agua, 4 peces ornamentales y 15 plántulas de tomate de 7 semanas.

Cada reservorio de agua contiene aproximadamente 40 litros de agua que se cambiarán cada dos semanas, el reservorio que contiene a los peces incluye un filtro para limpiar los excesos de materia inorgánica y un oxigenador de agua que produce burbujas.

Resultados y análisis del PH

En la gráfica de barras podemos visualizar una sola barra del PH a lo largo del tiempo en la cual se mantuvo estable entre 5.99 a 6.99. Esto nos indica que el agua se encuentra en un rango aceptable para sustentar la vida de los peces y las plántulas de tomate. Ver Figura 4.6.

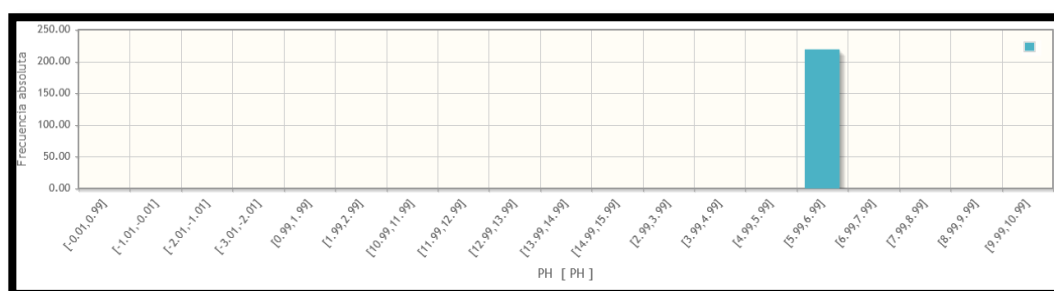


Figura 4.6: Resultados del PH graficados en un diagrama de barras.

En la gráfica del diagrama de caja del PH observamos lo siguiente:

Número de datos: 219

Media: 6.87

Varianza: 0.0093

Desviación estándar: 0.098

Moda: 6.91

Valor mínimo: 6.00

Valor máximo: 6.97

Primer cuartil: 6.86

Segundo cuartil: 6.89

Tercer cuartil: 6.93

Esto nos indica que el 50% de los datos tienen un valor de PH inferior 6.89, y podemos observar que la media se encuentra entre el 25 y 50% del total de los datos con un valor de 6.87 como vemos en la Figura 4.7.

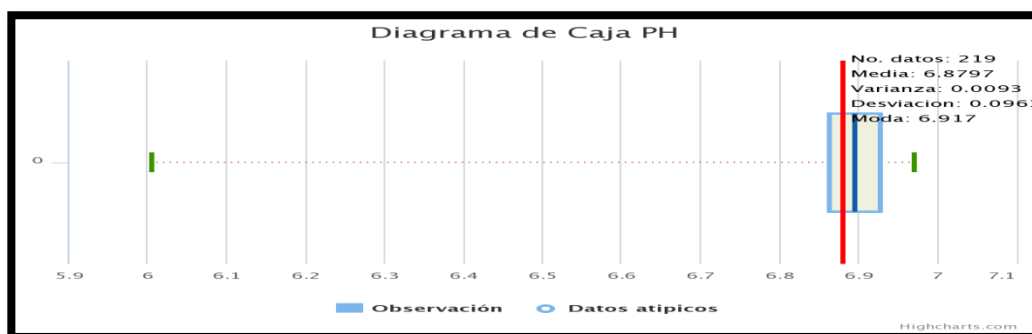


Figura 4.7: Resultados del PH graficados en un diagrama de caja.

En la gráfica de serie temporal de datos podemos visualizar rápidamente los cambios bruscos y en que sucedieron por lo que una rápida correlación de datos nos puede indicar que parámetros o variables externas provocaron estos picos.

Tal como vemos en la Figura 4.8.



Figura 4.8: Resultados del PH graficados en una serie temporal de datos.

En este gráfico podemos realizar estimaciones de los datos obtenidos para tomar decisiones basadas en predicciones probabilísticas. Vemos como la distribución normal estándar está centrada alrededor de la media que es 6.87 con una desviación estándar de 0.098. Ver Figura 4.9.

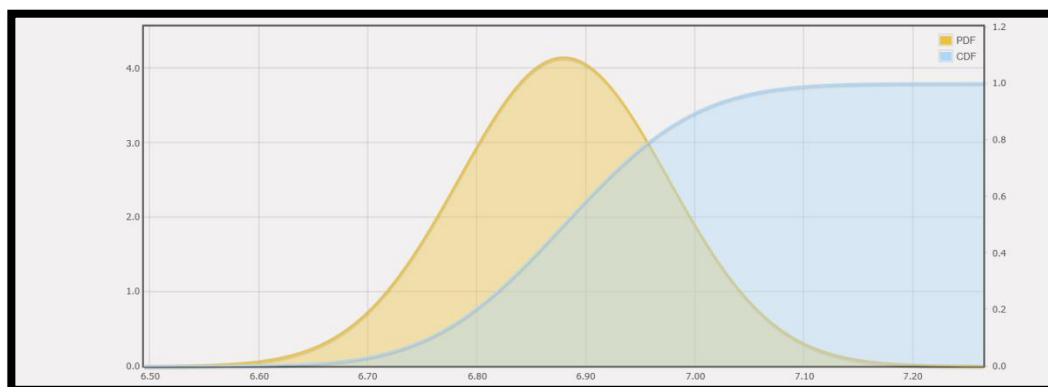


Figura 4.9: Resultados del PH graficados en el modelo de distribución normal estándar.

Resultados y análisis del ORP

En el diagrama de barras observamos como los valores se distribuyen en 5 rangos en los cuales la mayor parte de los datos esta agrupada en el rango de -2.8 a 0.2 mV esto es un indicativo de la cantidad microorganismos presentes en

el agua lo cual es resultado de los desechos de los peces, que sirven como nutrientes para las raíces de las plántulas. Ver Figura 4.10.

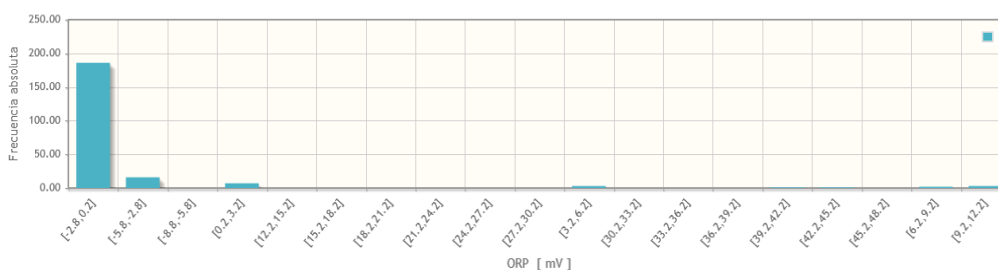


Figura 4.10: Resultados del ORP graficados en el diagrama de barras.

En la gráfica del diagrama de caja del ORP observamos lo siguiente:

Número de datos: 219

Media: -0.81

Varianza: 20.65

Desviación estándar: 4.54

Moda: -1.5

Valor mínimo: -3.40

Valor máximo: 4.2

Primer cuartil: -2.20

Segundo cuartil: -1.60

Tercer cuartil: -0.57

Observamos como la media se encuentra por encima del 50% de los datos tomados de la muestra. Ver Figura 4.11.

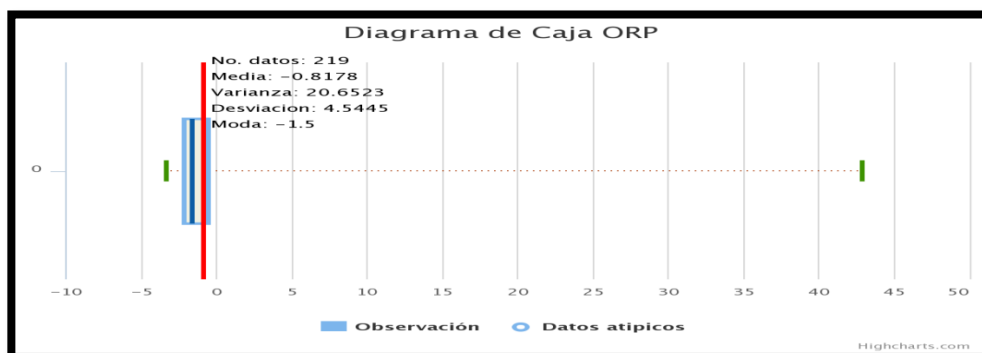


Figura 4.11: Resultados del ORP graficados en el diagrama de caja.

En esta gráfica podemos observar cómo se presenta un pico de alrededor de 42mV, esto se pudo haber dado por un cambio brusco en uno de los parámetros relacionados al ORP como puede ser el oxígeno disuelto y la temperatura. A mayor valor de ORP es un indicativo de agua más pura, libre de microorganismos. Ver Figura 4.12.



Figura 4.12: Resultados del ORP graficados en el diagrama de caja.

En la Figura 4.13 observamos el diagrama de distribución normal estándar donde vemos que esta parametrizada con una media de -0.81 y una desviación estándar de 4.54.

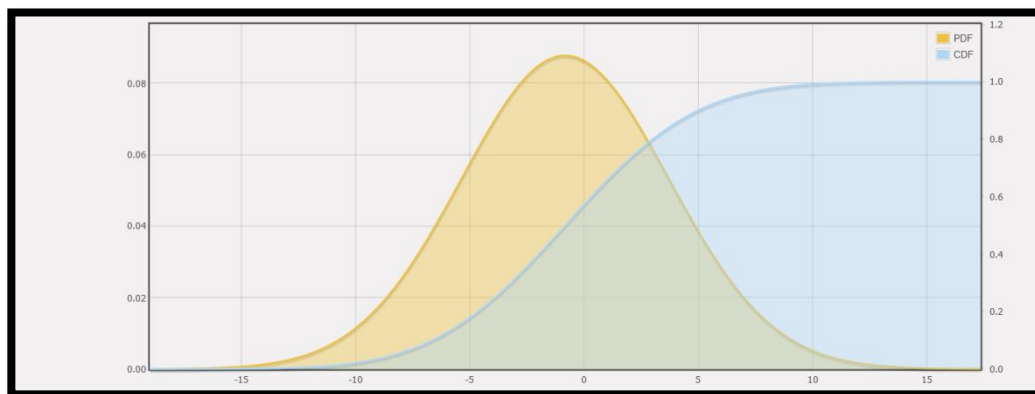


Figura 4.13: Resultados del ORP graficados en la distribución normal estándar.

Resultados y análisis del oxígeno disuelto

En el diagrama de barras podemos observar que la mayor parte de los datos se encuentran acumulados en dos rangos que son de 3.51 a 4.51 y de 4.51 a 5.51 mg/L, esto nos indica que el agua se encuentra poco oxigenada debido a factores como la temperatura y los desechos de los peces. Ver Figura 4.14.

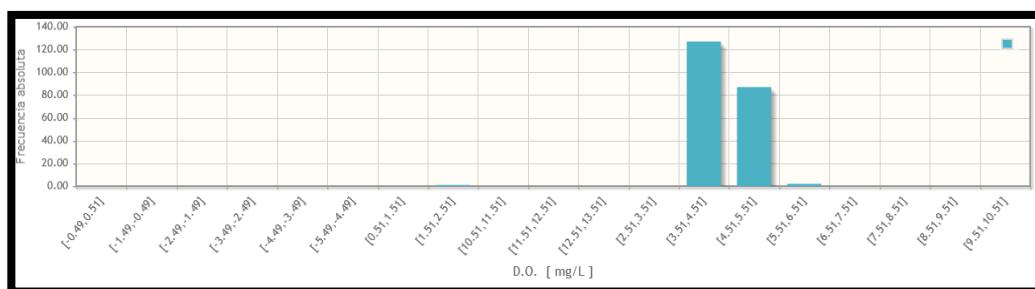


Figura 4.14: Resultados del D.O. graficados en el diagrama de barras.

En la gráfica del diagrama de caja del oxígeno disuelto observamos lo siguiente:

Número de datos: 217

Media: 4.55

Varianza: 0.28

Desviación estándar: 0.53

Moda: 4.45

Valor mínimo: 2.43

Valor máximo: 5.59

Primer cuartil: 4.19

Segundo cuartil: 4.42

Tercer cuartil: 5.09

En esta gráfica observamos como la media se encuentra por encima del 50% de los datos. Ver Figura 4.15.

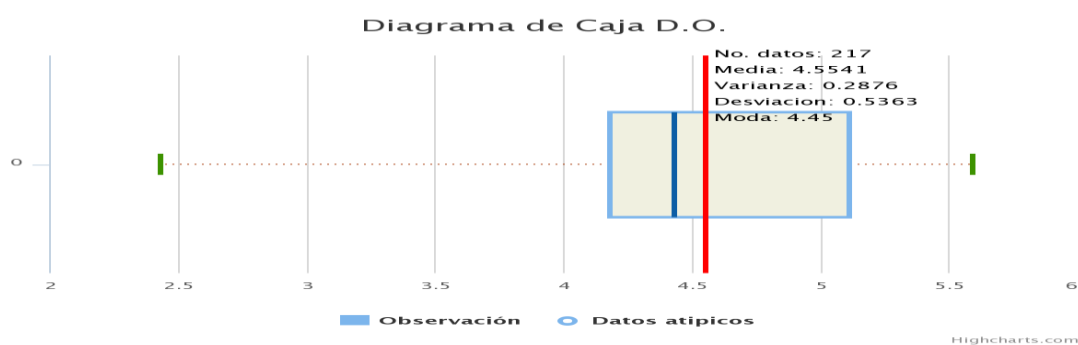


Figura 4.15: Resultados del D.O. graficados en el diagrama de caja.

En este diagrama podemos observar como el oxígeno disuelto varia rápidamente en un periodo de tiempo corto esto es debido al purificador de agua y al oxigenador que genera burbujas constantemente. Ver Figura 4.16.



Figura 4.16: Resultados del D.O. graficados en una serie temporal de datos.

En la distribución normal estándar observamos cómo esta parametrizada con una media de 4.55 y una desviación estándar de 0.53. Ver Figura 4.17.

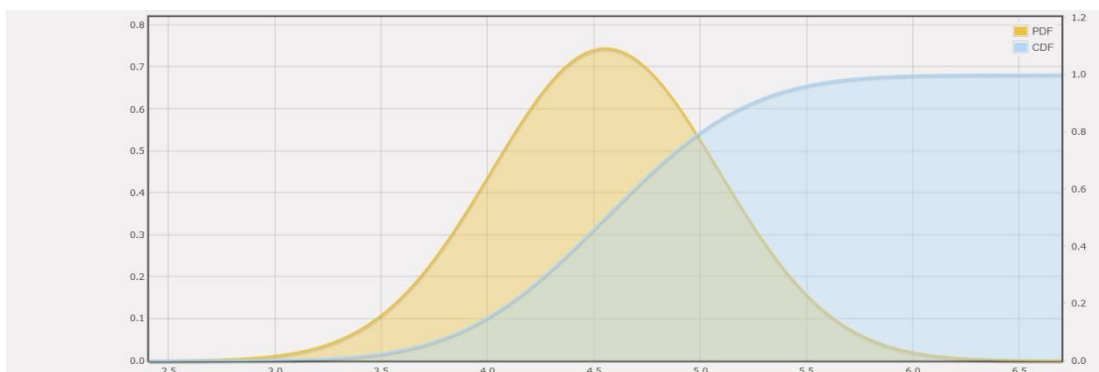


Figura 4.17: Resultados del D.O. graficados en el modelo normal estándar.

Resultados y análisis de la conductividad eléctrica

Este parámetro se encuentra agrupado en el diagrama de barras con un mayor porcentaje de datos en 122.75 a 129.75 $\mu S/cm$, estas agrupaciones nos indica que el agua contiene poco contenido salino, en altos valor de salinidad puede ser perjudicial para los peces. Ver Figura 4.18.

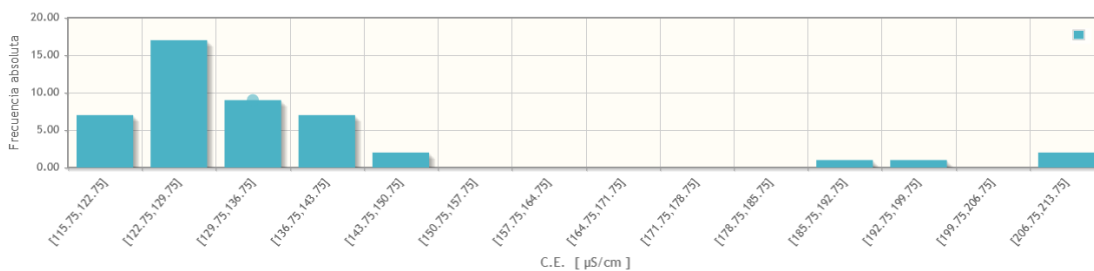


Figura 4.18: Resultados del C.E. graficados en el diagrama de barras.

En la gráfica del diagrama de caja de la conductividad eléctrica observamos lo siguiente:

Número de datos: 46

Media: 136.25

Varianza: 463.84

Desviación estándar: 21.53

Moda: 128.5

Valor mínimo: 121.10

Valor máximo: 208.4

Primer cuartil: 124.45

Segundo cuartil: 129.65

Tercer cuartil: 139.30

En el diagrama de caja observamos que la media está por encima del 50% de los datos. Ver Figura 4.19.

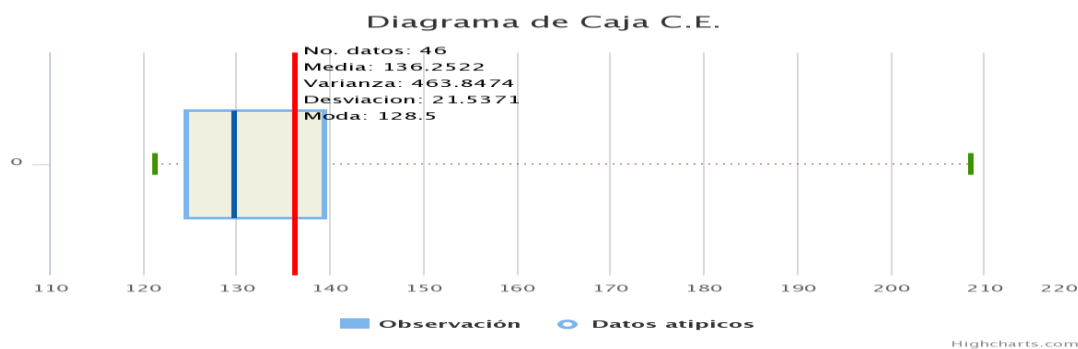


Figura 4.19: Resultados del C.E. graficados en el diagrama de caja.

En el diagrama de la serie temporal de datos vemos que hay un cambio brusco y que se mantiene, esto puede ser provocado por la temperatura ambiental o algún agente externo en el agua. Ver Figura 4.20.



Figura 4.20: Resultados del C.E. graficados en la serie temporal de datos.

La distribución normal esta parametrizada con una media de 136.25 y con una desviación estándar de 21.53. Ver Figura 4.21.

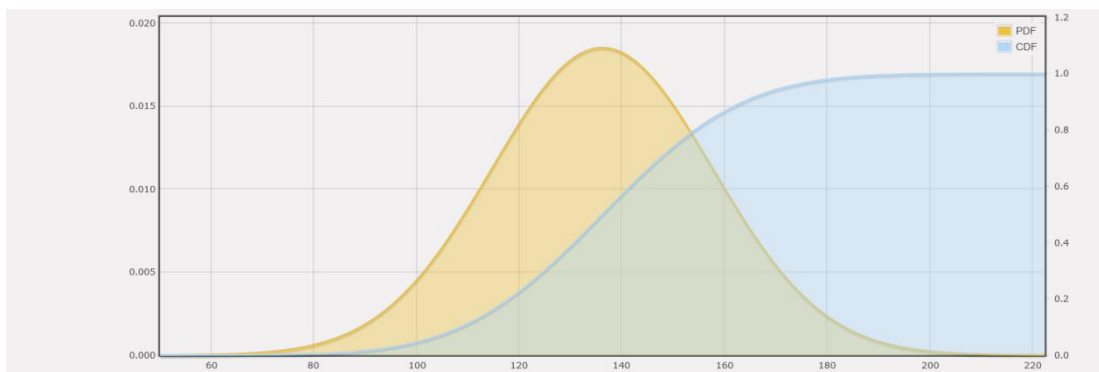


Figura 4.21: Resultados del C.E. graficados en el modelo normal estándar.

Resultados y análisis de la temperatura del agua

El diagrama de barras muestra como los datos están agrupados en su mayoría entre 24.37 a 25.37, 25.37 a 26.37 y 26.37 a 27.37, lo cual indica que toda muestra se encuentra dentro del rango adecuado de temperatura para la crianza de peces. Figura 4.22. Los cambios de temperatura hacen variar parámetros como el oxígeno disuelto y el ORP.

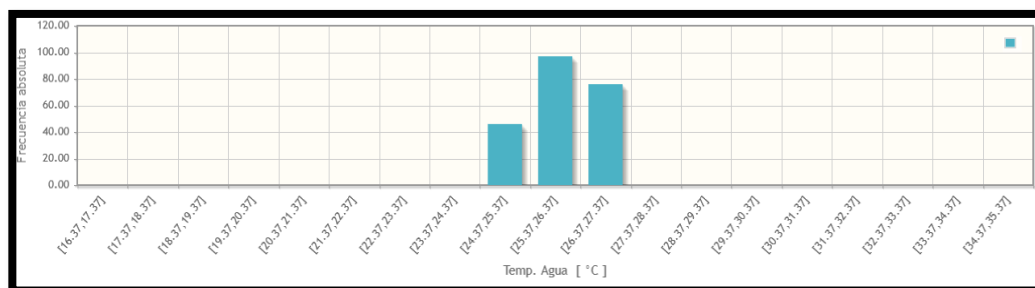


Figura 4.22: Resultados de la temperatura del agua graficados en el diagrama de barras.

En la gráfica del diagrama de caja de la temperatura del agua observamos lo siguiente:

Número de datos: 219

Media: 25.94

Varianza: 0.38

Desviación estándar: 0.61

Moda: 25.56

Valor mínimo: 24.81

Valor máximo: 26.94

Primer cuartil: 25.50

Segundo cuartil: 25.94

Tercer cuartil: 26.56

En el diagrama de caja observamos que la media está por encima del 50% de los datos. Ver Figura 4.23.

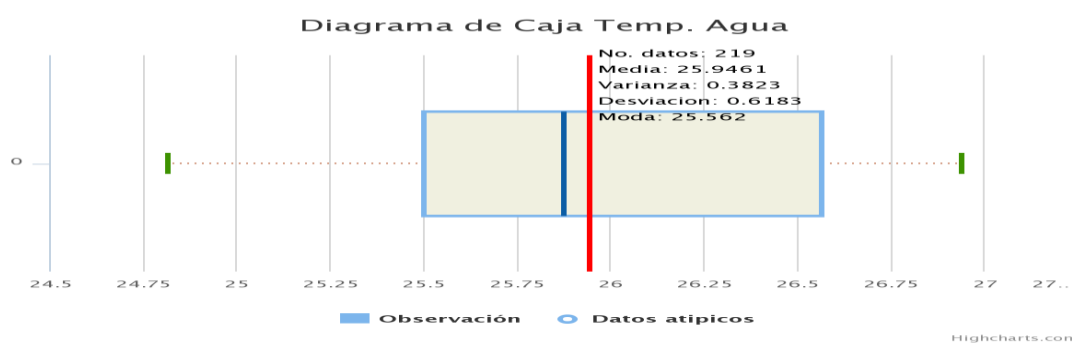


Figura 4.23: Resultados de la temperatura del agua graficados en el diagrama de caja.

La serie temporal de datos nos muestra como varia la temperatura a lo largo del día observado mayores temperaturas por las tardes y una menor temperatura por las noches. Ver Figura 4.24.

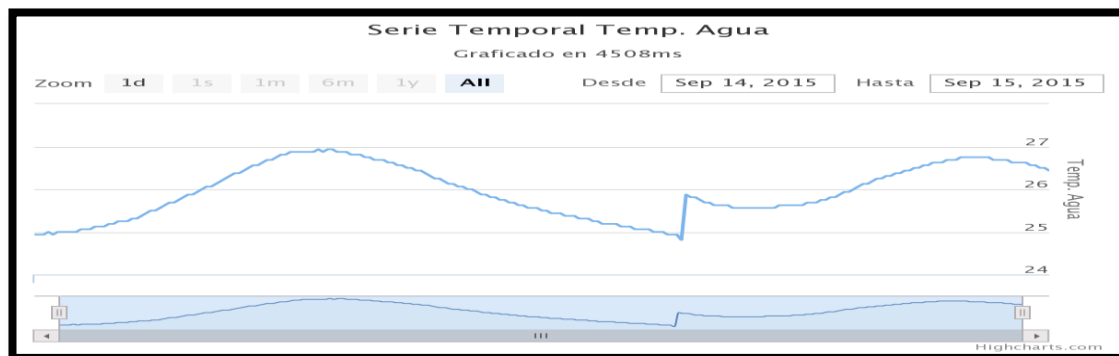


Figura 4.24: Resultados de la temperatura del agua graficados en la serie temporal de datos.

La distribución normal estándar esta parametrizado con una media de 25.94 y una desviación estándar de 0.61. Ver Figura 4.25.

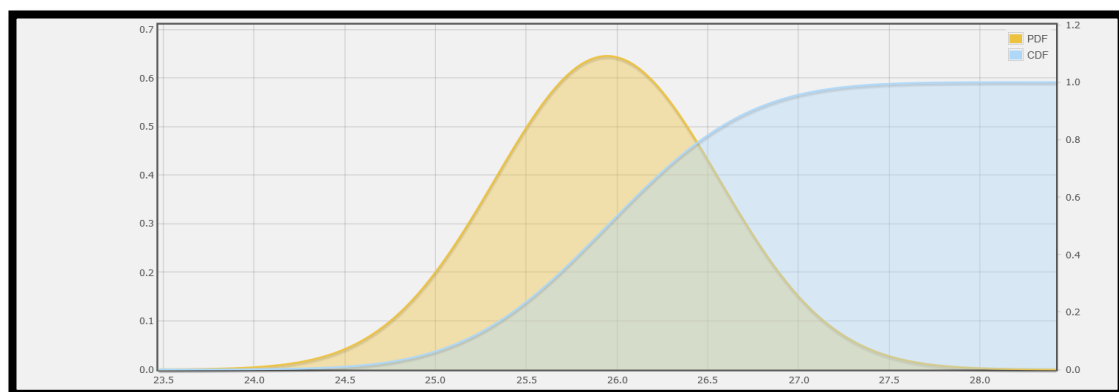


Figura 4.25: Resultados de la temperatura del agua graficados en el modelo normal estándar.

3.4 Resultados de rendimiento del controlador.

Se realizó las siguientes pruebas de retardo entre que se da clic al botón hasta que se activa el actuador, esta prueba fue realizada bajo los siguientes criterios.

Número de muestras: 50

Ancho de banda del proveedor externo: 3.6 Mbps

Escenario: todo el ancho de banda aplicado solo a esta prueba

Navegador usado: Firefox v40.0.3

Tiempo promedio hasta que se activa el actuador: aproximadamente 1.65 segundos.

Para realizar la prueba de estrés del servidor web y el controlador usaremos el software de evaluación JMeter desarrollado en java que nos permitirá crear ambientes de prueba parametrizados bajo varios hilos simultáneos y con periodos de tiempo programados.

Esta herramienta es OpenSource Apache, podemos usarla para evaluar protocolos como TCP/IP, HTTP, JDBC, etc. Una prueba de carga básicamente es medir el rendimiento de nuestra aplicación en diferentes escenarios extremos, hasta cuantos requerimientos soporta, la carga del CPU, la cantidad de memoria utilizada, etc.

Procedemos a crear la prueba con 50, 500 y 5000 conexiones simultaneas al servidor web.

Primero crearemos el nuevo plan de pruebas para lo cual seguimos los siguientes pasos como vemos en la Figura 4.26

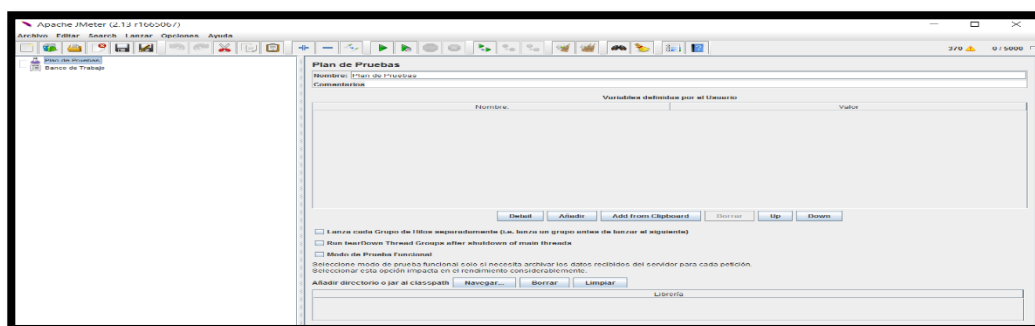


Figura 4.26: Creación del plan de pruebas en JMeter.

Segundo crearemos el grupo de hilos tal como se ve en la Figura 4.27 y en la Figura 4.28 nos muestra la configuración de la recurrencia y los tiempos de espera de la respuesta en el servidor.

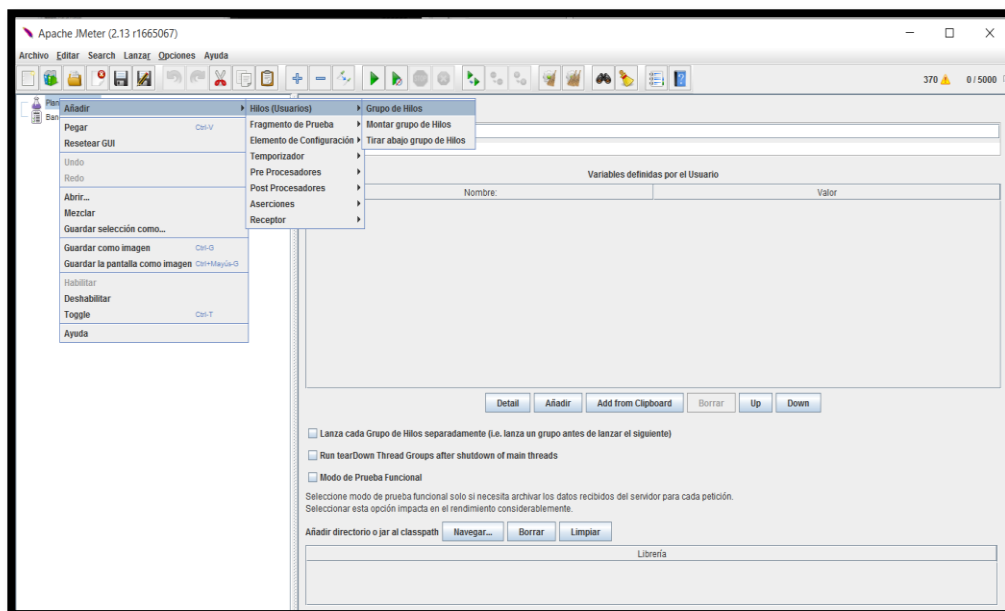


Figura 4.27: Creación del grupo de hilos en JMeter.

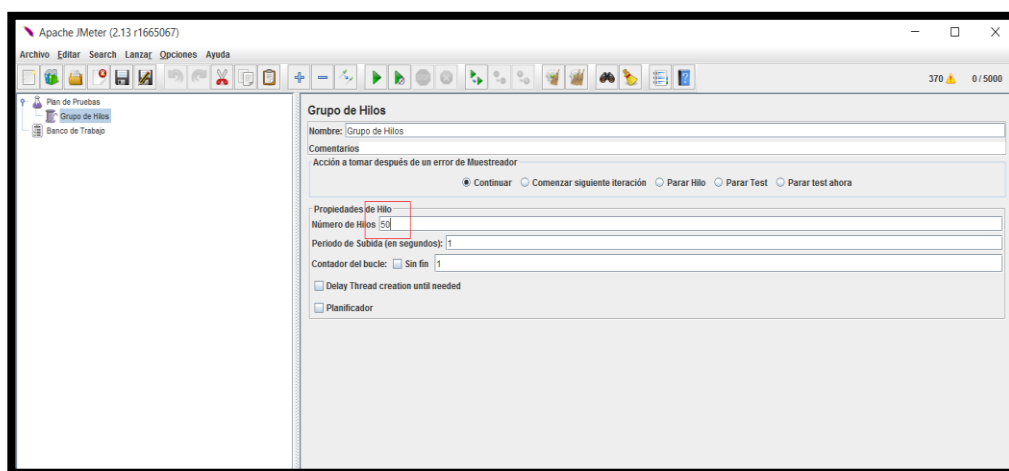


Figura 4.28: Parametrizar el grupo de hilos con un número de peticiones recurrentes.

Ahora configuraremos los hilos HTTP para enviar solicitudes al servidor, para esto deberemos parametrizar el muestreador de la siguiente manera, tal como se muestra en la Figura 4.29 y la Figura 4.30.

IP del servidor: 192.168.1.24

Puerto del servidor: 8080

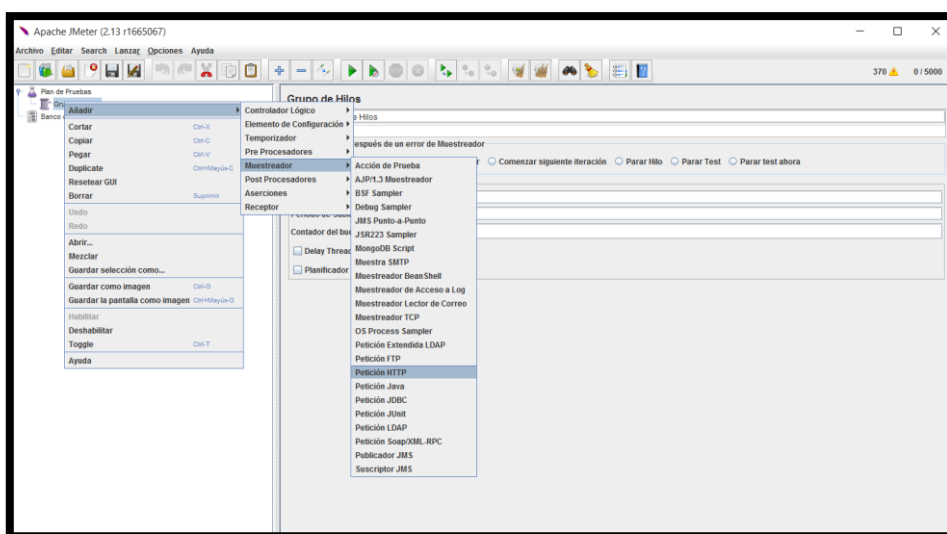


Figura 4.29: Creación del muestreador HTTP en JMeter.

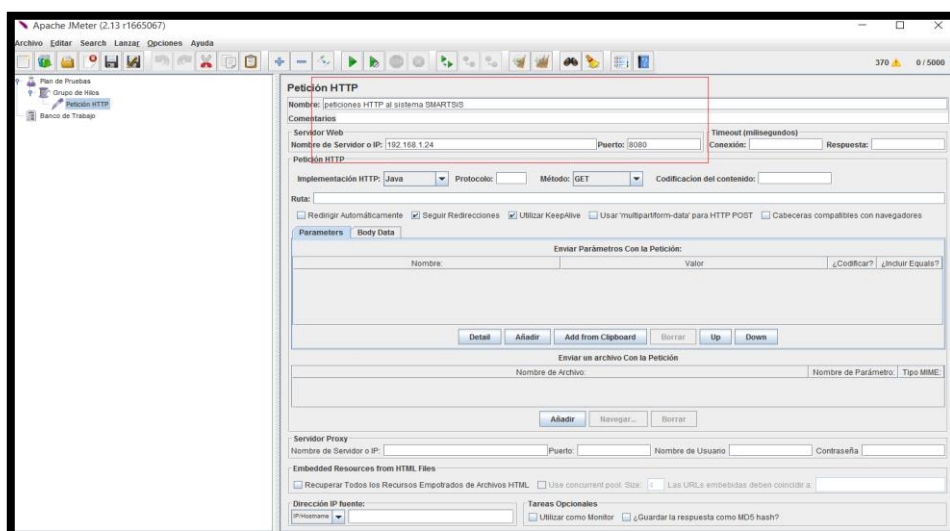


Figura 4.30: Parámetros del muestreador HTTP en JMeter.

Para finalizar configuraremos el árbol de resultados que nos mostrará de manera de árbol cada una de las respuestas obtenidas por la ejecución. Además de permitir ver los tiempos de retraso y los códigos de respuesta con su respectivos datos tal como se observa en la Figura 4.31.

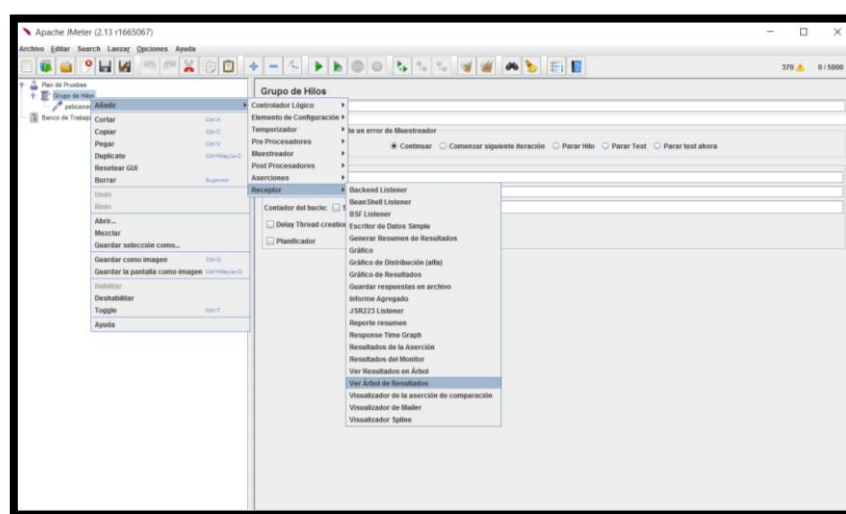


Figura 4.31: Crear el receptor del muestreador HTTP en JMeter.

Como podemos observar se obtuvieron buenos resultados de la prueba de estrés como lo detallamos a continuación.

Para las 50 conexiones simultáneas observamos que el servidor responde correctamente a las peticiones sin llegar a saturar el pool de conexiones, y así mismo para 500. Tal como vemos en la Figura 4.32.

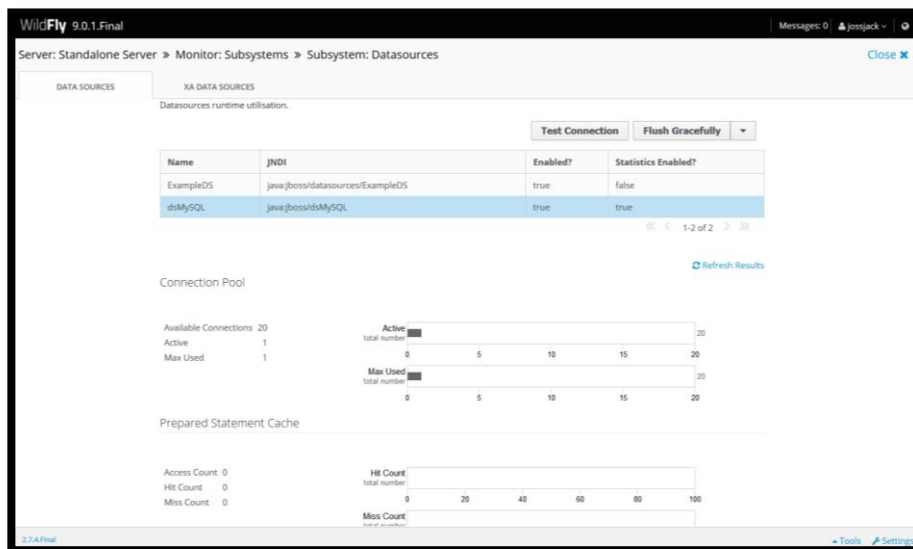


Figura 4.32: Pool de conexiones disponible del servidor Wildfly

El rendimiento del procesador y memoria del controlador a 50 y 500 conexiones HTTP nos muestra que efectivamente el tener configurado un pool de conexiones en un servidor de aplicaciones y procesos independientes sin hilos nos brinda un excelente rendimiento tal como se muestra en la Figura 4.33 y Figura 4.34.

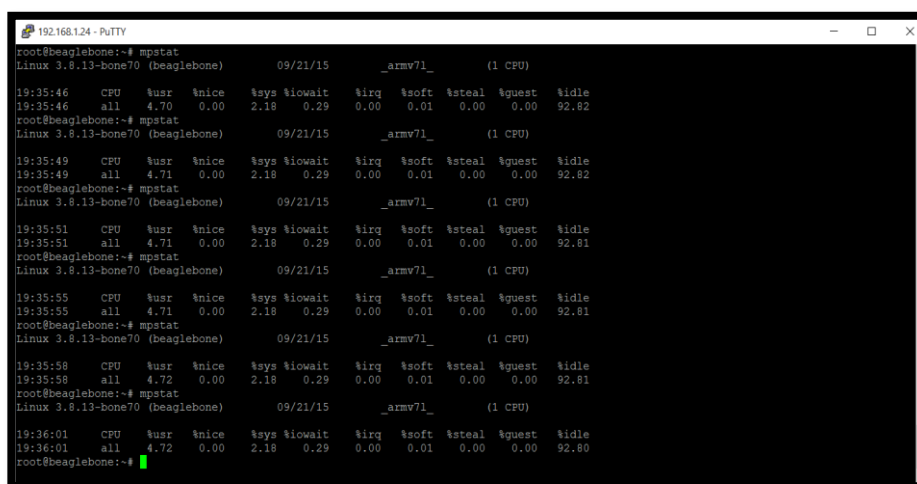


Figura 4.33: Revisión de la estadística del CPU del controlador mientras se ejecuta la prueba de estrés

```

192.168.1.24 - PuTTY
root 908 1.0 0.0 3160 488 ttyGS0 Ss+ Sep20 13:02 /sbin/agetty -s ttyGS0 115200 38400 9600
root 665 0.0 0.0 3340 476 tty1 Ss+ Sep20 0:00 /sbin/agetty tty1 38400
root 666 0.0 0.0 3160 476 tty00 Ss+ Sep20 0:00 /sbin/agetty -s tty00 115200 38400 9600
xrdp 786 0.0 0.0 4560 484 ? S Sep20 0:00 /usr/sbin/xrdp
root 626 0.0 0.0 1332 488 ? Ss Sep20 0:00 /usr/sbin/acpid
root 858 0.0 0.1 3356 600 ? S Sep20 0:00 /usr/sbin/xrdp-sesman
root 637 0.0 0.1 2632 700 ? Ss Sep20 0:00 /lib/systemd/systemd-logind
root 785 0.0 0.1 3380 728 ? Ss Sep20 0:01 /usr/sbin/cron
root 636 0.0 0.1 4604 760 ? Ss Sep20 0:00 /sbin/wpa_supplicant -u -s -O /var/run/wpa_supplicant
root 792 0.0 0.1 5160 804 ? Ss Sep20 0:00 /usr/sbin/sshd
avahi 617 0.0 0.1 2876 908 ? Ss Sep20 0:01 avahi-daemon: running [beaglebone.local]
root 24126 0.0 0.1 2484 908 pts/0 R+ 19:41 0:00 ps aux --sort pmem
root 912 0.0 0.1 2432 916 ? S Sep20 0:00 /sbin/udev
root 911 0.0 0.1 2432 940 ? S Sep20 0:01 /sbin/udev
102 628 0.2 0.1 2728 960 ? Ss Sep20 3:02 /usr/bin/dbus-daemon --system --address=systemd: --nofork --nopidfile --s
root 225 0.0 0.2 2436 1020 ? Ss Sep20 0:02 /sbin/udev
root 887 0.0 0.2 22960 1076 ? Ssl Sep20 0:00 /usr/lib/policykit-1/polkitd --no-debug
root 641 0.1 0.2 28408 1160 ? Ssl Sep20 1:35 /usr/sbin/rsyslogd -n -c5
root 23158 0.0 0.2 1256 pts/0 Ss 19:26 0:00 -bash
root 203 0.0 0.2 7536 1288 ? Ss Sep20 0:05 /lib/systemd/systemd-journald
root 640 0.0 0.2 25568 1304 ? Ssl Sep20 0:00 /usr/lib/upower/upowerd
root 23154 0.1 0.2 7848 1356 ? Ss 19:26 0:00 sshd: root@pts/0
root 639 0.0 0.3 25416 1652 ? Ssl Sep20 0:00 /usr/sbin/console-kit-daemon --no-daemon
root 1 0.0 0.3 4536 1780 ? Ss Sep20 0:01 /lib/systemd/systemd
root 1339 0.2 1.0 14668 5452 ? S Sep20 3:10 /usr/bin/python -O /usr/share/wicd/daemon/monitor.py
root 1220 0.5 1.2 23884 6472 ? S Sep20 7:11 /usr/bin/python -O /usr/share/wicd/daemon/wicd-daemon.py
root 634 0.0 1.2 64884 6556 ? Ssl Sep20 0:01 /usr/bin/node autorun.js
root 761 0.0 3.3 77488 16972 ? Ssl Sep20 0:11 /usr/bin/node server.js
root 635 0.0 4.0 36472 20512 ? Ssl Sep20 0:15 /usr/bin/ruby1.9.1 /usr/local/bin/jekyll build --destination bone101 --we
mysql 1444 0.6 6.6 315732 33648 ? s1 Sep20 7:29 /usr/sbin/mysqld --basedir=/usr --datadir=/var/lib/mysql --plugin-dir=/us
wildfly 1636 2.9 66.0 949704 335804 ? s1 Sep20 36:54 java -D[Standalone] -server -server -Xms64m -Xmx512m -XX:MaxPermSize=256m
root@beaglebone:~#

```

Figura 4.34: Revisión de la estadística de la memoria por proceso del controlador mientras se ejecuta la prueba de estrés

Ahora realizaremos una prueba de rendimiento general del servidor bajo 5000 conexiones HTTP, podemos observar que la máquina virtual JVM comienza a saturarse rápidamente a las 490 conexiones aproximadamente. Ver Figura 4.35.

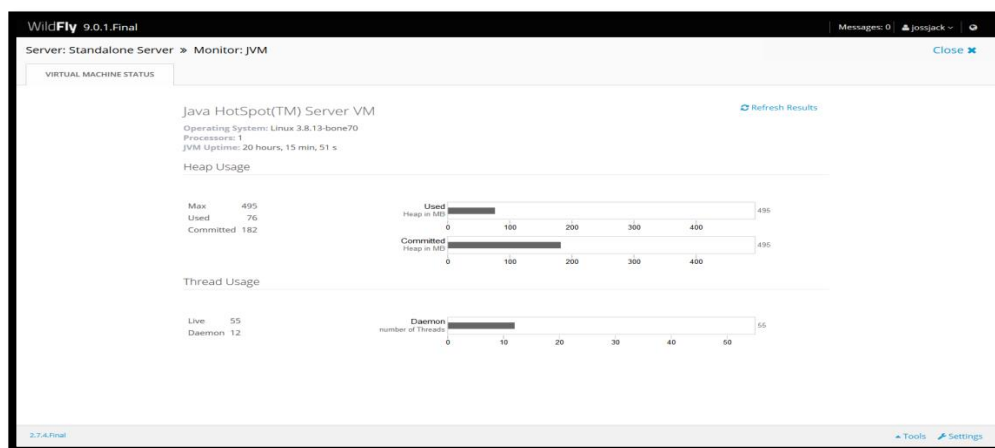


Figura 4.35: Estado general de la máquina virtual de Java bajo la prueba de estrés de 5000 conexiones HTTP.

Se muestra a continuación en la Figura 4.36 y la Figura 4.37 los datos obtenidos en el árbol de resultados del programa JMeter en el cual podemos observar los tiempos de retardo y en qué momento se pierde conexiones, las cuales aparecen de color rojo, esto sucede a partir de las 500 conexiones simultaneas.

Para evitar un problema de seguridad en nuestro servidor se configuro el pool de conexiones de hasta 50 conexiones simultáneas que son suficientes para el manejo de este sistema.

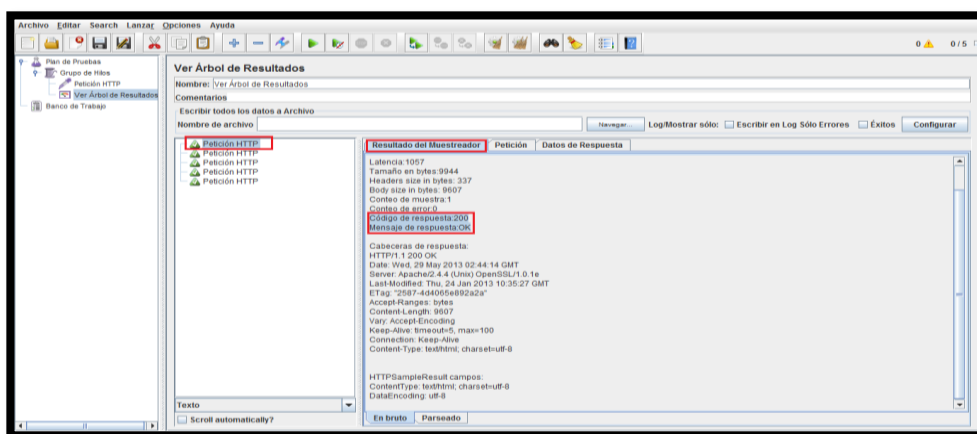


Figura 4.36: Árbol de resultados del programa JMeter bajo 5000 peticiones HTTP al servidor del controlador.

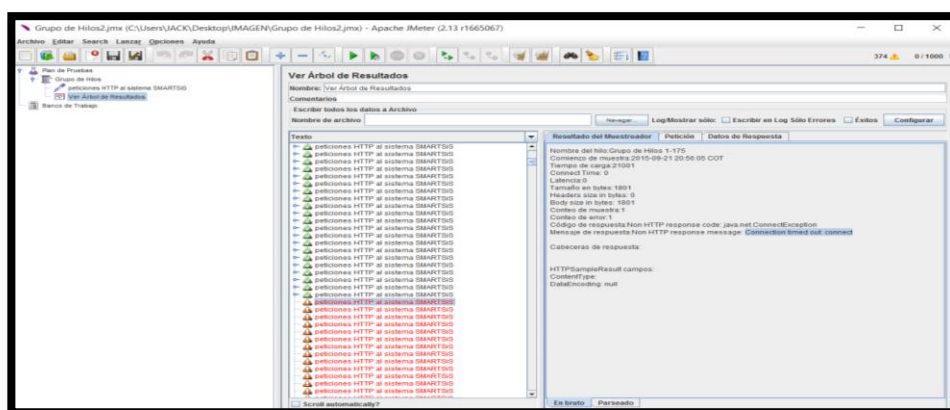


Figura 4.37: Árbol de resultados del programa JMeter en el cual vemos que a partir de la prueba 500 se pierde la conexión con el controlador.

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

1. El uso del controlador BeagleBone Black nos permitió obtener unos buenos resultados a la hora de implementar proyectos en los cuales se requiera de un equipo de cómputo de bajo costo y rápido procesamiento, a la vez que podemos crear numerosas aplicaciones en ambientes reales para solucionar problemas o necesidades.
2. El prototipo del sistema hidropónico NFT es un diseño muy sencillo y de bajo costo para implementar en nuestros campos o casas, debido a sus dimensiones en la que podemos cultivar de manera vertical y horizontal aprovechando espacios o zonas que pueden ser útiles para producir.
3. El cultivo hidropónico es una solución bastante fiable en la cual podemos cultivar de manera controlada nuestros productos, sin necesidad de depender de la tierra y las condiciones climáticas. Esto ayudará a combatir en muchas zonas rurales la desnutrición y muchas otras enfermedades producidas por químicos.
4. El agua es un recurso muy valioso y útil que si se sabe aprovechar puede ser muy beneficiosos como es el caso de este proyecto en el cual combinar la acuaponía y la hidroponía se obtiene un sistema robusto natural del cual las plantas se alimentan de los desechos de los peces a la vez que las raíces entregan oxígeno al agua.
5. Los cambios de uno o varios parámetros de la calidad del agua influyen en otros, dando como resultado una descompensación que se debe corregir según el análisis de los datos.
6. El uso de software libre como es Linux y java nos permite crear potentes herramientas de telemetría y telecontrol teniendo en cuenta el consumo del procesamiento y memoria. Esto es debido al uso correcto de normas y estándares de programación como algoritmos eficientes.

Recomendaciones

1. Se recomienda implementar el sistema en un ambiente abierto donde la luz solar incida directamente con las plantas y los peces de manera que se pueda producir la fotosíntesis y las raíces puedan absorber correctamente todos los desechos que producen los peces y reduciendo la tasa de intercambio del agua en al menos dos veces por mes.
2. Diseño de módulos que permitan correlacionar los parámetros de la calidad del agua de manera que se pueda ver cómo influye directa o indirectamente un parámetro sobre otro, para esto se puede hacer uso de la regresión lineal y la estadística multivariable.
3. Para un ambiente real de producción se recomienda tener peces como tilapia o trucha que son peces de mayor tamaño lo que nos dará una mayor proporción de desechos orgánicos que servirán para nutrir a nuestras plantas de manera eficiente.
4. Se debe tener la estructura hidropónica NFT bien sujeta a la superficie pues dependiente del tamaño y peso esta puede comenzar a moverse y provocar daños tanto en la estructura como en las plantas.
5. Los tomates suelen crecer mucho por lo que cuando lleguen a una etapa de madurez se debe de colocar soportes al tallo para que el peso del fruto no lo dañe y así este pueda seguir creciendo normalmente. Los tomates son productos que pueden soportar altos valores de salinidad que son buenos para su crecimiento y sabor, por lo que no es recomendable combinar este tipo de plantas con otras que no soporten estos niveles altos.
6. Este proyecto puede combinarse con nodos inalámbricos que pueden actuar como enrutadores y nodos finales que servirán para medir o activar dispositivos, permitiendo ampliar nuestra red de producción y alcance del proyecto.

BIBLIOGRAFÍA

Libros y Textos:

[1] Laura Alpízar Antillón, “Hidroponía, cultivo sin tierra”, 1th ed, Eds Tecnológica de Costa Rica, 2004.

[2] Papa Francisco, “Ecología ambiental, económica y social,” en Laudato Si, Tipografía vaticana, 2015.

[3] Sheila Guadalupe Rubio, “Análisis técnico de producción de tilapia *Oreochromis niloticus* y lechuga acropolis *Lactuca sativa* en acuaponía,” Tesis de Maestría, Dept. Recursos naturales y medio ambiente, Institu Politécnico Nacional, Cambridge, México, 2012.

[4] Beral Projects. (2014, Septiembre 16). ¿Qué es el ORP? [Online]. Disponible en: <http://www.outlet-piscinas.com>

[5] Sánchez del C y Escalante R, “Hidroponía un sistema de producción de plantas, principios y metodos de cultivo”, Imprenta Universitaria, Universidad Autónoma Chapingo, México, 1988.

[6] Cardenas F.J, V. González y J.M Hernández, “Cultivo protegido del tomate”, Edición agrotécnica, España, 2003.

[7] Hydro Enviroment. (2011, Octubre 4). Calidad del agua en la hidroponía [Online]. Disponible en: <http://www.hydroenv.com.mx>

[8] Humberto Estrada V. (2012, Agosto 22). La calidad del agua del riego [Online]. Disponible en: <http://www.smart-fertilizer.com>

[9] KLR. (2013, Abril 25). EC, TDS, PPM, FC: Conceptos en hidroponía [Online]. Disponible en: <http://www.elbruixot.com>

ANEXOS

RENDIMIENTO COMPARATIVO ENTRE EL CULTIVO TRADICIONAL Y EL HIDROPÓNICO, CALCULADOS EN TONELADAS POR AREA.

	HIDROPÓNICO		TRADICIONAL	
	PRODUCCION COSECHA	# DE COSECHAS AL AÑO	RENDIMIENTO TOTAL ANUAL	RENDIMIENTO TOTAL ANUAL
FRIJOL	11,5	4	46	6
REPOLLO	57,5	3	172,5	30
PEPINO	250	3	750	30
BERENJENA	28	2	56	20
LECHUGA	31,3	10	313	52
PIMENTON	32	3	96	16
TOMATE	187,5	2	375	100
FORRAJE	4Kg/m2/día	8		