

SISTEMA AUTOMATIZADO DE ADQUISICIÓN DE DATOS PARA SEGUIMIENTO Y
EVALUACIÓN DE PROCESOS DE ÓSMOSIS RETARDADA E INVERSA

ESTUDIANTES - INVESTIGADORES:

DAVID EDUARDO CERMEÑO PINZÓN



UNIVERSIDAD DE CARTAGENA

FACULTAD DE INGENIERÍA

PROGRAMA DE INGENIERÍA DE SISTEMAS

CARTAGENA DE INDIAS D. T. Y C., 2021.

SISTEMA AUTOMATIZADO DE ADQUISICIÓN DE DATOS PARA SEGUIMIENTO Y
EVALUACIÓN DE PROCESOS DE ÓSMOSIS RETARDADA E INVERSA

Instituto de Hidráulica y Saneamiento Ambiental

Grupo de investigación GIMATICA

Línea de investigación: Tecnologías de la Información y las Comunicaciones del
Semillero de Arduino – SEMARD

PROYECTO DE GRADO PRESENTADO COMO REQUISITO PARCIAL PARA
OPTAR AL TÍTULO DE INGENIERO DE SISTEMAS

DIRECTOR – INVESTIGADOR:

M. SC AMAURY CABARCAS ÁLVAREZ

ESTUDIANTE - INVESTIGADOR:

DAVID EDUARDO CERMEÑO PINZÓN



UNIVERSIDAD DE CARTAGENA

FACULTAD DE INGENIERÍA

PROGRAMA DE INGENIERÍA DE SISTEMAS

CARTAGENA DE INDIAS D. T. Y C., 2021.

AGRADECIMIENTOS

A Colciencias por apoyar el proyecto compuesto de 3 tesis de la Universidad de Cartagena, en los programas de Ingeniería Civil, de Ingeniería Química e Ingeniería de Sistemas brindando los rubros necesarios para su finalización.

A la Universidad de Cartagena, por presentarme muchos retos interesantes de crecimiento intelectual y personal durante este proceso; además de apoyar el espíritu investigativo del presente proyecto de investigación.

Al Instituto de Hidráulica por gestionar y guiar el proyecto de creación del sistema osmótico y dentro de este semillero a quienes hicieron parte activa: director del proyecto Edgar Quiñones, tesorera del proyecto Yolima, ingeniero civil Luis Vives, ingeniera civil Michelle Montero, ingeniero civil Juan, Asistente de laboratorios Remberto, todos muy excelentes compañeros de equipo y talentosos en sus áreas.

Al profesor Amaury Cabarcas, mi primer profesor en la universidad con la asignatura de algoritmos, mi director de semillero y director de proyecto de grado, muchas gracias por apoyarme durante todo mi transcurso en el semillero y su guía constante durante todo mi proceso académico y de proyecto de grado.

Gracias a mis padres por darme un espacio de apoyo desde donde poder esforzarme para realizar mis estudios y por su influencia en mi vida.

CONTENIDO

	Pág.
1. RESUMEN	11
2. ABSTRACT	11
3. INTRODUCCIÓN	13
4. PLANTEAMIENTO DEL PROBLEMA	14
4.1. <i>Descripción del problema</i>	14
4.2. <i>Formulación del Problema</i>	16
5. JUSTIFICACIÓN	17
6. OBJETIVOS	21
6.1. <i>Objetivo general</i>	21
6.2. <i>Objetivos específicos</i>	21
7. MARCO DE REFERENCIA	22
ESTADO DEL ARTE	22
MARCO TEÓRICO	25
8. METODOLOGÍA	31
9. RESULTADOS	39
10. CONCLUSIONES	86
11. RECOMENDACIONES	89
12. REFERENCIAS BIBLIOGRÁFICAS	90

FIGURAS

Pág.

Figura 1: Distribución de subestaciones en Colombia. (Ministerio de Minas y Energía, 2016)	17
<i>Figura 2: Boletín Estadístico de Minas y Energía. 2012 - 2016; según la unidad de planeación minero energética (UPME)</i> (Ministerio de Minas y Energía, 2016).	18
<i>Figura 3: Esquema de sistema de control.</i> (Roman, 2010)	27
<i>Figura 4: Esquema general de un sistema automatizado</i> (Roman, 2010)	32
<i>Figura 5: imagen de muestra del orden de las iteraciones del proyecto.</i>	33
<i>Figura 6: diagrama general de comunicación del sistema</i>	35
Figura 7: Tablero de la reunión donde se sentaron las bases del sistema y se explicó una propuesta base.	39
<i>Figura 8: foto de prueba de sensibilidad de posibles sensores propuestos durante la segunda reunión.</i>	40
<i>Figura 9: foto de la reunión donde se presentaron opciones de prototipos a implementar en los distintos puntos de toma de datos de las reuniones anteriores.</i>	41
<i>Figura 10: diagrama de casos de uso del sistema conjunto (físico y remoto).</i>	43
<i>Figura 11: diagrama de secuencia que muestra el workflow utilizado para realizar un inicio de sesión en el sistema remoto.</i>	44
Figura 12: diagrama de secuencia que evidencia el workflow utilizado para mostrar los datos en el sistema físico.	45
<i>Figura 13: diagrama de secuencia que muestra el workflow utilizado para comenzar la recolección de datos en el sistema físico y el remoto.</i>	47
Figura 14: diagrama de secuencia que evidencia el workflow utilizado para detener la toma de datos en el sistema remoto.	48

Figura 15: diagrama de secuencia que evidencia el workflow utilizado para mostrar los datos en el sistema remoto.....	49
<i>Figura 16: diagrama de secuencia que evidencia el workflow utilizado para exportar los datos guardados en el sistema remoto.....</i>	<i>50</i>
Figura 17: diagrama de componentes general de ambos sistemas, tanto el remoto como el físico en conjunto.....	51
<i>Figura 18: diagrama de despliegue que incluye tanto el sistema remoto como el físico.....</i>	<i>52</i>
<i>Figura 19: diagrama de general del sistema electrónico vista para usuarios e implementadores.....</i>	<i>53</i>
<i>Figura 20: diagrama representativo de composición del sistema, subsistema hidráulico.....</i>	<i>54</i>
<i>Figura 21: diagrama representativo de composición del sistema, subsistema de distribución de información.....</i>	<i>54</i>
<i>Figura 22: diagrama representativo de composición del sistema, porción de desglose del sistema remoto.....</i>	<i>55</i>
<i>Figura 23: diagrama de electrónico del sistema físico, conexión circuital del mismo.....</i>	<i>55</i>
<i>Figura 24: Propuesta de balanza simple vista aérea con superficie base.....</i>	<i>56</i>
<i>Figura 25: Propuesta de balanza simple vista aérea sin superficie base.....</i>	<i>56</i>
<i>Figura 26: Propuesta de balanza solo sensores de fuerza y base elevador que permita la Torsión.....</i>	<i>57</i>
<i>Figura 27: Plataforma web vista de acceso al sistema, también llamada vista de login, donde se muestra la estructura base de la página con su formulario e infografía base.....</i>	<i>58</i>

<i>Figura 28: Plataforma web vista de dashboard, donde se muestran las gráficas que están evaluando los involucrados en el proyecto.</i>	58
<i>Figura 29: Plataforma web vista de los sets/conjuntos de medidas tomados en el tiempo.....</i>	59
<i>Figura 30: Plataforma web vista de los sets medidas, donde se muestran las medidas los momentos base donde se tomaron datos y se ofrece la posibilidad de verlos detalladamente.....</i>	59
<i>Figura 31: Plataforma web vista de administrador de máquinas con el formulario de creación y actualización de las mismas.</i>	60
<i>Figura 32: Plataforma web vista de administrador de máquinas con la tabla del listado de máquinas registradas que inicia la eliminación o actualización de las mismos. .</i>	60
<i>Figura 33: Plataforma web vista de administrador de usuarios con el formulario de creación y actualización de los mismos</i>	61
<i>Figura 34: Plataforma web vista de administrador de usuarios con la tabla con el listado de usuarios registrados que inicia la eliminación o actualización de los mismos. ...</i>	61
<i>Figura 35: Plataforma web vista de configuraciones para actualizar la información del usuario actualmente logeado.</i>	62
<i>Figura 36: Foto del sistema físico implementado con el administrador del laboratorio Reberto Avila y el estudiante David Eduardo Cermeño Pinzón tomando medidas.....</i>	63
<i>Figura 37: figura de primeras implementaciones del sistema (iteración 3) de recolección de datos en el laboratorio.</i>	64
<i>Figura 38: foto de primeras implementaciones del sistema (en iteración 3) de recolección de datos en el laboratorio.</i>	65
<i>Figura 39: foto de primeras implementaciones del sistema de recolección de datos (en iteración 10) en el laboratorio.....</i>	66

<i>Figura 40: foto de las implementaciones en etapa mediana del sistema de recolección de datos (en iteración 10) en el laboratorio.</i>	67
<i>Figura 41: foto de las implementaciones en etapa mediana del sistema de recolección de datos (en iteración 10) en el laboratorio.</i>	68
<i>Figura 42: foto de las implementaciones en etapa mediana del sistema de recolección de datos (en iteración 10) en el laboratorio.</i>	69
<i>Figura 43: foto de las implementaciones en etapa mediana del sistema de recolección de datos (en iteración 10) en el laboratorio.</i>	70
<i>Figura 44: foto de las implementaciones en etapa mediana del sistema de recolección de datos (en iteración 10) en el laboratorio.</i>	71
<i>Figura 45: foto de las implementaciones en etapa mediana del sistema de recolección de datos (en iteración 10) en el laboratorio.</i>	72
<i>Figura 46: foto de las implementaciones en etapa mediana del sistema de recolección de datos (en iteración 10) en el laboratorio.</i>	73
<i>Figura 47: foto de las implementaciones en etapa de finalización del sistema de recolección de datos (en iteración 10) en el laboratorio.</i>	74
<i>Figura 48: foto de las implementaciones en etapa mediano avanzada del sistema de recolección de datos (en iteración 11) en el laboratorio.</i>	75
<i>Figura 49: foto de las implementaciones en etapa avanzada del sistema de recolección de datos (en iteración 12) en el laboratorio.</i>	76
<i>Figura 50: foto de las implementaciones en etapa avanzada del sistema de recolección de datos (iteración 15) en el laboratorio.</i>	77
<i>Figura 51: foto de las implementaciones en etapa avanzada del sistema de recolección de datos (en iteración 16) en el laboratorio.</i>	78

<i>Figura 52: foto de las implementaciones en etapa avanzada del sistema de recolección de datos (en iteración 16) en el laboratorio.</i>	79
Figura 53: foto de las implementaciones en etapa avanzada del sistema de recolección de datos (en iteración 16) en el laboratorio.	80
<i>Figura 54: foto de las implementaciones en etapa avanzada del sistema de recolección de datos (en iteración 16) en el laboratorio.</i>	81
<i>Figura 55: foto de las implementaciones en etapa avanzada del sistema de recolección de datos (en iteración 16) en el laboratorio.</i>	82
<i>Figura 56: foto de las implementaciones en etapa avanzada del sistema de recolección de datos (en iteración 16) en el laboratorio.</i>	83
<i>Figura 57: foto de las implementaciones en etapa avanzada del sistema de recolección de datos (en iteración 16) en el laboratorio.</i>	84

TABLAS

Pág.

Tabla 1: requisito funcional “Recolectar datos de los sensores” presente en el anexo 1 documento de especificación de requisitos	41
Tabla 2: requisito funcional “Recolectar datos de los sensores” presente en el anexo 1 documento de especificación de requisitos.	41
Tabla 3: requisito funcional “Recolectar datos de los sensores” presente en el anexo 1 documento de especificación de requisitos.	42

1. RESUMEN

Los procesos físicos y químicos modelados e investigados a través de prototipos usualmente son más fáciles de seguir cuando la tarea de recolección de datos es simplificada por algún sistema de toma de datos en el mismo, lo anterior toma mucha más importancia en procesos complejos de implementar y/o monitorizar como el de la ósmosis.

Por lo anterior, se formuló el presente con el objetivo realizado de establecer un sistema de recolección de datos para el prototipo que opera bajo procesos de Ósmosis Inversa y Ósmosis por Presión Retardada del programa de Ingeniería de Civil de la Universidad de Cartagena en su semillero adscrito GIMA usando Hardware libre, C++ como lenguaje para el sistema embebido y tecnologías y lenguajes programación frecuentemente utilizados para desarrollos de aplicaciones web en tiempo real (MongoDB, NodeJS, React), usando los protocolos necesarios para las conexiones con los sensores y las pantalla en el sistema embebido desde el microcontrolador; además, del uso de un protocolo de conexión por paquetes ligeros para MQTT en la conexión del sistema embebido y el sistema remoto.

Lo que infiere que como producto final se llegó a la realización de dos sistemas una plataforma para la recepción de los datos (ubicado en el VPS del servidor del semillero SEMARD) y el desarrollo de un sistema embebido de recolección de datos, que al ser analizados permite concluir que, aunque la Open Hardware permite crear un sistema de recolección de datos adecuado dadas las presiones encontradas en el proceso osmótico y la alta sensibilidad de la lectura de la variación de masa durante el proceso, la optimización de este requiere de mucha parametrización (que puede tomar mucho tiempo) razón por la que se tiende a elegir la tecnología propietaria para agilizar desarrollos.

2. ABSTRACT

The researched and modeled through machine prototypes chemical and physical processes are usually easier to follow when the data collection task is simplified via a data collector system implemented on it, which takes a lot more relevance when these processes are complex to implement (in a structural way) and/or monitor such as the osmosis process.

Because of the explained, the present had objective of implementing a data collection system for the prototype that works under the processes of Reverse Osmosis and Pressure Retarded Osmosis located in the GIMA research group ascribed to the Civil's Engineering faculty of Universidad de Cartagena, using Open Hardware, C++ as language for the embedded

system and technologies and programming languages frequently used to develop Real Time Web Apps (MongoDB, NodeJS, React), using the adequate protocols for the sensors and TFT connections to the microcontroller, along side the usage of a light weight package protocol to connect the embedded system to the remote system.

Of which can be inferred that the final product are two systems one for remote data collection (placed in SEMARD research group's VPS) and one for the local recollection and displaying for people in place, which when analyzed allows to draw the conclusion that even though Open Hardware allows to create an adequate data collection system given the found pressure precision ranges in the osmotic process and the sensitivity of the measurements of mass variation during the process, the setup of this requires a lot of calibrations and parametrizations (which can take a lot of time) reason why most of the time proprietary technology tend to be selected to speed developments.

3. INTRODUCCIÓN

A nivel de industria y en las invenciones frutos de la investigación humana se encuentran muchos procesos y sistemas difíciles de supervisar, algunos manejan complejidades inherentes al sistema por sus propiedades físicas y químicas, que son altamente susceptibles a errores ante pequeñas irregularidades en su medición.

En la Universidad de Cartagena se encuentra en desarrollo uno de estos sistemas, éste se encarga de producir agua potable a partir del agua salada funcionando bajo el proceso de ósmosis inversa y generar energía eléctrica operando bajo la ósmosis por presión retardada (PRO – Pressure Retarded Osmosis). Este segundo modo de funcionamiento se considera como energía renovable desde aproximadamente el año 2000 (Achilli & Childress, 2010).

El proyecto mencionado está en etapa avanzada y al involucrar ambos modos de funcionamiento en el mismo sistema se requiere de un alto nivel de parametrización para hacer óptimo el funcionamiento general del sistema osmótico y lograr un control de su regulación en el tiempo, esto último dependiendo actualmente de variar la posición de válvulas con base en la cantidad de información que se pueda reunir del sistema en corto tiempo para afectar las condiciones internas del sistema osmótico.

El presente documento enuncia la realización de un proyecto de grado que acorde al grupo de investigación GIMATICA y soportado en la línea de investigación temática de ingeniería de software aprobada por la facultad, ya que permite un nuevo punto de enfoque para la ingeniería de software dentro del ámbito de la energías limpias, centrado en la regulación (tal vez automática en tiempo futuro) de los componentes físicos que permiten la mayor obtención de la energía específica (en este caso la energía por presión de ósmosis retardada); para así dar muestra de la creación de un sistema automatizado, que se encargue de la administración de los datos asociados a los procesos osmóticos antes mencionados usando las tecnologías software y hardware apropiadas para el nivel de sensibilidad y rapidez que se necesita en la toma y muestra de datos.

Logrando ayudar a los encargados de operar la máquina a obtener información precisa para la configuración preliminar y mantenimiento en el tiempo del sistema osmótico, aspecto con el que tuvieron problemas y que no permitía avances significativos en cortos periodos de tiempo, lo anterior se realiza en el laboratorio de hidráulica de la sede Piedra de Bolívar en la Universidad de Cartagena (lugar donde se encuentra el prototipo mencionado).

4. PLANTEAMIENTO DEL PROBLEMA

4.1. Descripción del problema

El grupo de investigación GIMA del programa de Ingeniería Civil de la Universidad de Cartagena, contaba con un proyecto financiado por Colciencias con el nombre “GENERACIÓN DE ENERGÍA ELÉCTRICA A PARTIR DE LA DESALINIZACIÓN DE AGUAS CIANURADAS Y MARINAS MEDIANTE PROCESOS DE OSMÓTICOS” en la línea temática “SISTEMAS PARA SUMINISTRO PERMANENTE DE AGUA POTABLE Y ENERGÍA A BAJO COSTO” de la convocatoria para proyectos de investigación y desarrollo en ingeniería del año 2015. Entre los objetivos de este proyecto se encontraba la puesta en funcionamiento de un sistema osmótico que, aprovechando la presión por ósmosis retardada aplicada sobre una turbina interna, pueda generar la mayor cantidad de energía eléctrica posible. Esta presión por ósmosis retardada (PRO) no es más que un proceso que: “se basa en el aprovechamiento de diferencias en las concentraciones salinas de soluciones separadas por una membrana semipermeable para generar un flujo de agua constante. Cuando esto sucede, el agua se difundirá desde el lado de la solución de alimentación en el lado de la solución de empate que es el lado de agua de mar que está presurizado.” (Vives, 2017).

A este sistema osmótico además le fue implementado un modo de funcionamiento en el que usa la ósmosis inversa para producir agua potable, este tipo de proceso es también llamado RO y se diferencia de la ósmosis normal, en que incluye una presión sobre el área del sistema que posee el agua de mayor concentración, para generar un flujo a través de la membrana al área con líquido menos concentrado, produciendo así el tipo de agua que se desea.

El uso de los dos procesos en un mismo prototipo no es común, ya que cada uno tiene un objetivo específico; sin embargo, el sistema osmótico ya que opera bajo ambos procesos crea la posibilidad de investigación para el aprovechamiento de ambos procesos en conjunto, lo que supone la creación de conocimiento importante (que explicaría la alta pertinencia científica de la propuesta). La mejora del prototipo dependerá entonces de cuanta información se pueda extraer del sistema en funcionamiento.

Para hacerse una idea de los puntos de variación que hay que medir en estos sistemas para obtener información mínima viable hay que tener en cuenta que los procesos osmóticos dependen de una alta cantidad de variables presentes en diferentes puntos del sistema de los cuales se debe obtener un comportamiento (variaciones de temperaturas, de presiones, cambios

en el flujo de agua, medición de cantidad de energía generada, cantidad de flujo de agua potable obtenida), de tal forma que se le pueda realizar seguimiento y evaluación mínimos a los procesos internos de la máquina, siempre partiendo de la exactitud de los datos que las medidas obtenidas deben poseer.

Algunos puntos específicos a monitorear y que fueron obtenidos por medio del levantamiento de requisitos de las distintas reuniones con los encargados del proyecto incluyen:

- Entrada de agua salinizada a la membrana.
- Salida de agua salinizada.
- Entrada de agua con bajo índice de salinidad a la membrana.
- Salida de agua con bajo índice de salinidad a la membrana.
- Alrededor de la turbina generadora.
- Salida del sistema en la producción de agua con bajo índice de salinidad a la membrana.

Pero debido a la falta en su momento de un método fácil o sistema integrado que permita obtener información sobre los sistemas osmóticos se dificulta el proceso de recolección, pues usando sensores de tipo industrial puestos entre las tuberías (método utilizado actualmente por los encargados del prototipo) se reduce la cantidad de medidas que se pueden hacer en tiempo real (o cercano).

Otro factor problemático, era el esfuerzo por parte de los encargados de operar la máquina que esta metodología de medición puede producir, ya que no solo tendrían que centrarse en obtener la mayor cantidad de datos posibles del sistema para obtener la trazabilidad del proceso en cada instante de tiempo, sino que también deben hacer luego el proceso de análisis de lo obtenido manualmente, lo que puede llevar fácilmente a conclusiones de investigación erróneas y generación de conocimiento no veraz.

Por todo lo anterior la presente muestra la creación de un sistema como este centralizado, de tal forma que se le pueda dotar al encargado de la información que necesita para operar el sistema de manera apropiada y dándole la posibilidad de mejorar los procesos acordes a la trazabilidad de datos que necesitan.

Viendo específicamente a las comunidades y con ello un ámbito de pertinencia social, este proyecto sustenta las bases de un sistema, que al ser instalado cerca de pueblos alejados de acueductos o líneas eléctricas puede proveer a estos de ambos servicios, siempre que estos se

encuentren medianamente cerca de una costa; además, brinda la posibilidad a los países de obtener más medios para satisfacer la necesidad energética y/o de agua potable de la población, incluyendo (luego de optimizados los procesos) plantas de generación de energías limpias y/o agua para consumo humano basadas en ósmosis.

4.2. Formulación del Problema

¿Cómo realizar el seguimiento y evaluación de variables asociadas a los procesos en una máquina que opera bajo procesos de ósmosis retardada e inversa?

5. JUSTIFICACIÓN

El proyecto plantea el diseño e implementación de un sistema automatizado para el seguimiento de variables en una máquina de ósmosis inversa y por presión retardada, orientada a la generación de energía que aprovecha el encuentro de aguas salinas y dulces y que por las características hídricas de Colombia resulta ser una oportunidad ideal para su desarrollo, esta investigación situaría a Colombia como un precursor de esta tecnología en América Latina por lo que hasta hace dos años no existía ninguna máquina que utilizará estos procesos para la generación de energía en ella, a pesar de su uso en otros países como Canadá.

Un punto importante a tener en cuenta, es que en Colombia y el mundo la generación de la energía se maneja a través de distintos flujos eléctricos que vienen de generadoras, esa gran cantidad de flujos energéticos son complementarios entre ellos y mantienen la estabilidad eléctrica del país. Así como en otros países, en Colombia la carga de la generación de energía está distribuida entre muchas subestaciones en el territorio nacional como lo muestra la unidad de planeación minero energética de Colombia (UPME) (Ministerio de Minas y Energía, 2016).

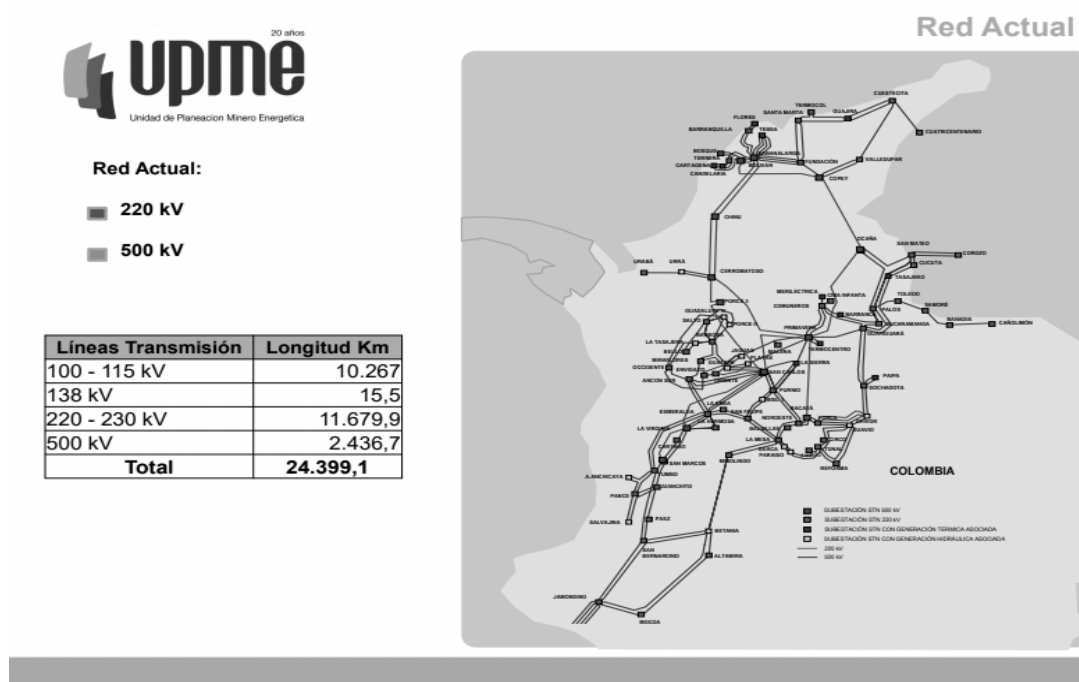


Figura 1: Distribución de subestaciones en Colombia. (Ministerio de Minas y Energía, 2016)

Según el boletín estadístico de minas y energía en el apartado de “Capacidad efectiva de energía eléctrica”, se muestra el aporte del tipo de energía que se da por departamento en el territorio colombiano, ver figura 2 (Ministerio de Minas y Energía, 2016):

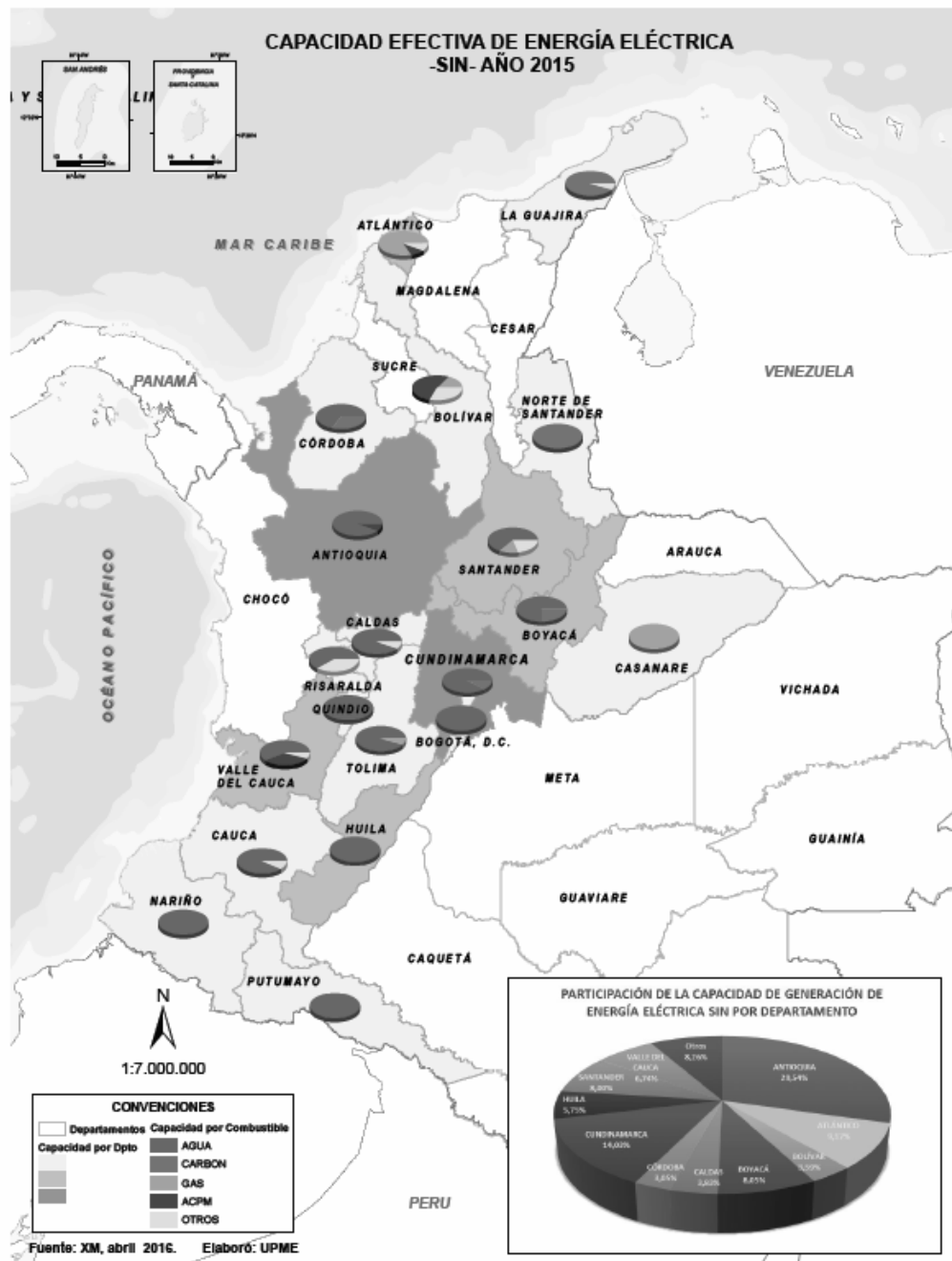


Figura 2: Boletín Estadístico de Minas y Energía. 2012 - 2016; según la unidad de planeación minero energética (UPME) (Ministerio de Minas y Energía, 2016).

Como se puede apreciar en la figura 1 la mayor cantidad es producida por la energía hidráulica, una energía muy estable que no está determinada por factores situacionales como las corrientes de viento o la cantidad de sol, si no que más bien está limitada por la distribución y capacidad hídrica del lugar donde se va a implementar (Kádár, 2014). Cada método de energía

renovable es aplicado a partir de estudios previos sobre un área particular, para ver si este posee las características necesarias para sacar un flujo constante de energía aprovechable.

En Colombia esta utilización de recursos hídricos es razonable, partiendo de la gran cantidad que posee Colombia tanto de agua salada como dulce; de hecho, según la GWP (Global Water Partnership), el continente cuenta con la mayor cantidad de agua dulce en el mundo, que equivale casi un tercio de los recursos hídricos renovables de la tierra (Redacción Actualidad, 2015) y según informa la revista semana “Colombia es 50% mar” gracias a su ubicación geográfica que le permite tener costas sobre el océano pacífico y el mar caribe. A pesar de estas condiciones ventajosas no sólo en Colombia, sino a nivel de América Latina, no existe conocimiento de ninguna máquina de ósmosis, lo que significa que de crearse una en la Universidad de Cartagena, está sería la primera. El hecho de implementar un sistema osmótico en un país con tan propicias características hídricas, supone una gran oportunidad de investigación en términos de energías renovables para Colombia.

Por lo anterior este proyecto propuso que la máquina de ósmosis tenga a disposición un sistema automatizado de toma y tratamiento de datos en tiempo real con trazabilidad de datos históricos, que permitiera facilitar su parametrización, adaptación y supervisión, permitiendo la posibilidad de mejoras de rendimiento sustentables y evaluables de manera fácil, gracias a la cantidad de datos pertinentes que proporcionan esta clase de sistemas, siendo estas mejoras fruto de la evaluación y análisis de los encargados de operar y dar seguimiento a la máquina.

Ahora bien, para la ingeniería esta investigación enriquecería el material disponible sobre esta clase de tecnologías y supondría una base para futuras investigaciones de esta clase de energía renovable a nivel institucional, de Colombia y/o América Latina, producto de las conclusiones y mejoras que se le apliquen al sistema con base en la información obtenida presentada por el sistema automatizado.

El programa Ingeniería de Sistemas fomenta el desarrollo tecnológico de sus estudiantes tanto en materia de hardware y software en sus asignaturas de Electrónica, Sistemas Digitales, Organización de Computadores, redes y comunicaciones en sus asignaturas Redes I, Redes II e ingeniería de software, siendo coherente la formación recibida como estudiante para el desarrollo que se llevó a cabo en el proyecto.

Al momento el final del año 2020, esta investigación tiene el apoyo y financiamiento de Colciencias con un presupuesto establecido de \$60.000.000 de los cuales \$10'000.000 han sido destinados para la implementación del sistema automatizado, destinados para la adquisición de

hardware libre y software abierto, que en esencia es tecnología altamente accesible a nivel local e internacional, por su alta presencia en los locales de productos electrónicos, tiendas online y por su bajo costo económico (en ocasiones costo nulo, hablándose específicamente del software libre).

6. OBJETIVOS

6.1. Objetivo general

Desarrollar un sistema automatizado de adquisición de datos para el seguimiento y evaluación de las variables presentes en los procesos de ósmosis retardada e inversa, haciendo uso de hardware y software de código abierto.

6.2. Objetivos específicos

- Especificar los requerimientos que involucra la creación de un sistema de monitoreo de variables osmóticas a nivel de software y hardware que permita elaborar el diseño de un backlog completo del sistema.
- Diseñar un modelo de supervisión de variables osmóticas que se ajuste a las necesidades rudimentarias de uso de los encargados de operar el sistema de generación eléctrica por presión osmótica retardada y que puede operar bajo procesos de ósmosis inversa de la Universidad de Cartagena.
- Desarrollar un prototipo funcional que permita a los encargados de operar y dar seguimiento al sistema osmótico de la Universidad de Cartagena obtener mediciones precisas de los procesos internos en la misma.
- Evaluar los resultados obtenidos del prototipo a través de una prueba piloto que permita realizar el control de calidad pertinente para el mejoramiento del sistema.

7. MARCO DE REFERENCIA

ESTADO DEL ARTE

Los sistemas de automatización abarcan desde aplicativos para simplificar pequeñas tareas hasta grandes sistemas industriales para automatizar procesos en complejos entornos, con énfasis en este último desde la época de la industrialización ha tenido un auge importante.

Como referentes de los sistemas antes mencionados se encuentran sistemas de planificación de secuencias de inicio para generadores en sistemas eléctricos de potencia (Aleagaloaiza et al., 2015), el uso de sistemas de supervisión y adquisición de datos para la automatización de procesos en un centro de ingeniería genética y biotecnología (Bárzaga Martell et al., 2016), sistema automatizado para la vigilancia epidemiológica de la tuberculosis (Gelabert Machín & Pérez González, Edilberto. Ochoa Armas, 1995), Sistema automatizado para la organización racional del proceso cosecha-transporte de cereales (Yanoy et al., 2014) y que al igual que otros solo presentan el diseño, siendo un ejemplo de este el diseño de un sistema automatizado para la planta de alimento ensilado "Héctor Molina" (Silva-Díaz et al., 2017).

Como se puede apreciar son diversos los sistemas en este ámbito de la automatización, pero cada uno está llamado a cumplir una función o serie de ellas, misma para la que fue diseñado y que nace en base a una necesidad.

Para el caso de la ósmosis por presión retardada y como afirma Sebastián Delgado en su discurso de energías renovables acerca de la energía Azul (energía por presión osmótica retardada) es “una de las energías renovables muy poco conocida, por no estar ampliamente comercializada a nivel industrial, pero con grandes posibilidades de futuro, es la energía osmótica.” (Delgado, 2017).

Anexado el mismo devala que: “si se tiene en cuenta que el caudal total de agua de río vertida al mar en el planeta es del orden de 37300 km³/año, y si se aprovechara el 10% de este caudal para producir energía se obtendrían unos 1370 TWh/año, equivalente al consumo eléctrico de unos 520 millones de personas y un ahorro de emisión de CO₂ a la atmósfera de más de mil millones de toneladas/año (Logan & Elimelech, 2012); (I. L. Alsvik. Norwegian University of Science and Technology (www.ntnu.no))” (Delgado, 2017).

Lo anterior se evidencia en los distintos avances presentes en los muchos artículos científicos publicados en revistas de investigación como: “Experimental investigation of

pressure retarded ósmosis for renewable energy conversion: Towards increased net power” (Maisonneuve et al., 2016), Thin-Film Composite Pressure Retarded Ósmosis Membranes for Sustainable Power Generation from Salinity Gradients (Yip et al., 2011), natural organic matter fouling in pressure retarded ósmosis (Thelin et al., 2013), pero estos son dirigidos hacia mejoras en la estructura misma del proceso (siendo las adaptaciones en las membranas las más recurrentes por la dependencia del proceso a el tipo de membrana utilizada).

Los sistemas de ósmosis inversa y ósmosis directa presentan diversos sistemas automatizados, muchos enfocados en trabajar bajo condiciones industriales, generalmente diseñados para plantas de desalinización de agua principal uso del proceso de ósmosis inversa en la industria, ejemplos de estos serían: la compilación y análisis de distintos prototipos de sistemas automatizados para plantas de desalinización (Kim et al., 2009) y el uso de sistemas computacionales para automatizar y controlar procesos de plantas de ósmosis inversa (Zilouchian & Jafar, 2001), que sí bien es una buena alternativa para el manejo autónomo de la ósmosis inversa no presenta la posibilidad de avances que puedan significar el trabajo en simbiosis de ambos procesos.

Pero esta clase de proyectos tendían a utilizar tecnologías más estandarizadas de control como PLCs para sus implementaciones y era poco común hasta años recientes el uso de hardware libre (un término bastante actual), que centra su enfoque las tecnologías Arduino “Like”, para implementaciones de investigación como el proyecto que utiliza Arduino en procesos de forward ósmosis para la toma de datos (Bogler et al., 2019) y la evaluación de la concentración de “Volatile Fatty Acids” en aguas residuales por ósmosis directa (Blandin et al., 2019) también usando microcontroladores libres para manejar la información que se está generando; que aunque más frecuentes siguen siendo escasos y todavía deben ser explorados para esta clase de procesos.

Hablando específicamente de los proyectos que incluyen el proceso de ósmosis por presión retardada y el uso de hardware libre para su automatización, sus ocurrencias son todavía pocas, más aún con dispositivos altamente recientemente y disponibles que tienden a tener mejores características base conforme avanza el tiempo en aspectos como el poder de computación, IO pins, conectividad incluida (WiFi, Bluetooth), entre otros.

A ejemplo de los anteriores, se encontraron proyectos como el que une el proceso de PRO y el microcontrolador Arduino AtMega para el estudio del “Biofouling” en sistemas de filtración de membranas, que abarca por la amplitud abarca distintos procesos de ósmosis

(incluida la ósmosis por presión retardada), pero que necesita de un computador cercano al cual el microcontrolador hará el *data logging* (Suwarno et al., 2018) o el proyecto acerca de “membrane distillation”, que también utiliza la tecnología Arduino para recuperación de datos.

Con lo anterior y dado el transcurso de la tecnología Arduino es normal favorecer su uso en el mundo del hardware libre, pero cuando la complejidad de los proyectos va incrementando como en la lectura de diversos sensores, sumado a una presentación adecuada y sin requerir a computadores en el lugar físicos por requerimiento explícito, existe la posibilidad de sopesar estándar por hardware más potente, aunque relativamente nuevo, como es el caso de la tecnología ExpressIf en sus módulos Esp32 y Esp8266 para los que no se encontraron aplicaciones en proyectos concluidos para proyectos relacionados con la ósmosis.

En Colombia, no hay una vasta cantidad de proyectos que utilicen el proceso, aún menores relacionados con automatizaciones computacionales que propongan metodologías de recolección de información usando el software y hardware actual, pero hay ocurrencias de proyectos relacionados; casos particulares de la automatización de este proceso en Colombia incluyen:

- la reciente investigación que abarca el río de Magdalena como un potencial lugar para planta de energía utilizando PRO (Salamanca et al., 2019)
- el prototipo general para conectar plantas descentralizadas de energía renovable basado en prototipos electrónico de toma de datos y computador que mencionaba el proceso osmótico como un objetivo futuro (Jucá et al., 2011).

En la Universidad de Cartagena, sin contar el presente estuvieron activos 3 tesis que favorecen la investigación del proceso, bajo un proyecto adscrito a Colciencias a cargo del profesor Edgar Quiñones, en relación con ellas la presente se enfoca en la automatización de la toma de datos del mismo.

MARCO TEÓRICO

- **PROCESOS OSMÓTICOS**

Actualmente los procesos osmóticos a base de membranas poliméricas se presentan como posibles soluciones para problemáticas básicas alrededor del mundo como fuentes eficientes de agua para el consumo humano y la generación de energía eléctrica con un mínimo impacto al medio ambiente. Por esto son muchos los estudios que se adelantan con el fin de mejorar las características tales como aumentar su resistencia a las presiones, mejores soluciones de drenaje, la retención de las partículas salinas y la disminución del efecto de polarización de dichas membranas semipermeables que a su vez optimizaría los procesos de ósmosis inversa (RO) y la ósmosis directa (FO) utilizadas para la desalinización y la ósmosis de presión retardada para la generación de energía eléctrica. (Klaysom et al., 2013).

- **ENERGÍA OSMÓTICA**

Para entender este fenómeno se puede considerar como un proceso inverso de la energía de separación que utilizan las plantas desalinizadoras, es decir en lugar de utilizar energía para separar una mezcla en sus elementos constitutivos, la energía en este caso se libera cuando los elementos se combinan. Cuando dos soluciones con diferente grado de salinidad (gradiente de salinidad) se mezclan, se libera lo que se conoce como energía libre de mezcla (Tiraferri et al., 2011)

Debido al movimiento Browniano, si la concentración de partículas no es uniforme, se produce un flujo de material desde la zona de alta concentración hacia la de más baja concentración hasta alcanzar un equilibrio. Esta es la razón por la cual, si se introduce una celda hecha de una membrana semipermeable que contenga cierta dispersión coloidal, en un líquido puro, este trata de introducirse dentro de ella para establecer una concentración uniforme, diluyendo la dispersión coloidal y produciendo una elevación de líquido dentro de la celda. El incremento de volumen produce una presión llamada Presión Osmótica (Valencia, 1981)

De acuerdo con Van't Hoff, la presión osmótica P (bar) es directamente proporcional a la concentración C (g/l) y la temperatura absoluta T , es inversamente proporcional al peso molecular M y Donde R es la constante de los gases igual a 0,082 litros atm/mol grados kelvin (Hoek et al., 2002)

$$p = \frac{C}{M} * R * T$$

- **PRESIÓN DE ÓSMOSIS RETARDADA PRO**

PRO se basa en el aprovechamiento de diferencias en las concentraciones salinas de soluciones separadas por una membrana semipermeable para generar un flujo de agua constante. Cuando esto sucede, el agua se difundirá desde el lado de la solución de alimentación en el lado de la solución de empate que es el lado de agua de mar que está presurizado. El agua salada a presurizada y diluida se divide entonces en una corriente que va a través de una hidroturbina para generar energía mediante la despresurización de la misma, y el otro que pasa a través de un intercambiador de presión para reaprovechar la energía transmitiéndola al agua de mar para lograr el mantenimiento de la circulación. En un proceso de PRO, densidad de potencia (W) se normaliza por el área de la membrana y se utiliza comúnmente para representar la eficiencia de conversión de energía de la membrana.

Esta se puede expresar de la siguiente manera:

$$W = J * w * \Delta P$$

- **SISTEMAS AUTOMATIZADOS (INDUSTRIALES)**

Son sistemas de control usualmente orientados a la industrialización según la Universidad de Buenos Aires (Ii, 2000): “Un sistema de control manipula indirectamente los valores de un sistema controlado, siendo su objetivo gobernar ese sistema sin que el operador intervenga directamente sobre sus elementos. El operador manipula valores de referencia y el sistema de control se encarga de transmitirlos al sistema a gobernar a través de los actuadores.

El sistema de control opera, en general, con magnitudes de baja potencia, llamadas señales y gobierna unos actuadores que son los que realmente modulan los accionamientos que son los que realmente modulan la potencia entregada al sistema controlado potencia entregada al sistema controlado.”

Una imagen que describe la estructura de un sistema de control común se muestra a continuación:

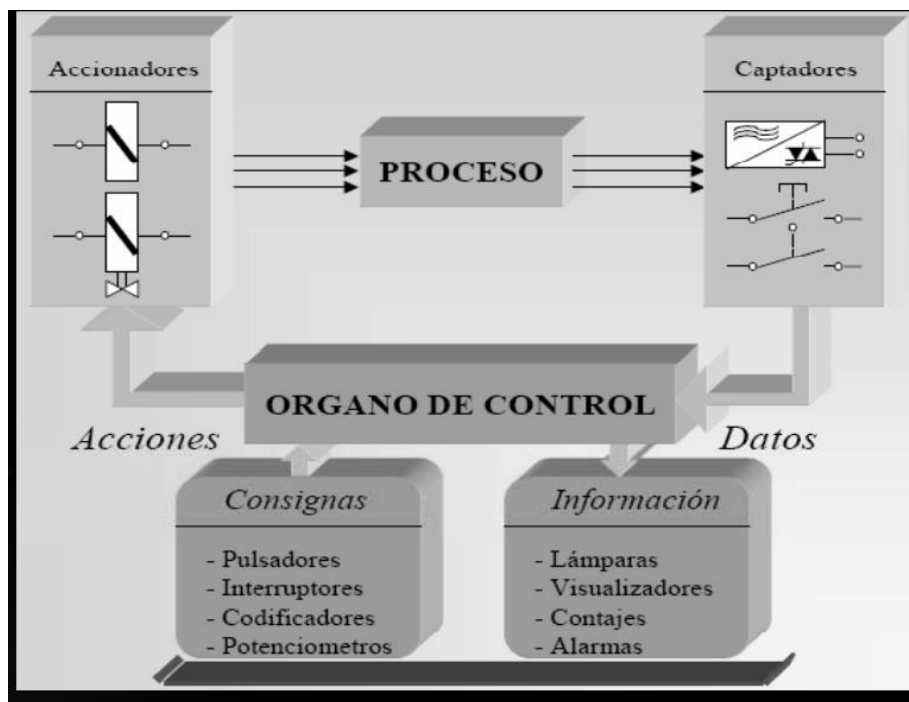


Figura 3: Esquema de sistema de control. (Roman, 2010)

- **MQTT (Message Queue Telemetry Transport)**

Es un protocolo de red leve y flexible que ofrece el equilibrio ideal para los desarrolladores de IoT (Yuan, 2017).

- Un protocolo liviano permite la implementación en hardware de dispositivos altamente restringidos y en redes de ancho de banda limitada y de alta latencia (Yuan, 2017).
- Su flexibilidad hace posible el soporte a diversos escenarios de aplicación para dispositivos y servicios de IoT (Yuan, 2017).

- **NodeJS**

Es un entorno de ejecución de Javascript basado en el motor V8, orientado a eventos asíncronos y diseñado para crear aplicaciones network escalables (OpenJS-Foundation, n.d.) y rápido en operaciones IO (Input/Output) o donde se realizan muchos requests, gracias a su paradigma de “non-blocking, event-driven I/O paradigm” (Capan, 2017).

Además, se tienen forma de sortear médium size computer intensive operations con técnicas como los brokers y librerías optimizadas (Patel, 2019).

- **Mongo Driver**

Es un entorno de ejecución de Javascript basado en el motor V8, orientado a eventos asíncronos y diseñado para crear aplicaciones network escalables (OpenJS-Foundation, n.d.); rápido en operaciones IO (Input/Output) o donde se realizan muchas peticiones al motor de base de datos MongoDB.

- **Dump components**

Son componentes de react que se enfocan específicamente en el aspecto grafico de la aplicación recibiendo solo props con la información a desplegar para el usuario (Arnold, 2017).

- **Smart components**

Son componentes de react que se enfocan en la implementación de la lógica de la aplicación manteniendo el control sobre los estados y asegurando que la aplicación funcione (Arnold, 2017).

- **Patrón de diseño**

Según el libro “patrones de diseño, elementos de reutilizables del software orientado a objetos”.:” Un patron describe un problema que ocurre una y otra vez, y luego describe una solución base/central de tal manera que pueda ser utilizada un millon de veces y no ser usada de la misma manera dos veces” (Gamma et al., 1994).

- **Patrón adapter**

En esencia, permite convertir la interface de una clase en otra interface que el cliente espera. El patrón adapter permite trabajar juntas clases u objetos que de otra forma no podrían por tener interfaces incompatibles.

Erich Gamma, et al propone un ejemplo de un editor de dibuja, donde la implementación de una clase que TextView va contra el dominio específico del sistema de clases por lo que decide crear un adaptador que pueda hacer cohesiva la existencia de esta clase (y con ella su funcionalidad dentro del sistema) (Gamma et al., 1994).

- **Patrón proxy.**

Permite la creación de un representativo de un objeto o espacio de memoria de para un objeto y controlar el acceso al mismo (Gamma et al., 1994).

Permitiendo como uno de sus aplicabilidades (y principal para su uso en el presente) un objeto representativo que pueda representar el acceso a una información que se encuentra de manera remota (Gamma et al., 1994).

- **Patrón visitor**

En esencia, representa una operación a ser realizada en los elementos de una estructura de objetos. Visitor permite definir una nueva operación sin cambiar las clases de los elementos en los cuales opera (Gamma et al., 1994).

- **Patrón de acceso a datos DAO**

Siglas que provienen de “Data Access Object”, se utiliza para separar la lógica de persistencia de datos en una capa separada en la aplicación; el servicio permanece como una

caja oscura sobre como las operaciones de bajo nivel de acceso a la base de datos son hechos (Wikipedia, n.d.)(Oracle, n.d.)

- **EcmaScript**

ECMAScript es una especificación de lenguaje de programación publicada por ECMA International. El desarrollo empezó en 1996 y estuvo basado en el popular lenguaje JavaScript propuesto como estándar por Netscape Communications Corporation. Actualmente está aceptado como el estándar ISO/IEC 22275:2018.

ECMAScript define un lenguaje de tipos dinámicos ligeramente inspirado en Java y otros lenguajes del estilo de C. Soporta algunas características de la programación orientada a objetos mediante objetos basados en prototipos y pseudoclases.

- **Inmutabilidad**

Un patrón relacionado a javascript y que cobra alta importancia con las mejoras de EcmaScript aplicadas sobre el lenguaje en el nivel de codificación donde se prefiere no cambiar el valor de una variable (en especial aquellas complejas) y en su defecto manejar los cambios de información con variables de tipo constantes (“const”), que es la forma en la que actualmente se deciden manejar el cambio de estados y de información en frameworks de la actualidad como React.

- **Desestructuración de objetos (en Javascript con EcmaScript)**

Capacidad de reasignación y manejo de variables de manera instantánea y con escritura de código limpia, permitiendo un mapeo de los objetos y arreglos de manera instantánea (Javascript.info, n.d.), permitiendo en arreglos (Mozilla-org, 2020b):

- Asignación básica de variables
- Asignación separada de la declaración
- Valores predeterminados
- Intercambio de variables
- Analizar un arreglo devuelto por una función
- Ignorar algunos valores devueltos
- Asignar el resto de un arreglo a una variable
- Desempacar valores coincidentes con una expresión regular

Permitiendo en objetos (Mozilla-org, 2020b):

- Asignación básica de variables
- Asignación separada de la declaración

- Asignar nombres a nuevas variables y proporcionar valores predeterminados
- Desempacar campos de objetos pasados como parámetro de función
- Establecer el valor predeterminado de un parámetro de función
- Rest en la desestructuración de objetos
- Identificador de JavaScript no válido como nombre de propiedad
- Nombres de propiedades de objetos calculados y desestructuración
- Interacción con iteración "for...of"
- Búsqueda de la cadena de prototipos al desestructurar el objeto.

Todas usadas ampliamente en el desarrollo con Javascript moderno.

- **Closures**

“Una clausura o closure es una función que guarda referencias del estado adyacente (ámbito léxico). En otras palabras, una clausura permite acceder al ámbito de una función exterior desde una función interior. En JavaScript, las clausuras se crean cada vez que una función es creada.” (Mozilla-org, 2020a).

Estas dan una gran oportunidad para hacer componentes reutilizables dentro de Javascript y junto con los exports o includes (según en donde se esté aplicando javascript) permiten una alta capacidad de modularización y creación de componentes reutilizables, llegando a ser la base de los componentes funcionales en React.

8. METODOLOGÍA

La presente investigación es de tipo adaptativa ya que hace uso de las tecnologías de sensado actuales, a través de la automatización de sistemas, para permitir un mayor nivel de supervisión de los procesos de ósmosis retardada e inversa que usan algunos prototipos tecnológicos de la universidad. La presente investigación también es de tipo aplicada pues usó los conocimientos obtenidos hasta la actualidad de sistemas de control y de sensado para el desarrollo de un sistema de monitorización de procesos osmóticos en los prototipos de la Universidad de Cartagena, mismos que se encuentran en la sede Piedra de Bolívar y que fueron mejorados desde finales del año 2018 hasta junio de 2021.

Para cumplir los objetivos del proyecto se organizó el desarrollo del mismo en las siguientes fases:

1. Investigación preliminar del comportamiento del sistema objetivo.
2. Identificación de requerimientos y variables influyentes en el funcionamiento del sistema
3. Diseño y prototipado del sistema (algorítmicamente y a nivel de hardware)
4. Implementación del prototipo piloto.
5. Análisis de interacción recursos-sistema propuesto y sistema propuesto-usuario mediante toma de datos.
6. Regulación del sistema y monitoreo de evolución del prototipo.
7. Solución y cambios resultantes.

La primera y segunda fase están encaminadas a cumplir con el primer objetivo del proyecto propuesto (la etapa de identificación de requerimientos y con ello la definición de un modelo de negocio), la siguiente se encarga de cumplir la etapa de diseño del sistema (segundo objetivo del propuesto). La fase cuatro se encarga de la implementación del sistema buscando cumplir con lo planteado en el tercer objetivo propuesto y la quinta, sexta y séptima fase se ocupan de la validación y mejoramientos finales del mismo.

Si bien los pasos anteriores no se basan en los pasos propuestos por el ingeniero Avid Roman (Roman, 2010), su presentación de un sistema general de automatización apoya el nuestro; favoreciendo la pertinencia de nuestra propuesta una figura del mismo se muestra a continuación:

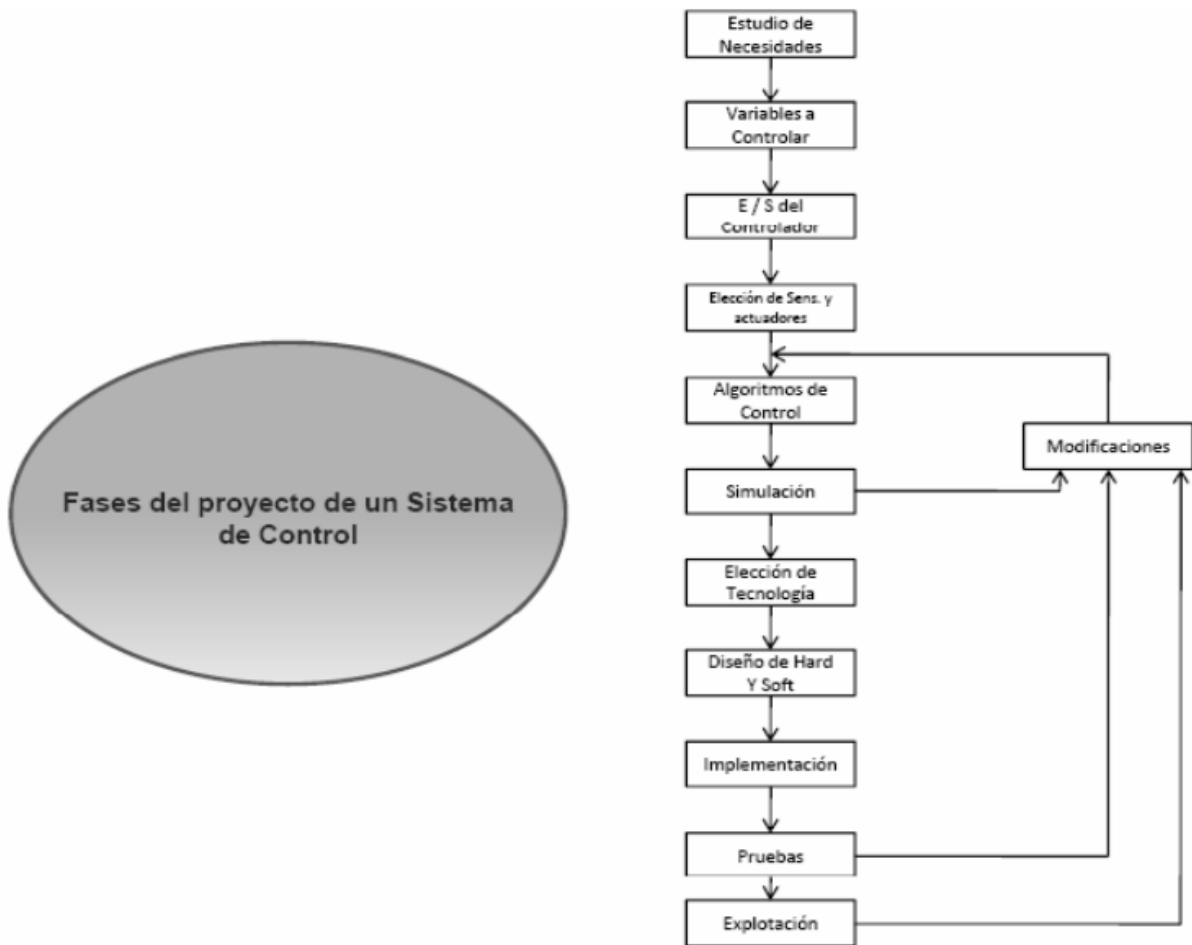


Figura 4: Esquema general de un sistema automatizado (Roman, 2010)

Todos los pasos anteriores se llevarán a cabo como un trabajo conjunto con concedores de procesos osmóticos que trabajan en el prototipo armado en la Universidad de Cartagena.

Para manejar la realización del producto propuesto en el presente la metodología de desarrollo de software fue una RUP modificado, en el que las iteraciones se dieron en el siguiente orden:

- Primero búsqueda de requerimiento inicial (1 iteración)
- Ideación de sistema inicial y diseño preliminar (1 iteración)
- Realización de la calibración de los sensores dejando un algoritmo capaz de manejarlos (9 iteraciones)
 - Presión (2 iteraciones)
 - Flujo (2 iteraciones)
 - Temperatura (2 iteraciones)

- Fuerza (3 iteraciones)
- Diseño de la interfaz de la pantalla TFT y su interacción con el microcontrolador (2 iteraciones)
- Conexión del microcontrolador a la red WiFi y a un primer receptor (3 iteraciones)
- Desarrollo del sistema backend final, la respectiva plataforma web y el alineamiento con el sistema de contenedores para ser desplegados más rápidamente (4 iteraciones).

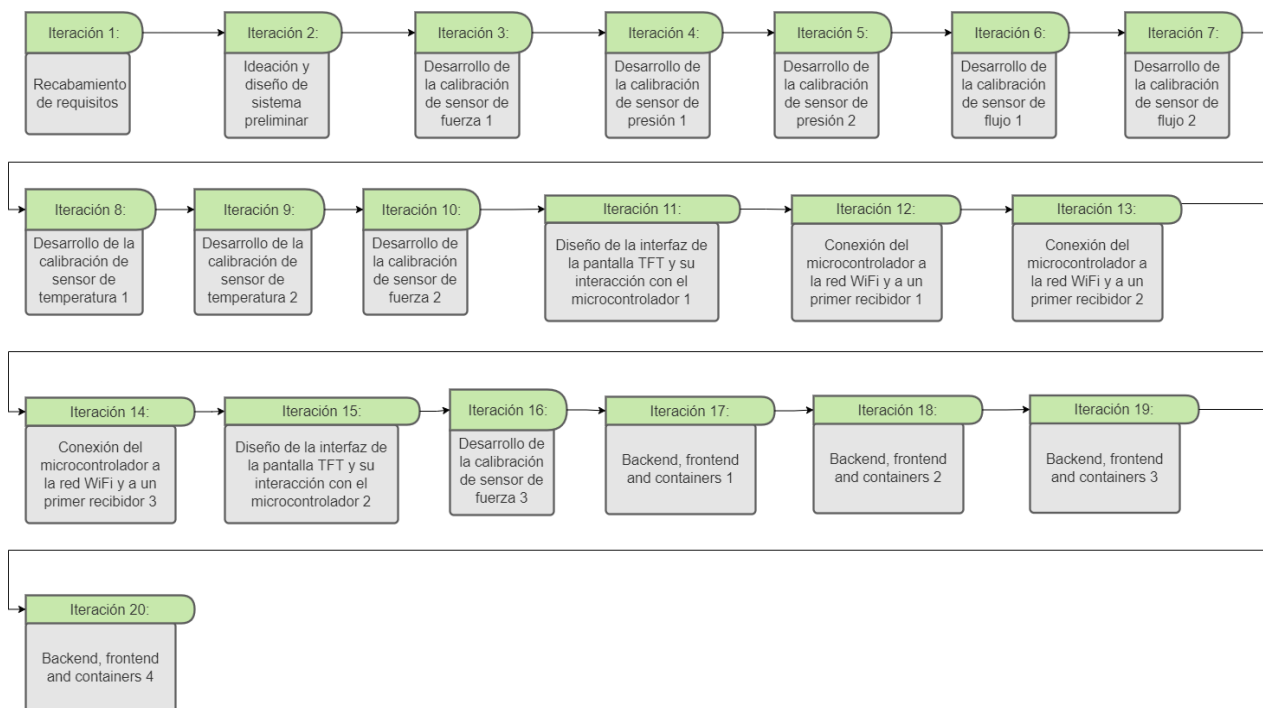


Figura 5: imagen de muestra del orden de las iteraciones del proyecto.

Hay que tener en cuenta que las iteraciones que denotan desarrollo, tuvieron una pequeña sub fase de búsqueda de requisitos individuales, diseño y de pruebas (tratando así cada una como un módulo funcional dentro del sistema)

Debido a las bases físicas sobre las que trabajan los procesos osmóticos el sistema ya cuenta con variables predefinidas: temperatura (°C), presión (PSI), flujo (Litro/minuto), fuerza (gramos); la naturaleza de los datos en general tiende a ser de propiedades intrínsecas al proceso bajo el cual se trabaja (ósmosis), para la obtención de esta información se pretende establecer hardware libre con rangos de funcionamiento de precisión industrial establecidos por los expertos en procesos osmóticos y se usarán los protocolos de comunicación establecidos por los sensores (Onewire, I2C, Lecturas analógicas, entre otros).

Por su parte, la metodología de toma de datos es como lo usual en sistema de microcontroladores, distribuyendo el ciclo de reloj de tal manera que se pueda obtener las lecturas por debajo del segundo para obtener frecuencias de datos adecuadas de las variables antes descritas, también haciéndose énfasis en la precisión de lectura en las mismas dependientes a las características bajo las que trabajan los sensores (histéresis, precisión, variación por repetitividad, desviación por operación continua, entre otras), siendo tratadas mediante técnicas matemáticas para tratar la varianza de datos como el promedio, remoción de saltos basados en offset calculados, offsets basados en especificaciones técnicas y obtenidos por pruebas de calibración y ecuaciones dadas en la ficha técnica para trabajar con los sensores.

El procesamiento de los datos se realizó en el sistema físico por medio de una pantalla Touch Screen, ya que los encargados de operar la máquina necesitan independencia funcional de la red, esto es posible con el microcontrolador ESP32 que contiene dos cores uno para procesamiento (de 240MHz) y otro para operaciones de red de (80MHz), por su parte la pantalla Nextion HMI 7 contiene las dimensiones adecuadas para visualizar datos.

El sistema final contempla la implementación de una parte remota que se encarga de capturar los datos, mostrarlos cuando es requerido y permitir su exportación en formato CSV, ya que la infraestructura no garantiza periodos de conexión a internet 100% garantizados y también es útil tener un sistema en red en el que puedan ser consignados los datos, mostrados y exportados sin tener que estar presente en el lugar del prototipo osmótico físicamente momento en el cual se puede utilizar Web la muestra y la exportación para los encargados de operar la máquina de operar la máquina, a continuación se muestra un diagrama de muestra de la comunicación del sistema (**ver figura 6**).

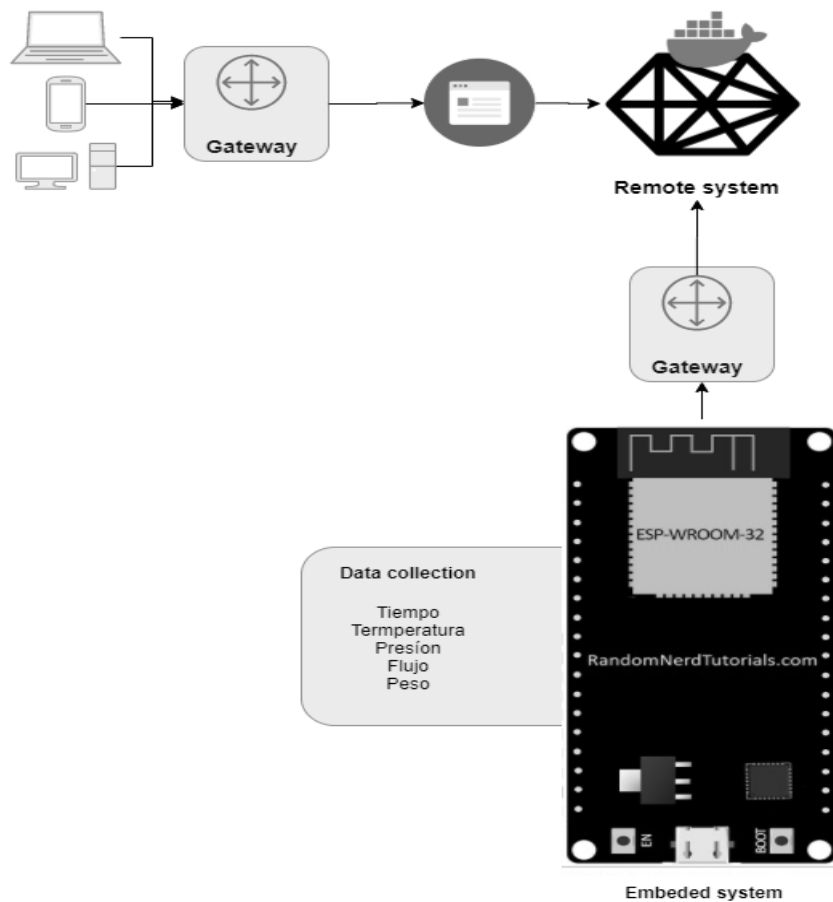


Figura 6: diagrama general de comunicación del sistema

La metodología de toma de datos se debe dividir en dos partes importantes del proyecto, la primera se encarga de la recolección de datos de cada sensor individualmente hasta lograr su calibración cuya confiabilidad se basa en que los datos recogidos fueron relacionados con los datos de sensores (con tecnología propietaria) que ya venían calibrados; de los cuales todavía se tienen 14 archivos de datos, la duración en las sesiones de medidas eran de unos minutos hasta media hora mientras se cambiaban los parámetros de funcionamiento y se volvía a subir código de corrección de ser necesario.

La segunda etapa de medida consistía en evaluar la frecuencia de toma de datos, con todos los sensores y la pantalla conectados y evaluando el envío de información hacía un servidor receptor, está basa la certeza de las medidas en base a la calibración hecha en la primera fase y evalúa más en detalle la frecuencia de medición que debe ser cercana al segundo, dada la conexión del microcontrolador con todos los sensores (con excepción de algunos que no se pudieron reemplazar por la poca factibilidad de reemplazo de sensores en el sistema), la

pantalla y demás elementos electrónicos, el aprobador final fue uno de los encargados de operar la máquina, quien después de ver los .CSV presentados, mencionó que la frecuencia de medidas era adecuada, cada sesión de toma de datos demoró de 30 minutos a 1 hora (tiempo de uso usual para estos sistemas).

En el sistema remoto se deben tener en cuenta varias partes que conforman el producto software funcional final: el backend, el frontend, la base de datos y el broker; cada uno de estos elementos constitutivos debe contener una estructura cohesiva que permita su interacción mutua con los demás y con el sistema físico que continuamente está enviando sus datos.

El broker se encarga de recibir y retransmitir todos los mensajes MQTT provenientes del sistema del sistema físico.

Por su parte, el backend utiliza un componente creado bajo el patrón proxy, para gestionar a través de él toda la información que viene de los canales MQTT y que es recibida por la suscripción a los canales.

Además, en el mismo sistema fueron establecidos módulos utilizando el patrón singleton, que pudieran gestionar las funcionalidades necesarias para el funcionamiento de los roles emisores de datos de las máquinas y administradores/supervisores/analistas de los usuarios, como servicios reutilizables que pudieran funcionar en los distintos routers y el manejador MQTT, sin tener que manejar un nivel de complejidad mayor en estos manteniendo innecesariamente múltiples instancias de los servicios.

Luego, estos servicios a su vez pueden moldear y transformar la información que reciben tomando el papel de la capa de la lógica del sistema, la implementación de esta organización del sistema está basada en un framework actual para NodeJS utilizado por las compañías para definir una estructura base escalable de su backend de manera fácil; es importante mencionar que no fue utilizado directamente en el sistema actual ya que inicialmente incrementaría la curva de aprendizaje para los desarrolladores a cargo de mantener este proyecto en el nivel universitario.

Los servicios del sistema de NodeJS luego de haber trabajado con los datos de la manera pertinente pueden contactar a las instancias de las clases de acceso a datos encargadas de manejar la capa de persistencia, siendo esta interacción entre clases la propuesta por el patrón de diseño DAO.

Por último, el sistema frontend que permite la muestra y exportación de datos de las lecturas enviadas por el prototipo osmótico, la capacidad de gestionar las máquinas y usuarios (tipo administrador, supervisor y analista), con permisos de solo consulta y exportación para el usuario analista y consulta, exportación, agregación, actualización y eliminación para el de tipo supervisor y todas las funcionalidades para el administrador esperando sea un rol de soporte para las operaciones del sistema, asegurando con estas capacidades los requisitos funcionales base del proyecto.

Para asegurar lo anterior con una buena estructura dentro del sistema frontend, se siguieron las buenas prácticas de dividir el sistema en componentes Smart y componentes Dumb que permitieran desestructurar los elementos constitutivos de la plataforma web y con ello se diera lugar a un mejor escalado y reutilización dentro del sistema.

Además, para agilizar el desarrollo de la parte gráfica y asegurar los lineamientos de desarrollo de las interfaces de usuario mucho más rápido, en el nivel de “Dumb components” que son los encargados de la presentación, se utilizó la librería “Ant Design”, misma que ya he usado en proyectos reales y que, aunque no da un nivel atómico de reutilización como “Atomic Design” (que tampoco es necesario para la interfaz del presente), provee un gran nivel de personalización a nivel de estilizado y muchos dumb components base con los que trabajar.

Ant Design más específicamente permite agilizar el desarrollo ofreciendo un set de componentes testeados de elementos base, de estructuración y ordenamiento, de controles de usuario y otros componentes visuales, permitiendo que, aunque se aumente un poco la curva de aprendizaje y se utiliza la expresión “poco” porque parte de su base es muy familiar con Bootstrap si hacemos un símil de las clases que este último proveía con los componentes de estructura y ordenamiento que el primero ofrece, un ejemplo de esto son las classes CSS “col” para ordenamiento de contenedores.

Y, además, tiene una gran cantidad de lineamientos y sugerencias para el desarrollo de UI asegurando el UX (Ant-Design, n.d.), lo que facilitó asegurar aspectos de calidad de la interfaz de usuario como: la familiaridad de la interfaz, la usabilidad, la facilidad de uso, la respuesta y la tolerancia a fallos, como aspectos principales para la interfaz; concentrando la accesibilidad en las personas que tengan el sentido de la vista adecuado para visualizar la información en una pantalla y de moción dentro de lo que se considera la media de usuarios que utilizan el computador con capacidades cognitivas y motoras no impedidas.

9. RESULTADOS

Teniendo como referencia el **primer objetivo** del presente que es la recolección de los requerimientos necesarios para establecer los puntos base del diseño.

Se llevó a cabo una investigación previa con los encargados de diseñar, construir y operar el sistema a través de reuniones con los involucrados, lo que incluyó la interacción con los estudiantes de ingeniería civil (Michelle Montero y Francisco Herrera) y química (Nohelia Polo e Neymar Reyes) que tenían tesis conjuntas para el mismo proyecto de la presente, el encargado de los laboratorios (Remberto Avila) de establecer la parte eléctrica y dar aportes en la electrónica de bajo nivel, las evidencias son las actas de reunión tomadas del semillero y las fotos tomadas como entregable de esas reuniones (ver *figuras 7, 8 y 9* a continuación).

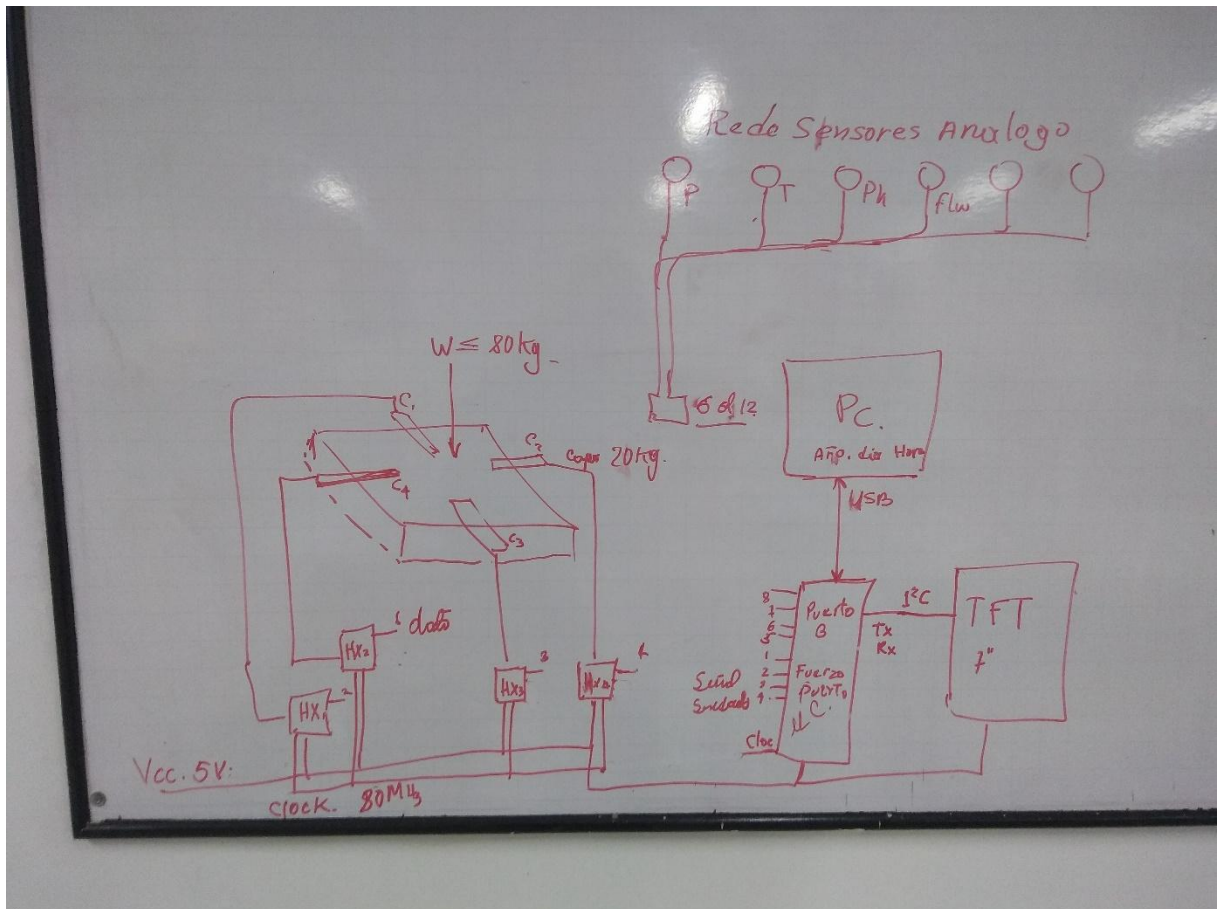


Figura 7: Tablero de la reunión donde se sentaron las bases del sistema y se explicó una propuesta base.

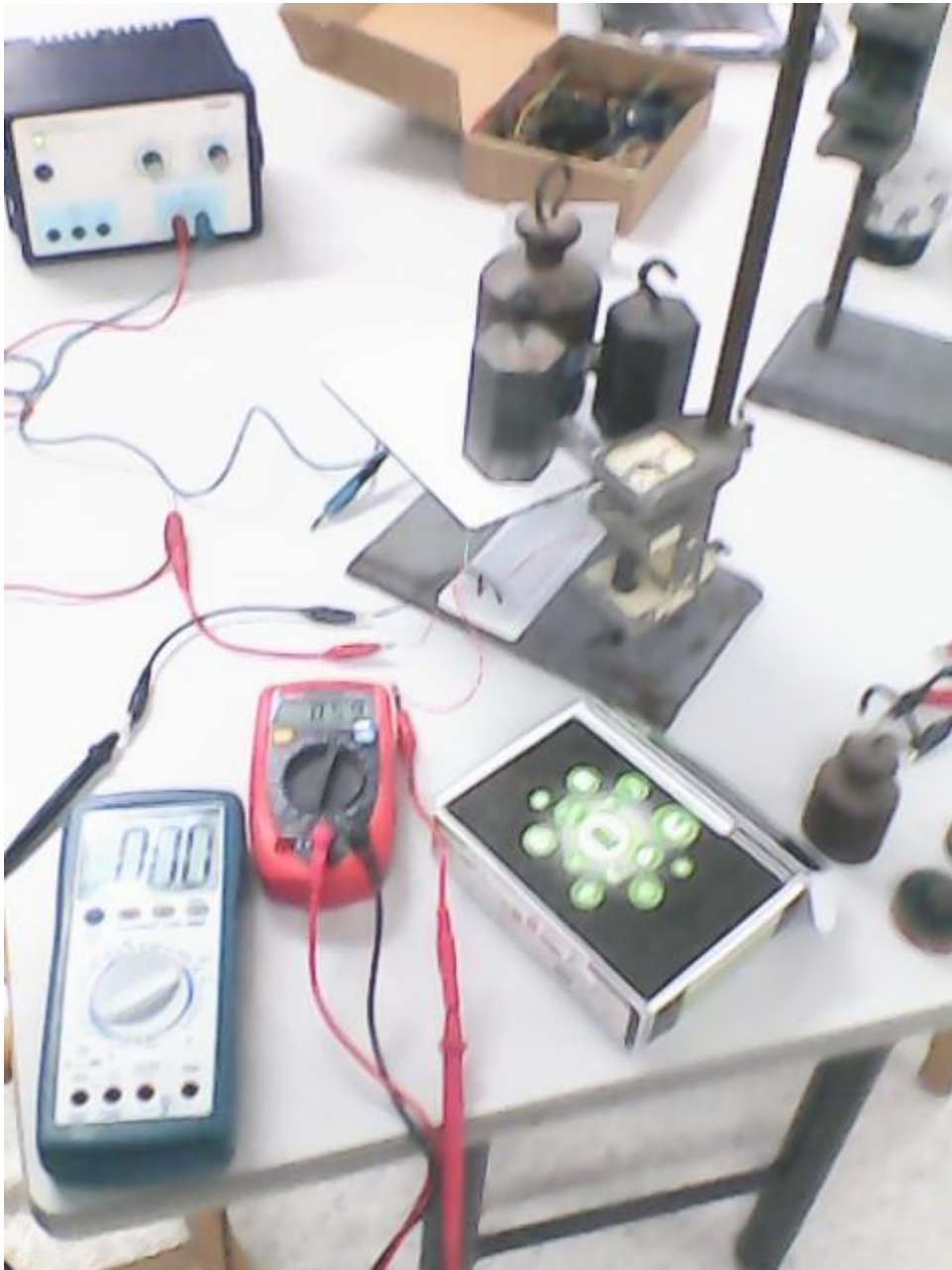


Figura 8: foto de prueba de sensibilidad de posibles sensores propuestos durante la segunda reunión.

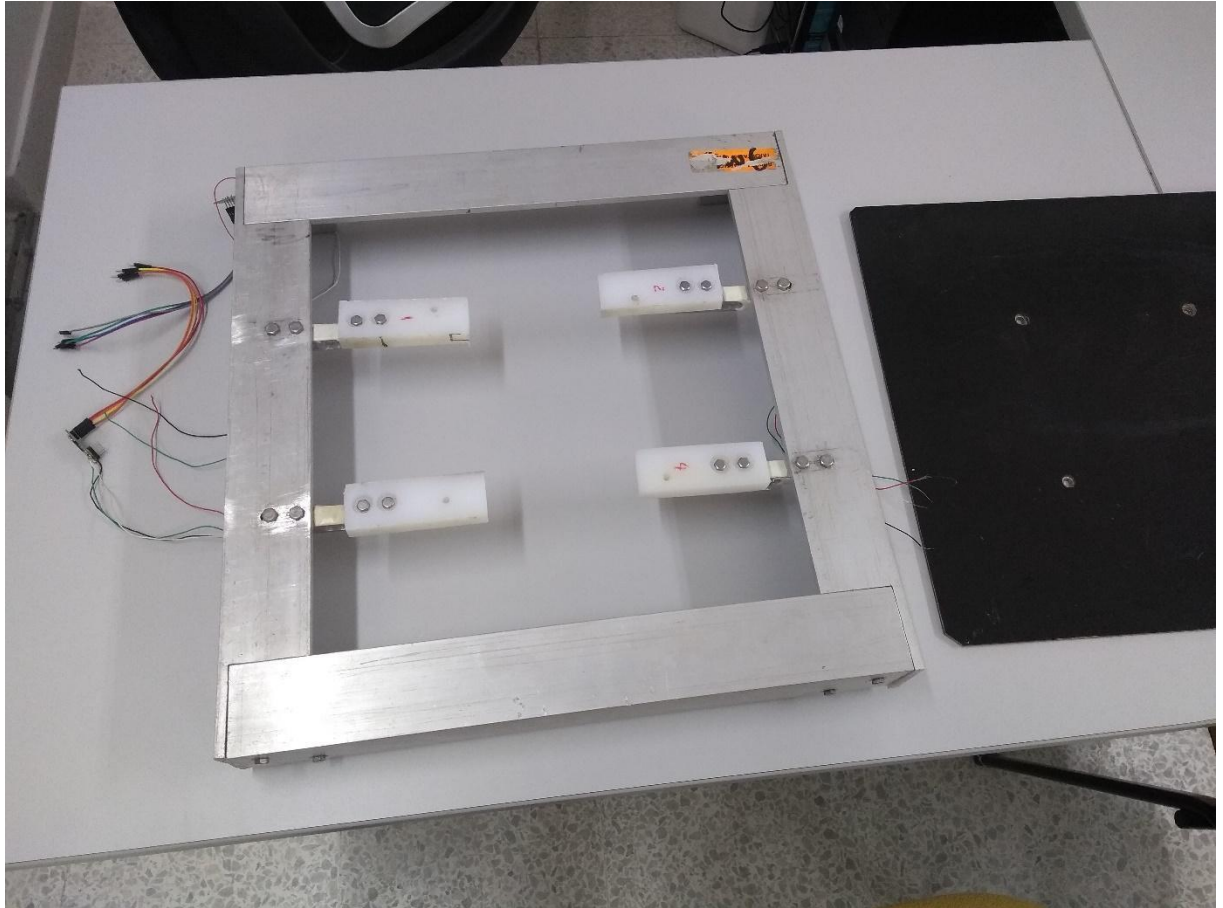


Figura 9: foto de la reunión donde se presentaron opciones de prototipos a implementar en los distintos puntos de toma de datos de las reuniones anteriores.

Esta fase se realizó para efectos de recabamiento de los requisitos funcionales y no funcionales según al formato de especificación de requisitos en el estándar IEEE 29148; algunos de los requisitos más importantes se enuncian a continuación.

Tabla 1: requisito funcional “Recolectar datos de los sensores” presente en el anexo 1 documento de especificación de requisitos

Número de requisito	RF1.1
Nombre de requisito	Recolectar datos de los sensores
Tipo	Requisito
Fuente del requisito	Cliente
Prioridad del requisito	Alta

Tabla 2: requisito funcional “Recolectar datos de los sensores” presente en el anexo 1 documento de especificación de requisitos.

Número de requisito	RF1.2
Nombre de requisito	Iniciar y detener la toma de datos
Tipo	Requisito
Fuente del requisito	Cliente

Prioridad del requisito	Alta
-------------------------	------

*Tabla 3: requisito funcional “Recolectar datos de los sensores” presente en el **anexo 1** documento de especificación de requisitos.*

Número de requisito	RF1.3
Nombre de requisito	Mostrar datos de lectura de los sensores localmente
Tipo	Requisito
Fuente del requisito	Cliente
Prioridad del requisito	Alta

Los anteriores y demás requisitos se encuentran en el **anexo 1 (documento de especificación de requisitos)**, que sirvió como contenedor de la investigación realizada con el cual se **cumplió el objetivo 1**.

El **segundo objetivo**, mismo que abarca el diseño del sistema, **se cumplió** con la realización para todo el sistema de diversos diagramas representativos de la composición del sistema desde múltiples aspectos, diagramas que son mostrados a continuación (estos pueden ser encontrados en el **anexo 2**).

1. Diagrama de casos de uso

Representa los principales casos de uso del sistema relacionados al usuario que hace uso de ellos (*ver figura 10*), para más información de estas funcionalidades se puede revisar el **anexo 3 (manual de usuario)** dado con el presente.

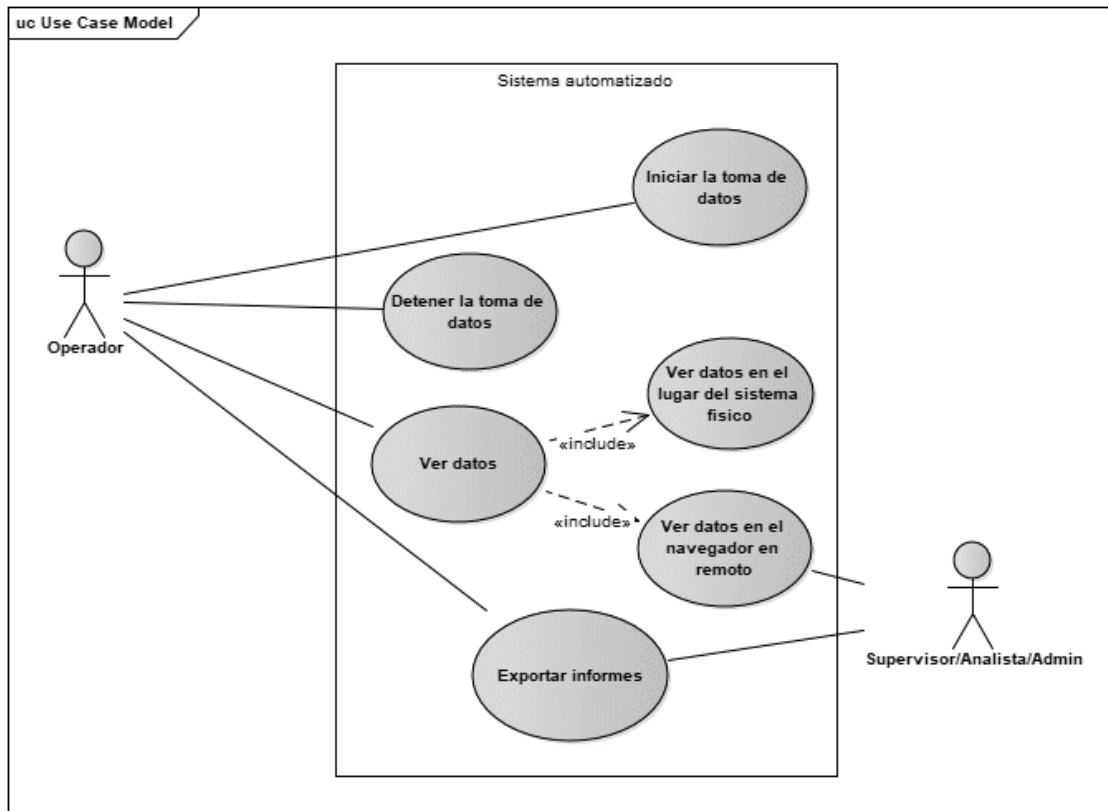


Figura 10: diagrama de casos de uso del sistema conjunto (físico y remoto).

2. Diagramas de secuencia de los casos de uso más importantes (y algunos secundarios que suplementan el workflow de los principales)

- Secuencia de login

Muestra como el usuario después de acceder a la plataforma debe ingresar sus credenciales de acceso, como el sistema maneja todos estos eventos generados por el

usuario y dependiendo del resultado, que se despliega el flujo de error de autenticación o el flujo de autenticación correcto (*ver figura 11*).

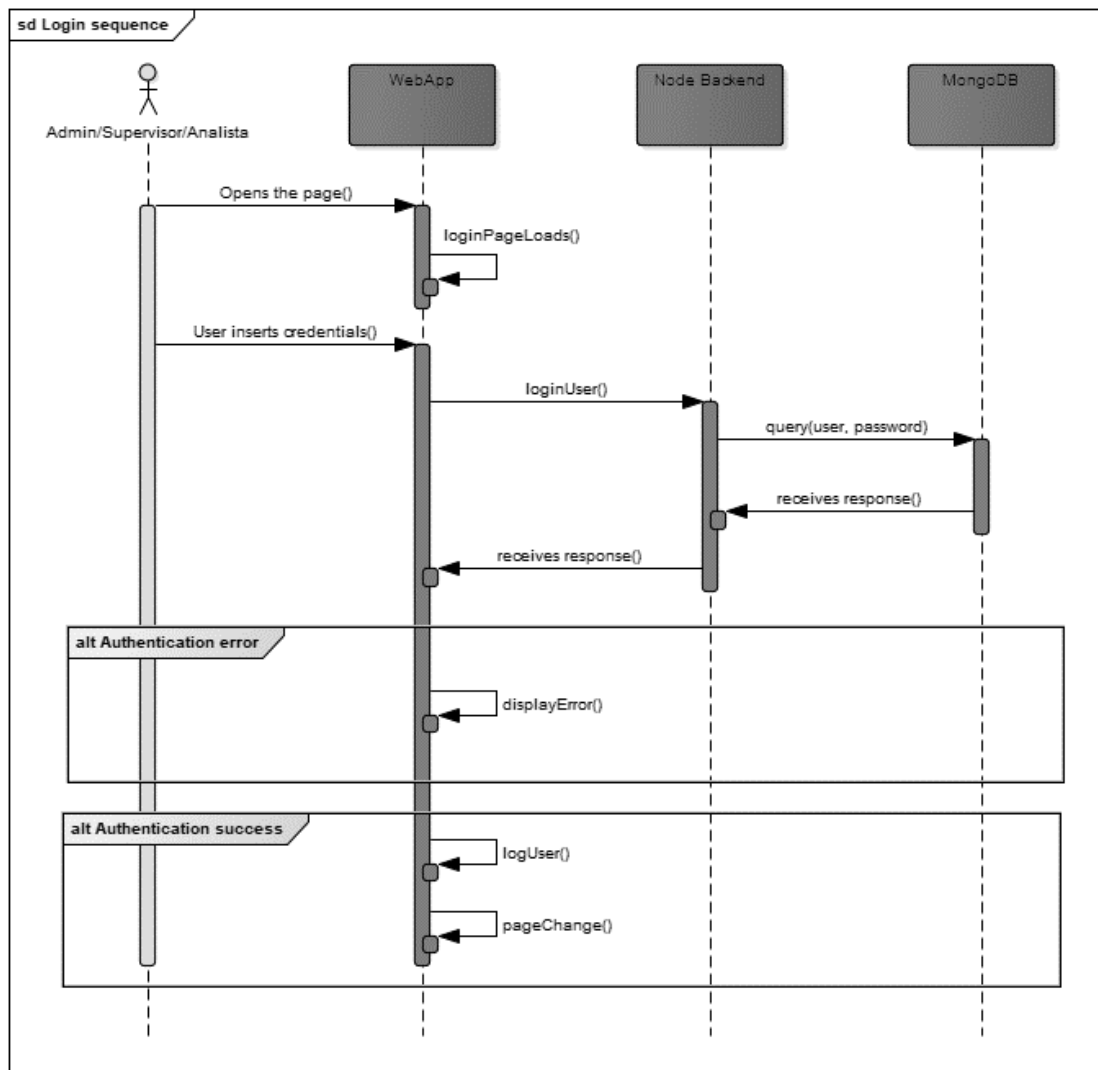


Figura 11: diagrama de secuencia que muestra el workflow utilizado para realizar un inicio de sesión en el sistema remoto.

- Mostrar data localmente

Muestra que luego de encender el sistema se realiza la función de configuración “main” presente en el código de los lenguajes programados con el framework de Arduino en PlatformIO y procede a entrar en la función “loop”, donde dentro existen procedimientos que en conjunto abarcan todas las lecturas de sensores, a este conjunto se le denomina de manera representativa “ReadMeasurements” en el diagrama, que luego actualiza en base a cada pointer que se tiene de las variables en la pantalla (*ver figura 12*).

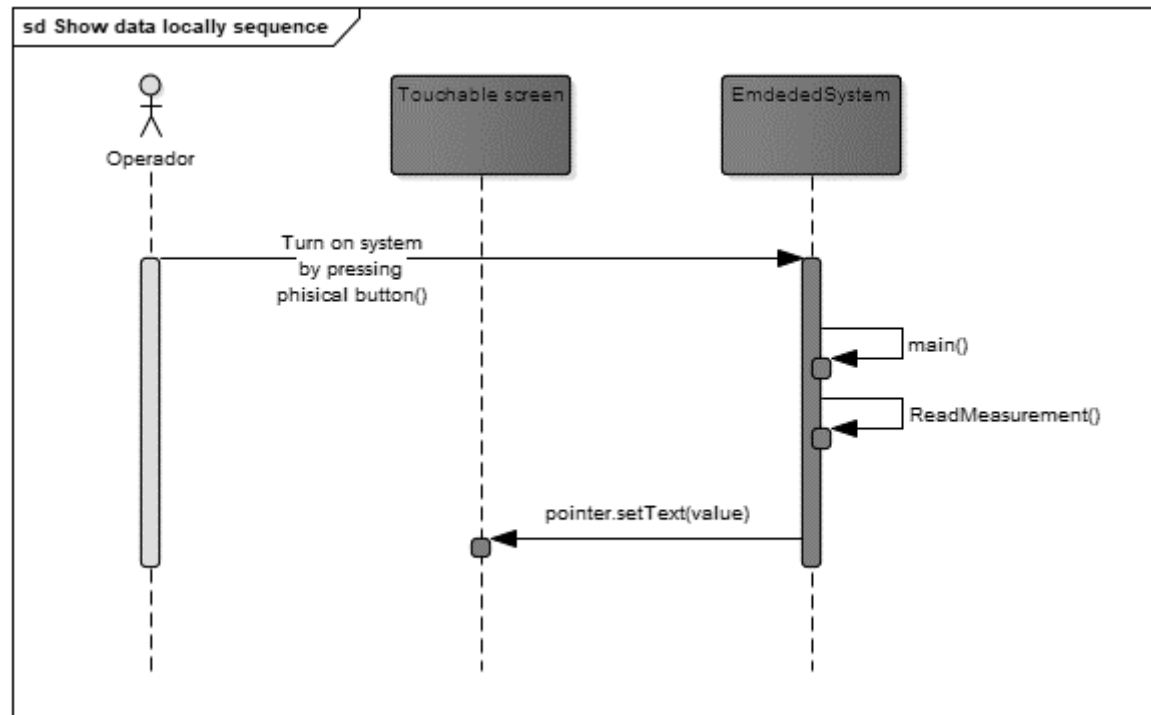


Figura 12: diagrama de secuencia que evidencia el workflow utilizado para mostrar los datos en el sistema físico..

- Comenzar la recolección de datos

Muestra como prerrequisito que debe cumplirse previamente con la secuencia “show data locally sequence” para mostrar datos en local y presionar el botón de “Comenzar toma de datos” en la pantalla táctil para que el sistema comience a hacer las peticiones pertinentes para preparar

la recepción de datos en el servidor, reflejando el resultado en la pantalla y solo restando enviar los datos por el protocolo MQTT hacia el bróker
(ver figura 13).

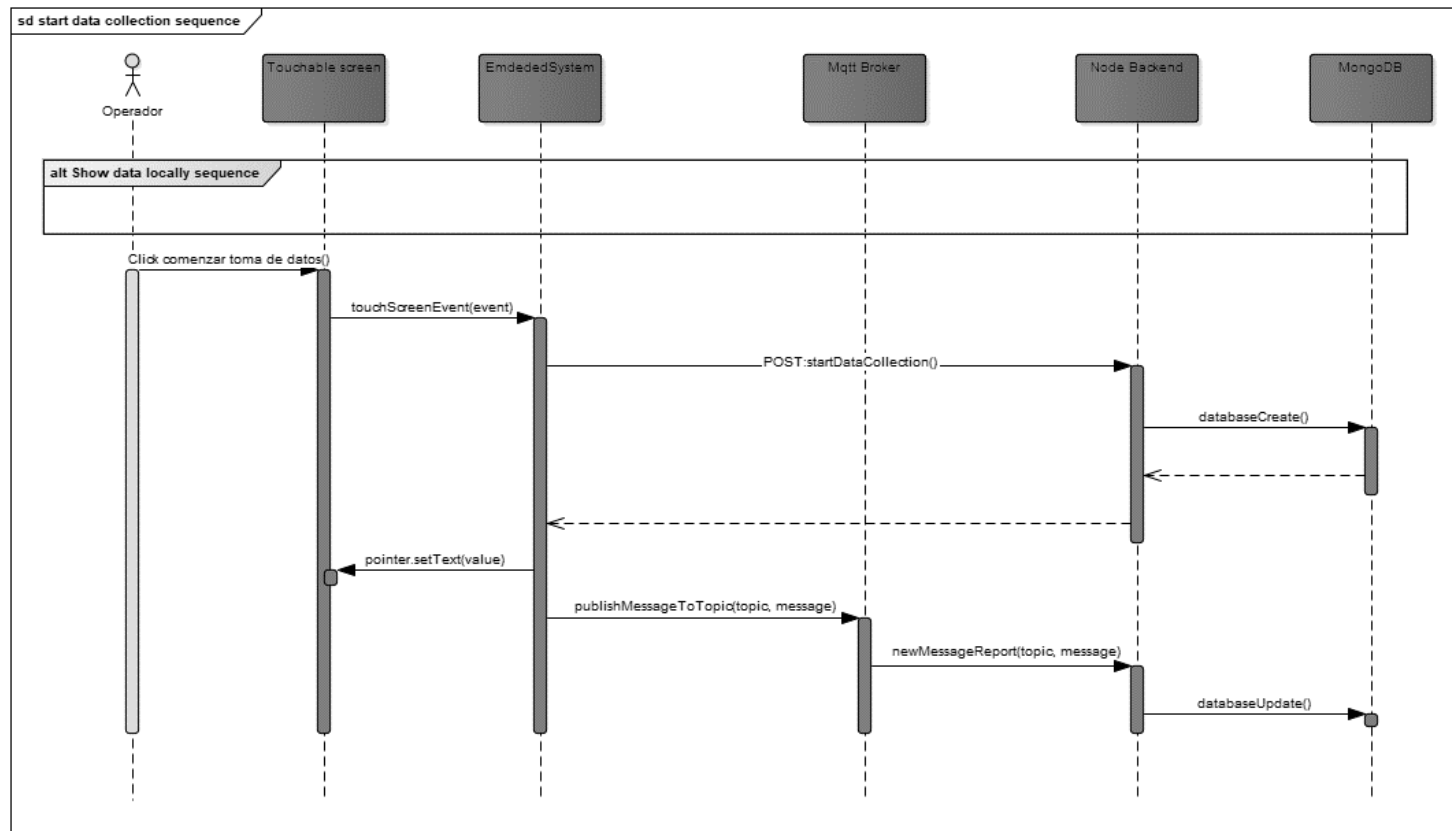


Figura 13: diagrama de secuencia que muestra el workflow utilizado para comenzar la recolección de datos en el sistema físico y el remoto.

- Detener la recolección de información

Muestra como prerrequisito que el sistema ya haya iniciado la toma de datos localmente, y luego para pararla que el usuario presione el botón “Parar la toma de datos”, el sistema consecuentemente hará la petición pertinente en el sistema remoto y retornará el resultado, momento en que el sistema físico cambiará el valor de la variable puntero en la pantalla para avisar al operador (*ver figura 14*).

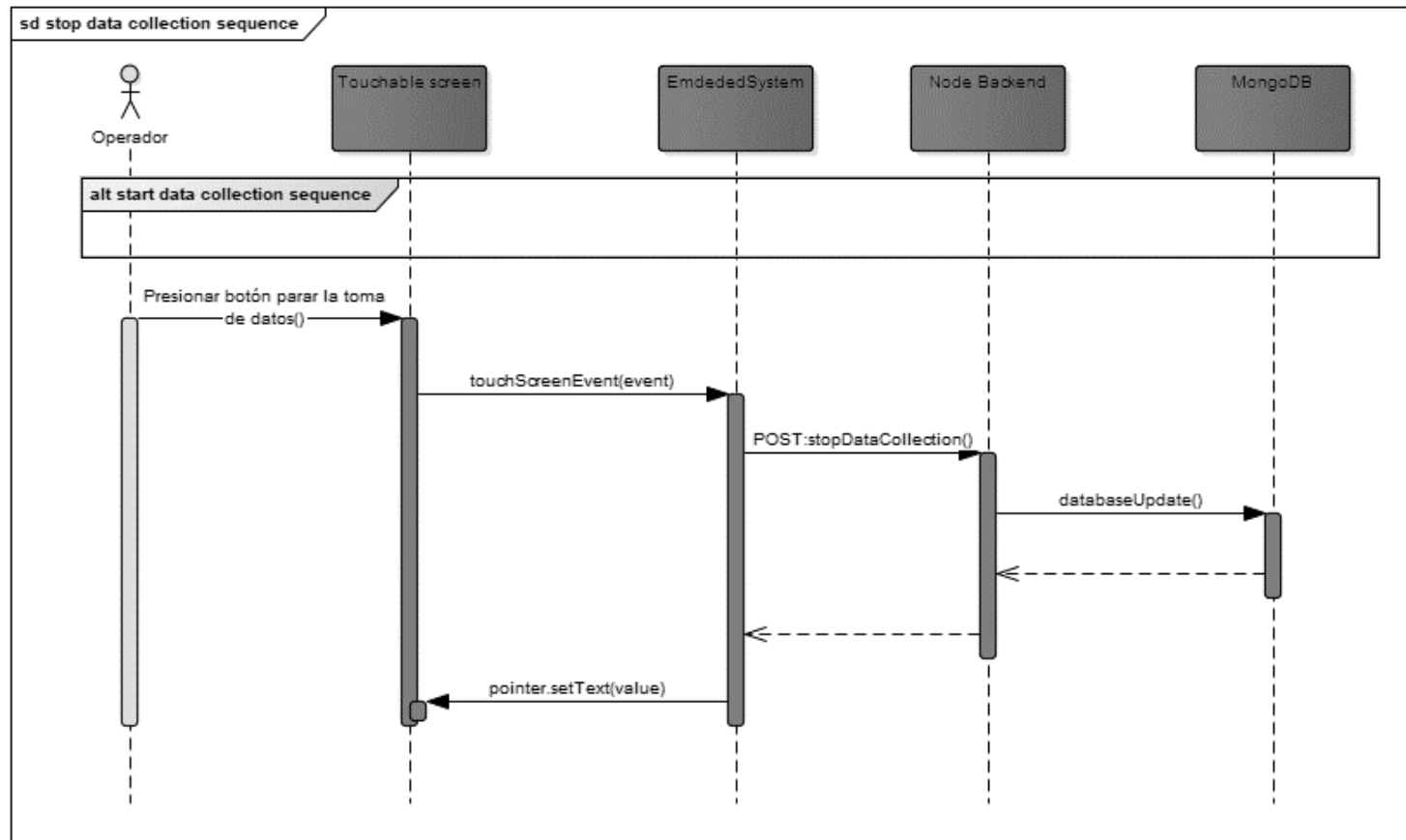


Figura 14: diagrama de secuencia que evidencia el workflow utilizado para detener la toma de datos en el sistema remoto.

- Mostrar datos remotamente

Tiene como prerequisite haber cumplido con los pasos de la secuencia “Login sequence”, luego se evidencia que comenzando con el acceso desde menú el sistema comienza a buscar toda la información necesaria para ser desplegada en la página de medidas, para momentos después actualizar la información de la página según se vaya recibiendo, los “Hook” que hacen esto posible son representados por medio de la función llamada “updateData” que aparece en el diagrama (*ver figura 15*).

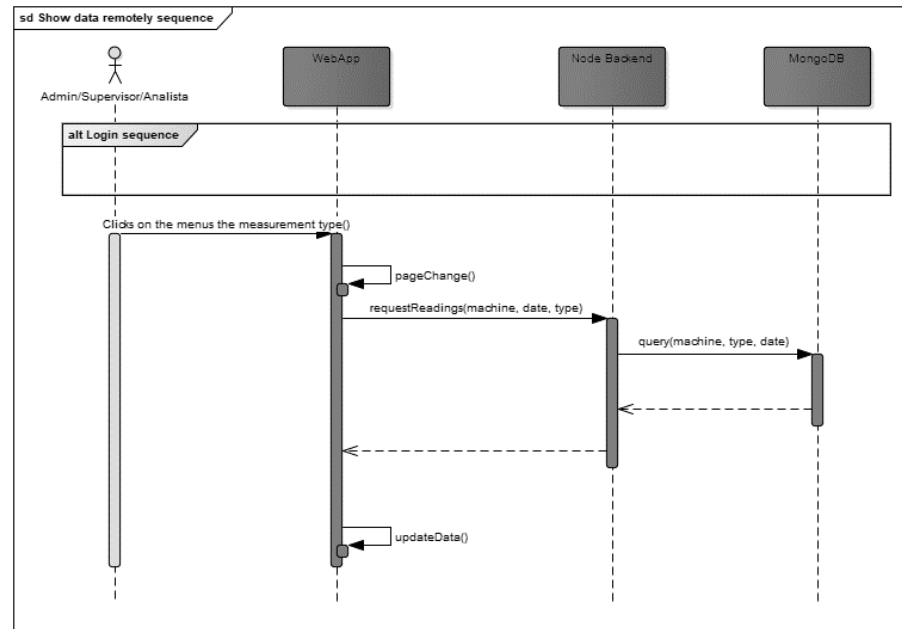


Figura 15: diagrama de secuencia que evidencia el workflow utilizado para mostrar los datos en el sistema remoto.

- Exportar datos remotamente

Evidencia que luego de lograr los pasos de la secuencia de mostrar los datos remotamente, se tiene que presionar el botón “Exportar” en la sección de las medidas y seguir con los pasos subsiguientes de manejo de archivos para guardar el documento en la computadora (*ver figura 16*).

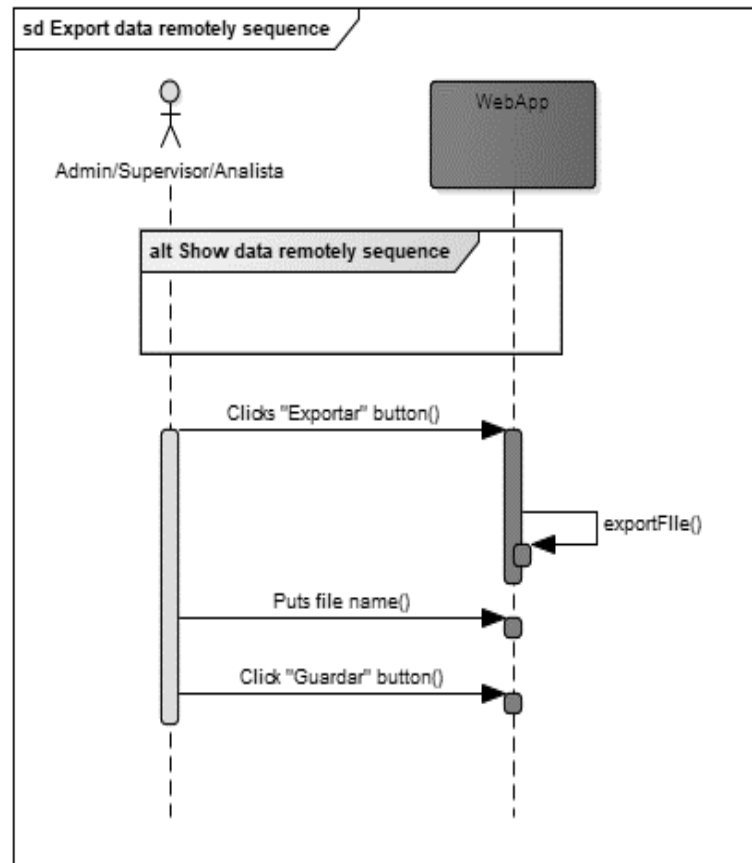


Figura 16: diagrama de secuencia que evidencia el workflow utilizado para exportar los datos guardados en el sistema remoto.

3. Diagrama general de componentes

Muestra como el sistema embebido tiene interacciones con el router de NodeJS y con el módulo creado de recepción de mensajes “MQTTBridge” del sistema backend por medio del sistema de caja negra que pertenece al bróker MQTT establecido, como el “ServicesHierarchy” permite exponer servicios que son utilizados por los endpoints del router de NodeJS y el módulo de MQTTBridge y como la WebApp alcanza sus funcionalidades por medio de su contacto con los API endpoints que expone el router, los “ServiceHierarchy” usan los Daos en su estructura esto puede verse más a detalle en el diagrama de componentes “ServiceHierarchy Model” (*ver figura 17*).

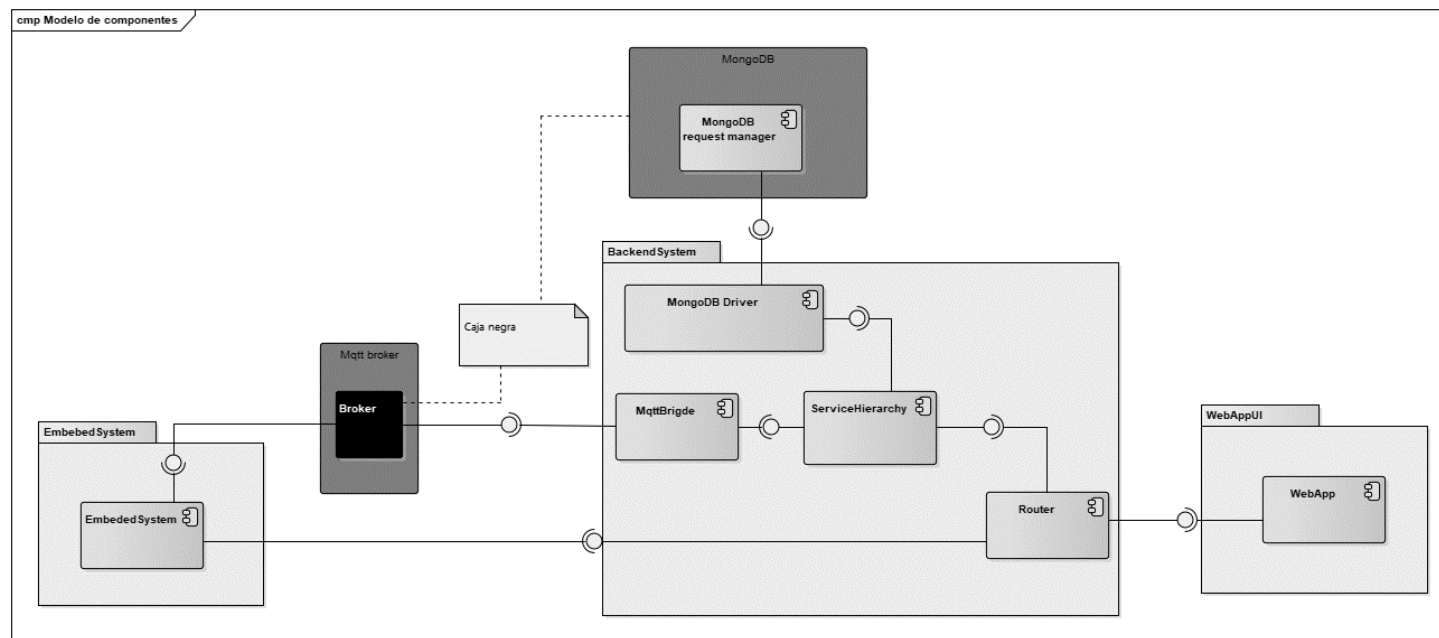


Figura 17: diagrama de componentes general de ambos sistemas, tanto el remoto como el físico en conjunto..

4. Diagrama de despliegue

En este se evidencia que el código C++ hecho en el framework de Arduino en PlatformIO se pasa al microcontrolador FireBeetle Esp32, que interacciona con la pantalla TFT Nextion HMI a través del protocolo serial (por los pines RX y TX definidos en ambos), con el bróker MQTT con puerto de comunicación 1883 y con endpoints en el backend a través de “Ngnynx” con el protocolo HTTP en el puerto 8080, que es usado también por el browser del usuario para descargar la WebApp, React está en modo proxy con NodeJS así que la asociación con “Ngnynx” favorece la exposición de los endpoints de manera automática (*ver figura 18*).

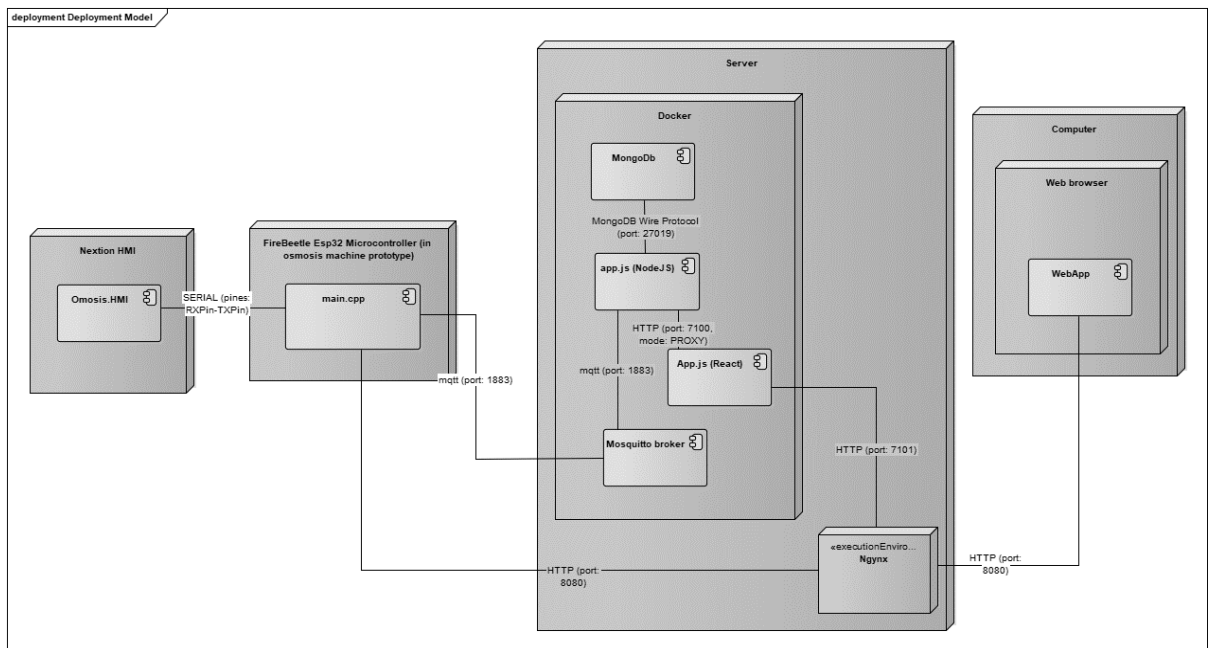


Figura 18: diagrama de despliegue que incluye tanto el sistema remoto como el físico.

Para el sistema electrónico (también llamado físico en todo el presente documento) se tienen algunos diagramas dispuestos en literales a continuación.

1. Una imagen holística representativa

Muestra de una manera más directa la organización del sistema electrónico, ideal para explicar su composición a las personas de manera un poco más fácil (**ver figura 19**)

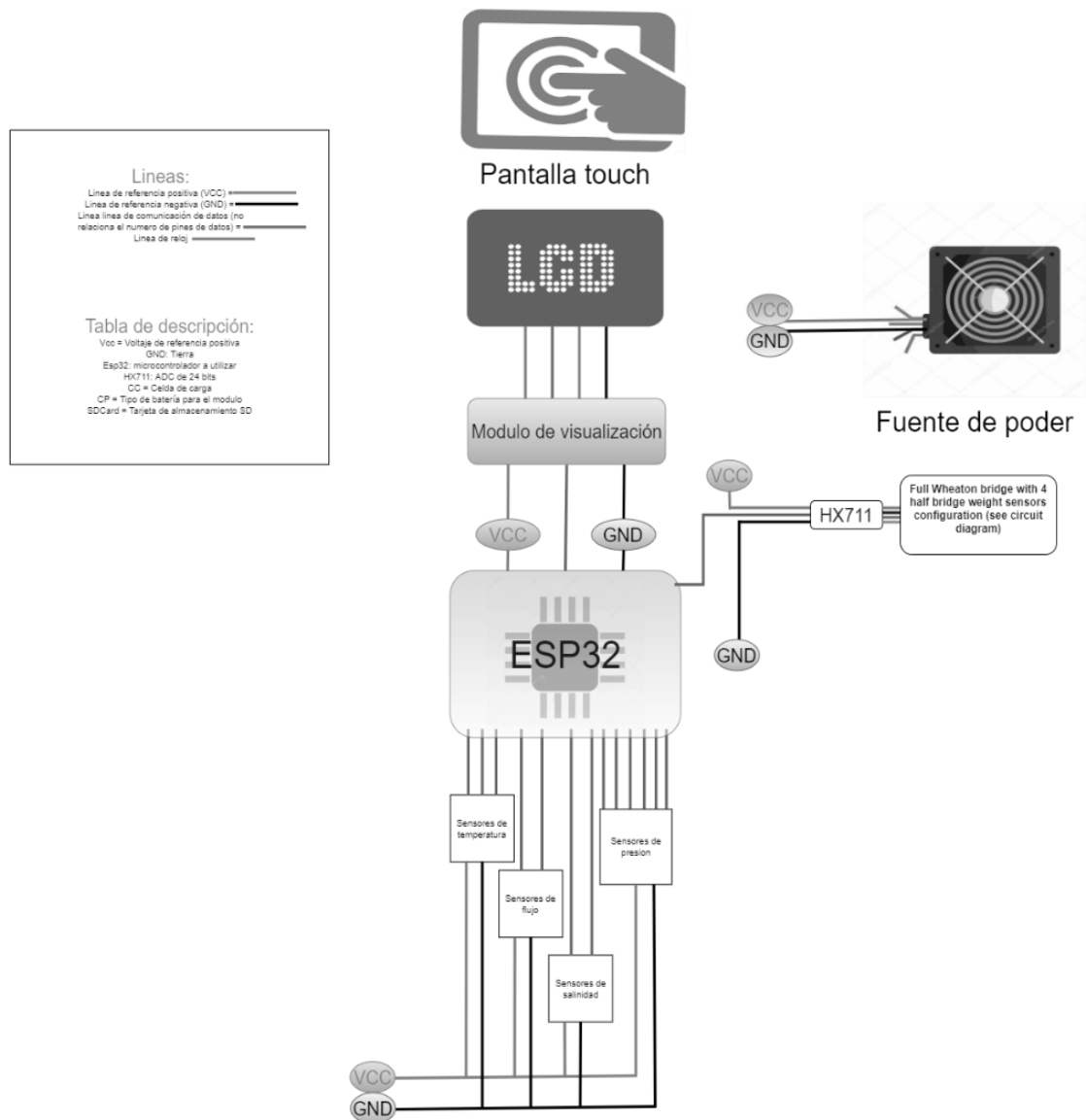


Figura 19: diagrama de general del sistema electrónico vista para usuarios e implementadores

2. Imagen representativa de composición del sistema

Presenta composición del sistema completo y cómo sus diferentes partes interactúan, para ello se divide en 3 bloques: el primero haciendo referencia a la distribución de los sensores en un esquema en la versión simplificada de un sistema hidráulico que opera bajo procesos de ósmosis (ver figura 20).

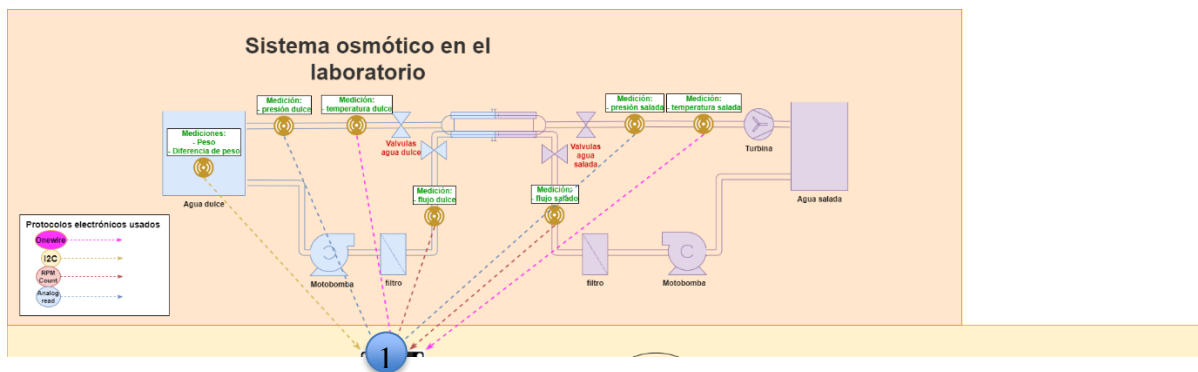


Figura 20: diagrama representativo de composición del sistema, subsistema hidráulico.

El segundo recibe la conexión de estos sensores y muestra como la información es dirigida hacia los usuarios finales (ver figura 21).

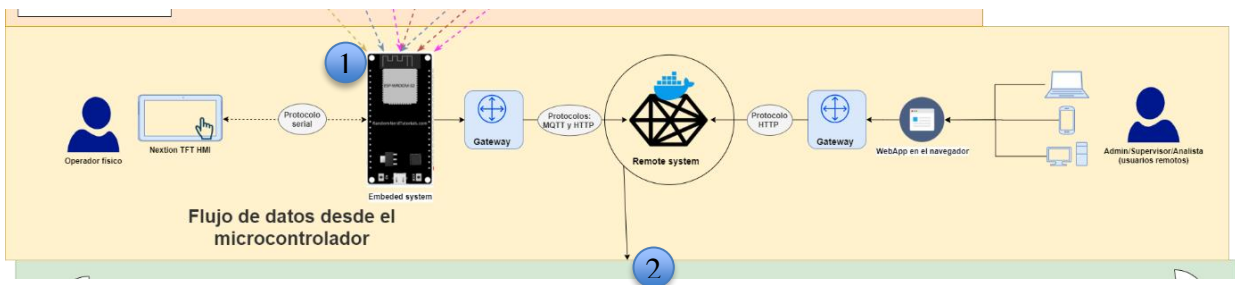


Figura 21: diagrama representativo de composición del sistema, subsistema de distribución de información.

Donde el tercero, como explicación de los módulos que componen al sistema remoto desglosa el sistema de contenedores docker donde se encuentra el sistema remoto distribuido (ver figura 22).

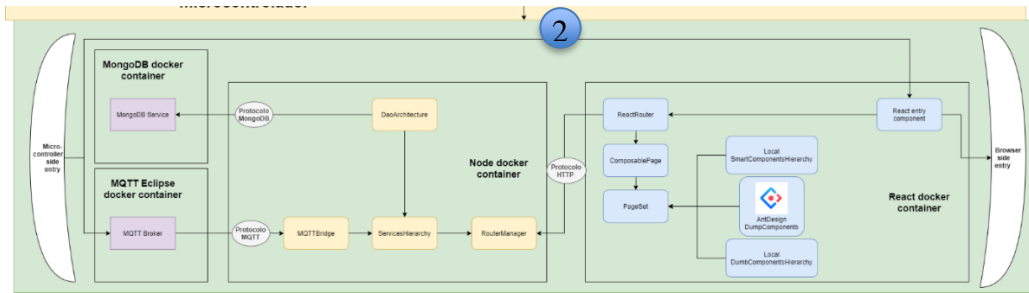


Figura 22: diagrama representativo de composición del sistema, porción de desglose del sistema remoto.

3. Diagrama electrónico de conexión

Muestra cómo están establecidas, a nivel de pines cableados, las conexiones con los sensores y la pantalla táctil para lograr formar el sistema (ver figura 23).

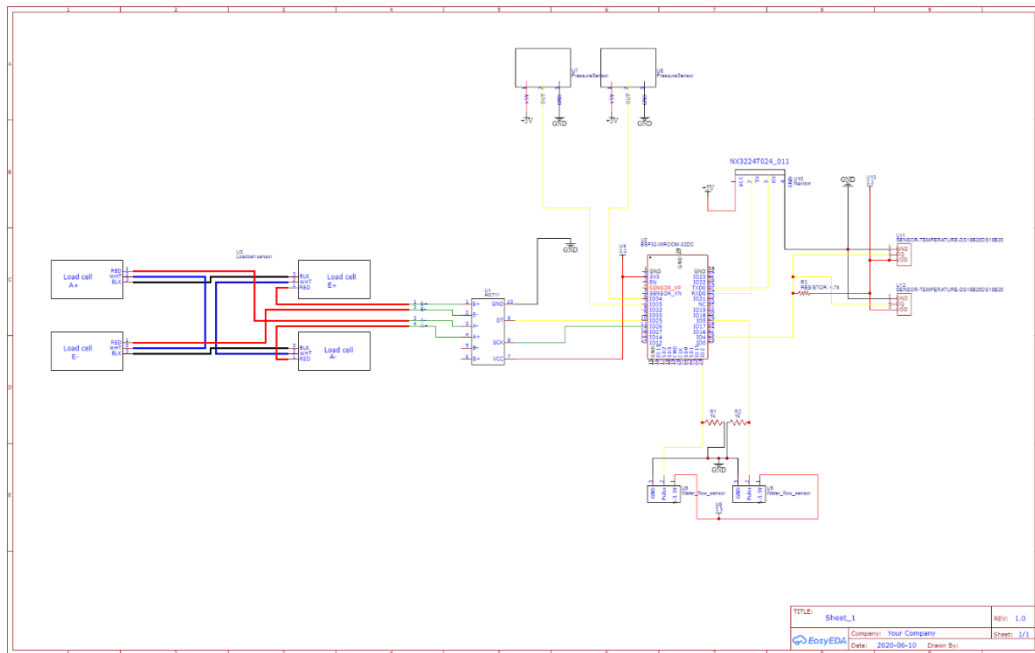


Figura 23: diagrama de electrónico del sistema físico, conexión circuital del mismo.

4. Diseños de la balanza iniciales del diseño base a seguir

Estos diseños fueron las propuestas iniciales de balanza que sirvieron para crear el primer prototipo de balanza y para el diseño de la balanza final (*ver figuras 24, 25 y 26*).

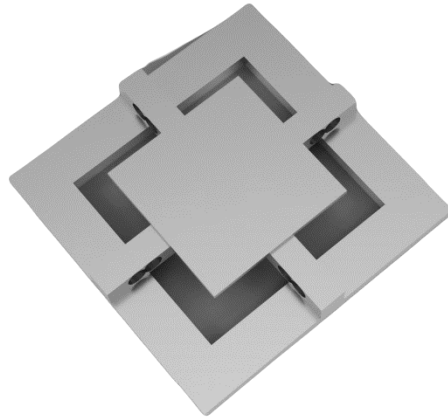


Figura 24: Propuesta de balanza simple vista aérea con superficie base.

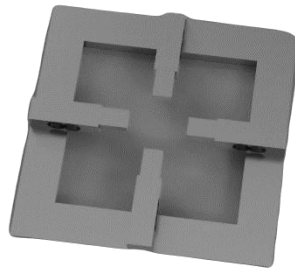


Figura 25: Propuesta de balanza simple vista aérea sin superficie base.

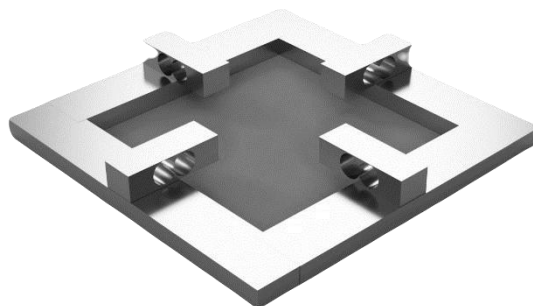


Figura 26: Propuesta de balanza solo sensores de fuerza y base elevador que permita la Torsión.

Con los diagramas anteriores se **cumplió** el **segundo objetivo** del presente.

Para el **tercer objetivo**, que busca la implementación del diseño realizado, se llevó a una plataforma y a un sistema electrónico que contienen la implementación de los requisitos funcionales, está primera solo le falta acordar los gráficos finales que irán en el Dashboard y la parte electrónica se logró implementar.

Este objetivo **se cumplió con** la entrega del **anexo 3 (manual de usuario)** y **anexo 4 (manual del sistema)**; además de las propias implementaciones, de las cuales se añaden referentes gráficos en forma de screenshots de la plataforma web (*ver figuras 26 a 34*) y que pueden ser accedidos por quien ve el presente visitando **<http://osmosis.semard.com.co>**, con credenciales nombre de **usuario**: johnjohn **contraseña**: DoeDoe78.

- La **figura 27** evidencia la interfaz de usuario para logear a la plataforma; además, se ofrece una pequeña descripción del objeto de su realización.

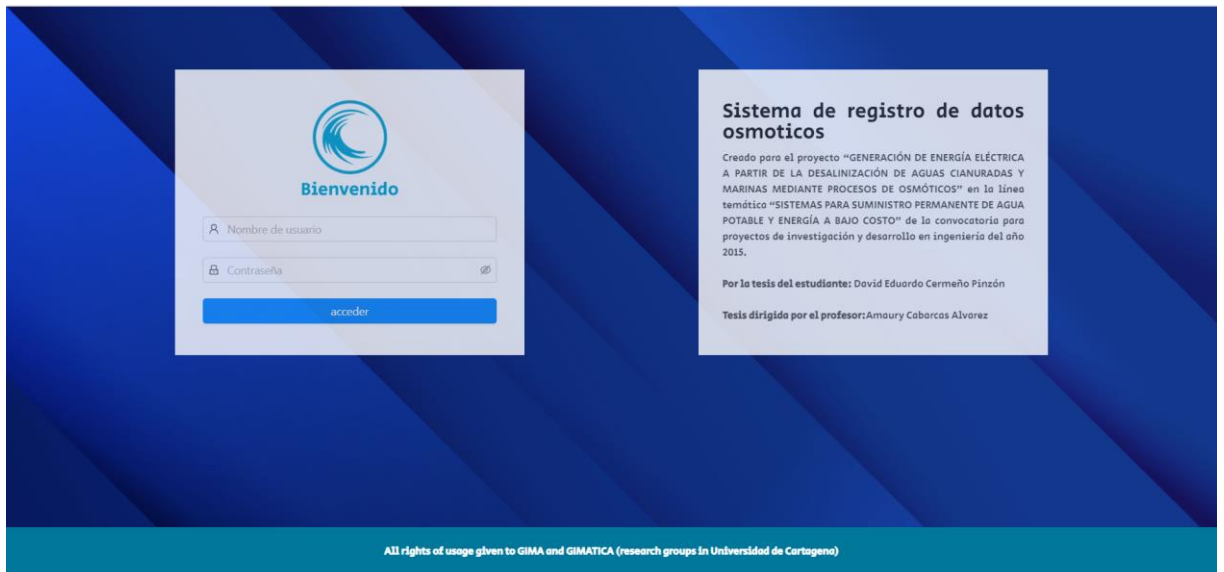


Figura 27: Plataforma web vista de acceso al sistema, también llamada vista de login, donde se muestra la estructura base de la página con su formulario e infografía base.

- En la **figura 28** se aprecia la vista de dashboard con gráficas y tablas que pueden programarse para ser pertinentes según las necesidades del sistema, en la imagen de ejemplo se muestra una tabla con máquinas activas e inactivas en el sistema y una gráfica que refleja la cantidad de set de mediciones tomados en promedio por todas las máquinas (que fueron las gráficas iniciales propuestas para el dashboard).

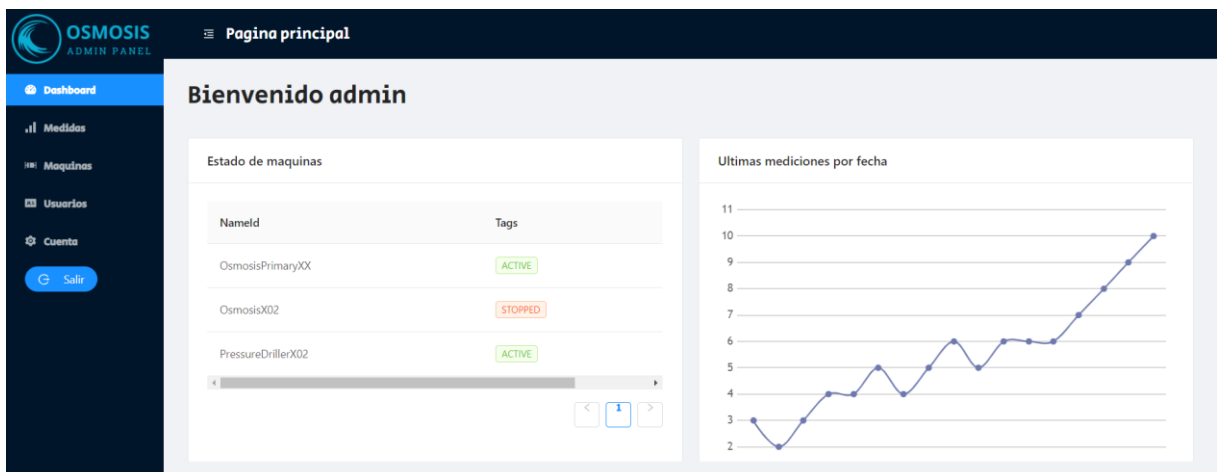


Figura 28: Plataforma web vista de dashboard, donde se muestran las gráficas que están evaluando los involucrados en el proyecto.

- En la **figura 29** se encuentra la vista de la primera parte de la página de medidas de la plataforma web, donde se encuentran listados los sets de medidas tomados anteriormente.

OSMOSIS ADMIN PANEL

Dashboard

Medidas

Maquinas

Usuarios

Cuenta

Salir

Historial de mediciones

Maquina involucrada	Progreso	Momento de inicio	Momento de finalizacion	Operaciones
OsmosisPrimaryXX	Terminada	7/11/2020 0:41:58	7/11/2020 0:44:51	Ver medidas Eliminar
OsmosisPrimaryXX	Terminada	7/11/2020 0:47:15	7/11/2020 0:51:54	Ver medidas Eliminar
OsmosisPrimaryXX	Terminada	7/11/2020 0:51:56	7/11/2020 0:56:56	Ver medidas Eliminar

Figura 29: Plataforma web vista de los sets/conjuntos de medidas tomados en el tiempo.

- En la **figura 30**, contiene la segunda parte de la página de medidas donde se muestran los datos específicos recogidos del set de lecturas seleccionado, mostrando la posibilidad de que se puede exportar.

OSMOSIS ADMIN PANEL

Dashboard

Medidas

Maquinas

Usuarios

Acciones de usuario

Errores del sistema

Cuenta

Salir

Visor de medidas

Identificador	Peso (gr)	Diferencia de peso (gr)	Presion salada (PSI)	Presion dulce (PSI)	Flujo salado (mL/minutos)	Flujo dulce (mL/minutos)	Temperatura salada (°C)	Temperatura dulce (°C)	Medida	Operaciones
1	21.025	1.64	0	0.408	0	0	31.5	-127	Inv	Eliminar
2	6407.909	6386.884	0	0.388	0.983	0	31.5	-127	Inv	Eliminar
3	2.604	-11.67	0	0.377	0	0	85	-127	Inv	Eliminar
4	22.471	19.867	0	0.4	0	0	31.438	-127	Inv	Eliminar
5	17.939	-4.533	0	0.397	0	0	31.438	-127	Inv	Eliminar
6	31.055	13.116	0	0.4	0	0	31.438	-127	Inv	Eliminar
7	13.02	-18.035	0	0.412	0	0	31.438	-127	Inv	Eliminar

All rights of usage given to GIMA and GIMATICA (research groups in Universidad de Cartagena)

Figura 30: Plataforma web vista de los sets medidas, donde se muestran las medidas los momentos base donde se tomaron datos y se ofrece la posibilidad de verlos detalladamente.

- La **figura 31** muestra el formulario que se necesita llenar para añadir nuevas máquinas en el sistema, presente en la página de gestión de máquinas.

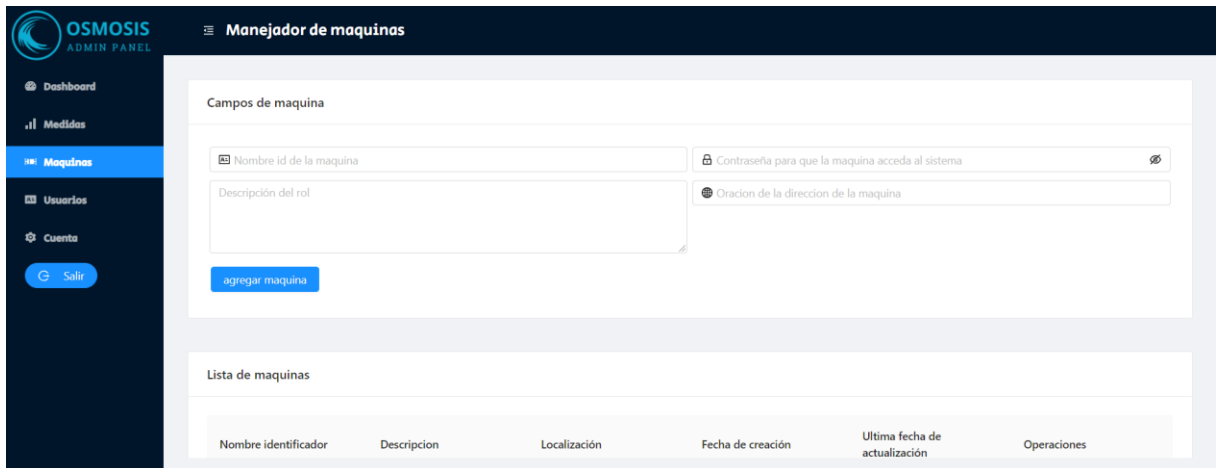


Figura 31: Plataforma web vista de administrador de máquinas con el formulario de creación y actualización de las mismas.

- La **figura 32** muestra la tabla de la página de gestión de máquinas, donde se encuentran, valga la redundancia, las máquinas registradas en el sistema y que de tener privilegios de usuario suficientes despliega también las respectivas opciones de actualización y eliminación correspondientes.

Nombre identificador	Descripción	Localización	Fecha de creación	Última fecha de actualización	Operaciones
OsmosisPrimary5XX	Maquina del laboratorio de GIMATICA, para pruebas prioritarias.	Piedra de Bolivar	31/10/2020 6:01:07	4/12/2020 19:58:02	Actualizar Eliminar
OsmosisPrimaryXX	Maquina del laboratorio de GIMATICA, para pruebas prioritarias.	Piedra de Bolivar	31/10/2020 19:59:51	4/12/2020 19:56:23	Actualizar Eliminar
OsmosisPrimary3XX	Maquina del laboratorio de GIMATICA, para pruebas prioritarias.	Piedra de Bolivar	31/10/2020 20:19:13	4/12/2020 19:56:32	Actualizar Eliminar
	Maquina del laboratorio de				Actualizar

Figura 32: Plataforma web vista de administrador de máquinas con la tabla del listado de máquinas registradas que inicia la eliminación o actualización de las mismas.

- La **figura 33** muestra el formulario que se necesita llenar para añadir nuevos usuarios remotos en el sistema, presente en la página de gestión de usuarios.

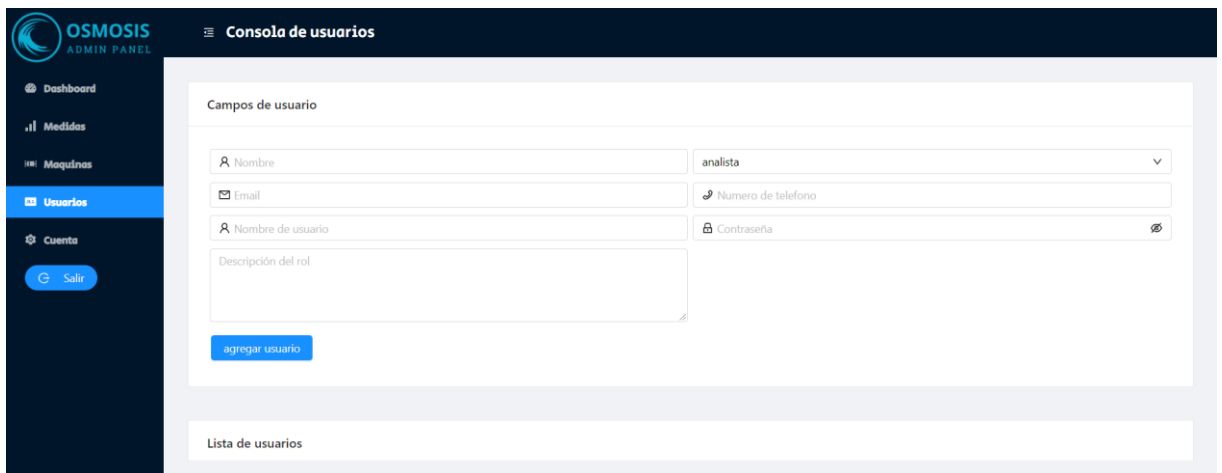


Figura 33: Plataforma web vista de administrador de usuarios con el formulario de creación y actualización de los mismos

- La **figura 34** muestra la tabla de la página de gestión de usuarios, donde se encuentran, valga la redundancia, los usuarios registrados en el sistema y que de tener privilegios de usuario suficientes despliega también las respectivas opciones de actualización y eliminación correspondientes.

Name	Role type	Role description	Email	Phone	Username	Actions
David Eduardo Cermeño Pinzón	analista	MixonMixon	daecepi@gmail.com	3174122546	dcermenop	Actualizar Eliminar
Michelle Montero	supervisor	Supervisora del sistema osmótico primario.	mmontero@gmail.com	57 3174445555	mmontero	Actualizar Eliminar
Michael Garcia	analista	garciagarcia	garcia@gmail.com	3174122547	garciagarcia	Actualizar Eliminar
Miguel Hernandez	analista	hernandezhernandez	hernandez@gmail.com	3174122546	hernandezhernandez	Actualizar Eliminar

Figura 34: Plataforma web vista de administrador de usuarios con la tabla con el listado de usuarios registrados que inicia la eliminación o actualización de los mismos.

- La **figura 35** muestra la pagina de actualización de datos de la cuenta logeada, que permite funciones básicas como actualizar los datos de la cuenta y el cambio de contraseña.

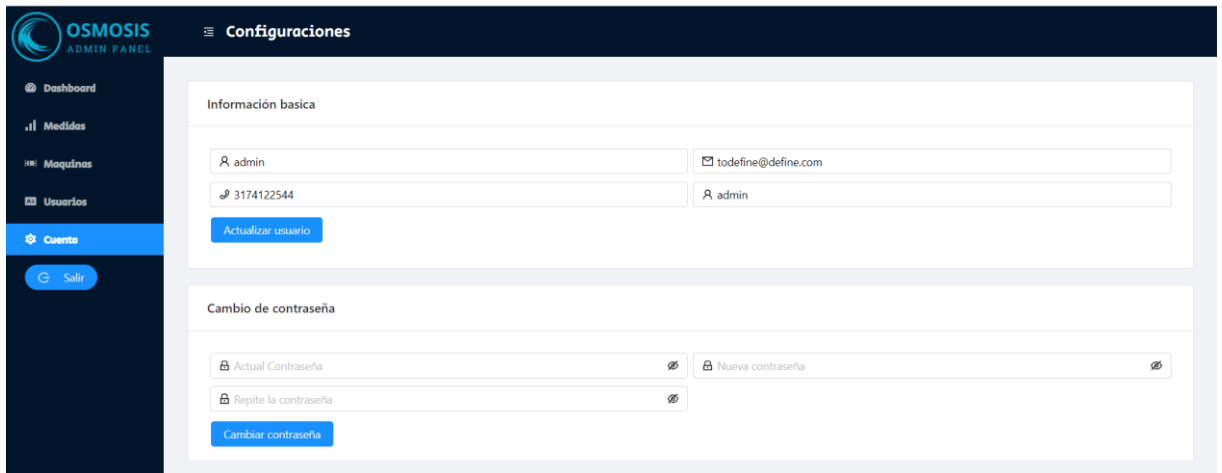


Figura 35: Plataforma web vista de configuraciones para actualizar la información del usuario actualmente logeado.

Para la interfaz de usuario local del sistema (o parte física) una apariencia tan cercana a la estructura común del sistema osmótico a nivel estructural como fuera posible, para ejemplo de esta (*ver figura 36*).



Figura 36: Foto del sistema físico implementado con el administrador del laboratorio Reberto Avila y el estudiante David Eduardo Cermeño Pinzón tomando medidas.

A continuación, algunas evidencias recogidas para el informe final de Colciencias y que soportan **tercer objetivo** del presente:

Implementación en protoboard de la conexión de la balanza y microcontrolador, monitorizado desde el computador (*ver figura 37*)

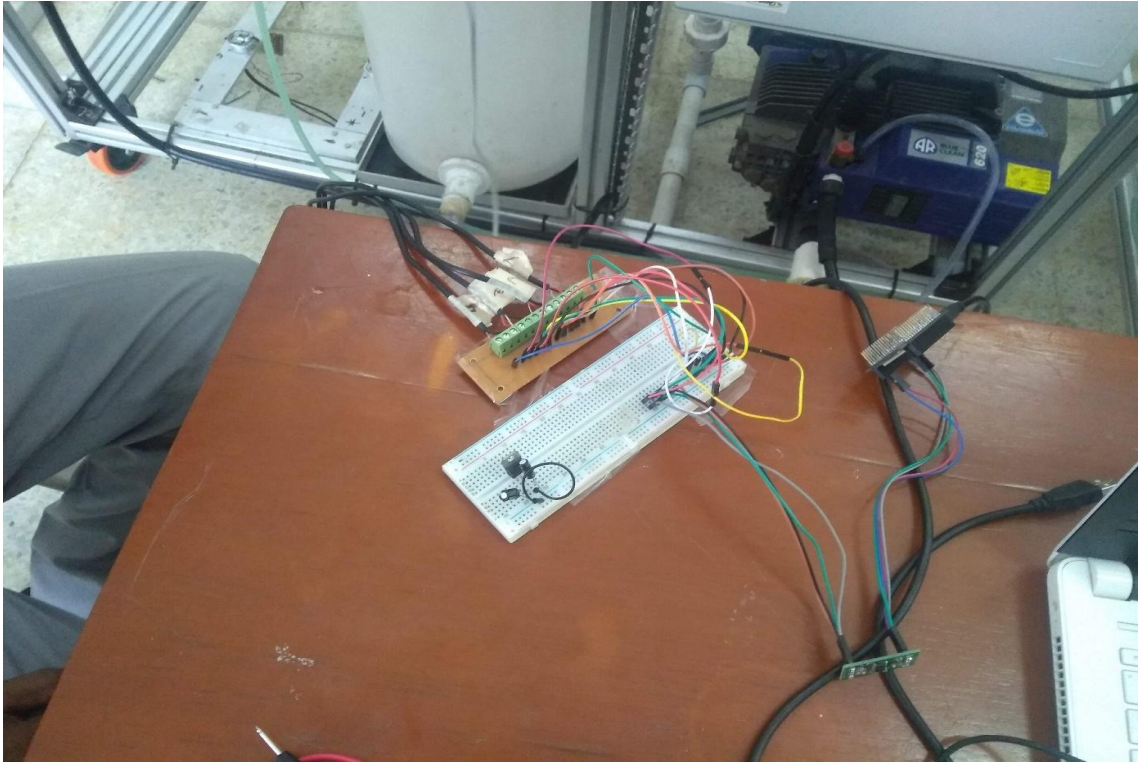


Figura 37: figura de primeras implementaciones del sistema (iteración 3) de recolección de datos en el laboratorio.

Sistema de balanzas con el prototipo de reguladores de voltaje para alimentar el sistema, incluye el pasó del cableado de los sensores de peso a baquelita de soldado (*ver figura 38*).

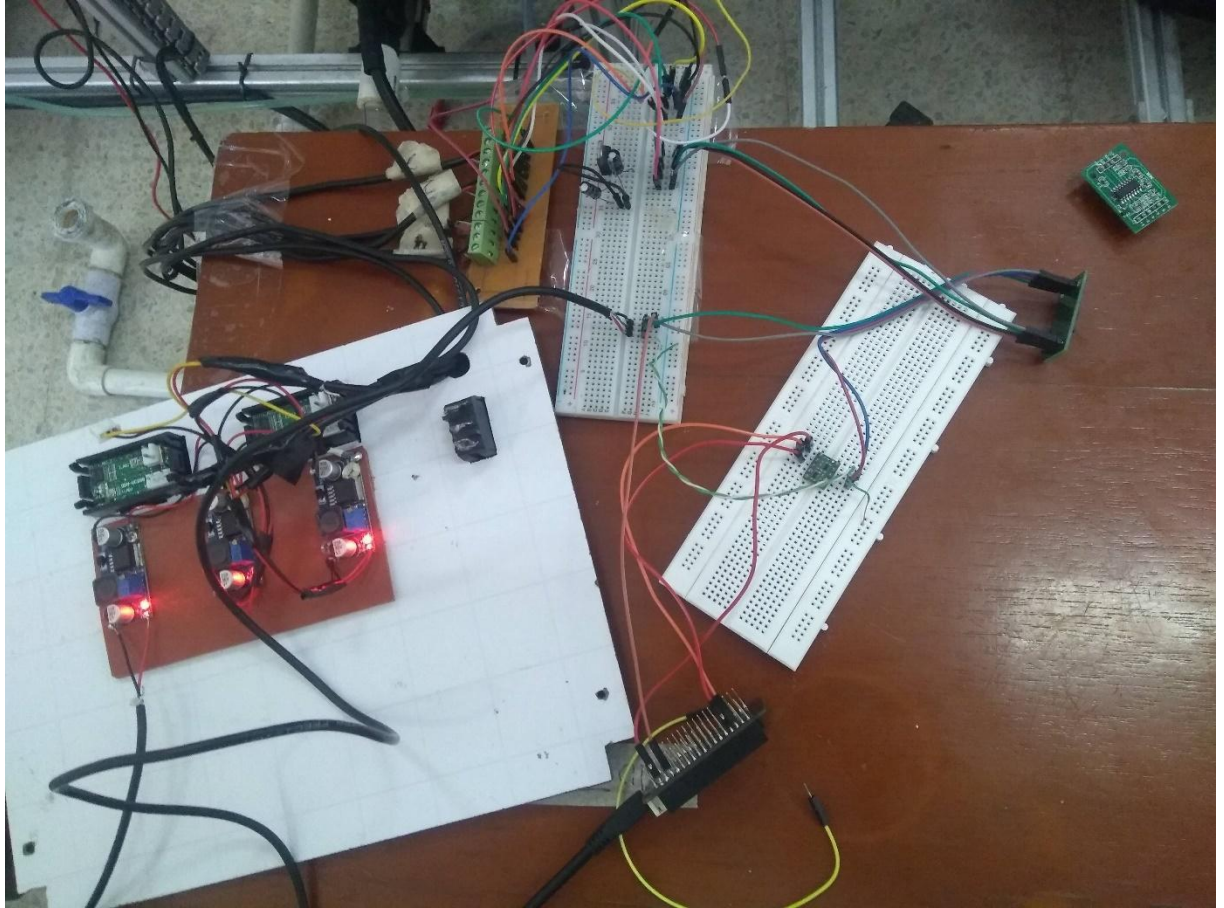


Figura 38: foto de primeras implementaciones del sistema (en iteración 3) de recolección de datos en el laboratorio.

Sesión de programación de scripts de calibración para los sensores (*ver figura 39*).



Figura 39: foto de primeras implementaciones del sistema de recolección de datos (en iteración 10) en el laboratorio.

Cableado de los demás sensores en protoboard para probar conexiones ángulo fotográfico 1, evidencias dadas para entrega a Colciencias (*ver figura 40*)



Figura 40: foto de las implementaciones en etapa mediana del sistema de recolección de datos (en iteración 10) en el laboratorio.

Cableado de los demás sensores en protoboard para probar conexiones ángulo fotográfico 2, evidencias dadas para entrega a Colciencias (*ver figura 41*)



Figura 41: foto de las implementaciones en etapa mediana del sistema de recolección de datos (en iteración 10) en el laboratorio.

Cableado de los demás sensores en protoboard para probar conexiones ángulo fotográfico 3, evidencias dadas para entrega a Colciencias (*ver figura 42*)

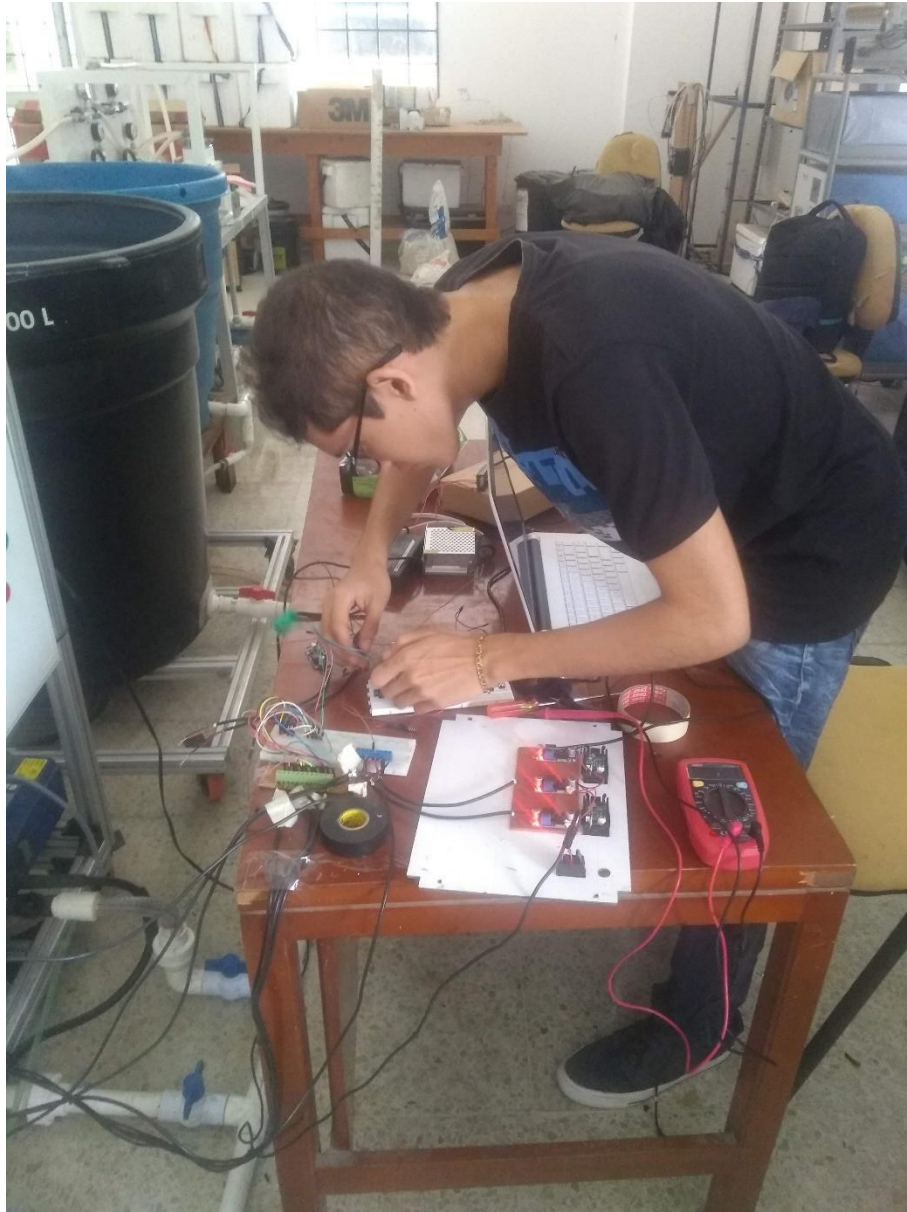


Figura 42: foto de las implementaciones en etapa mediana del sistema de recolección de datos (en iteración 10) en el laboratorio.

Mandando nuevo código para pruebas en el sistema de protoboard, evidencias dadas para entrega a Colciencias (*ver figura 43*)



Figura 43: foto de las implementaciones en etapa mediana del sistema de recolección de datos (en iteración 10) en el laboratorio.

Muestra de encendido del sistema desde la fuente eléctrica en fase de protoboard, evidencias dadas para entrega a Colciencias (*ver figura 44*)



Figura 44: foto de las implementaciones en etapa mediana del sistema de recolección de datos (en iteración 10) en el laboratorio

Presentación preliminar breve del sistema a los encargados de operar la máquina, evidencias dadas para entrega a Colciencias (*ver figura 45*)



Figura 45: foto de las implementaciones en etapa mediana del sistema de recolección de datos (en iteración 10) en el laboratorio.

Acercamiento al sistema preliminar hecho con protoboard y dos baquelitas, evidencias dadas para entrega a Colciencias (*ver figura 46*)



Figura 46: foto de las implementaciones en etapa mediana del sistema de recolección de datos (en iteración 10) en el laboratorio.

Preparación del sistema para su pasado electrónico a baquelitas, evidencias dadas para entrega a Colciencias (*ver figura 47*)

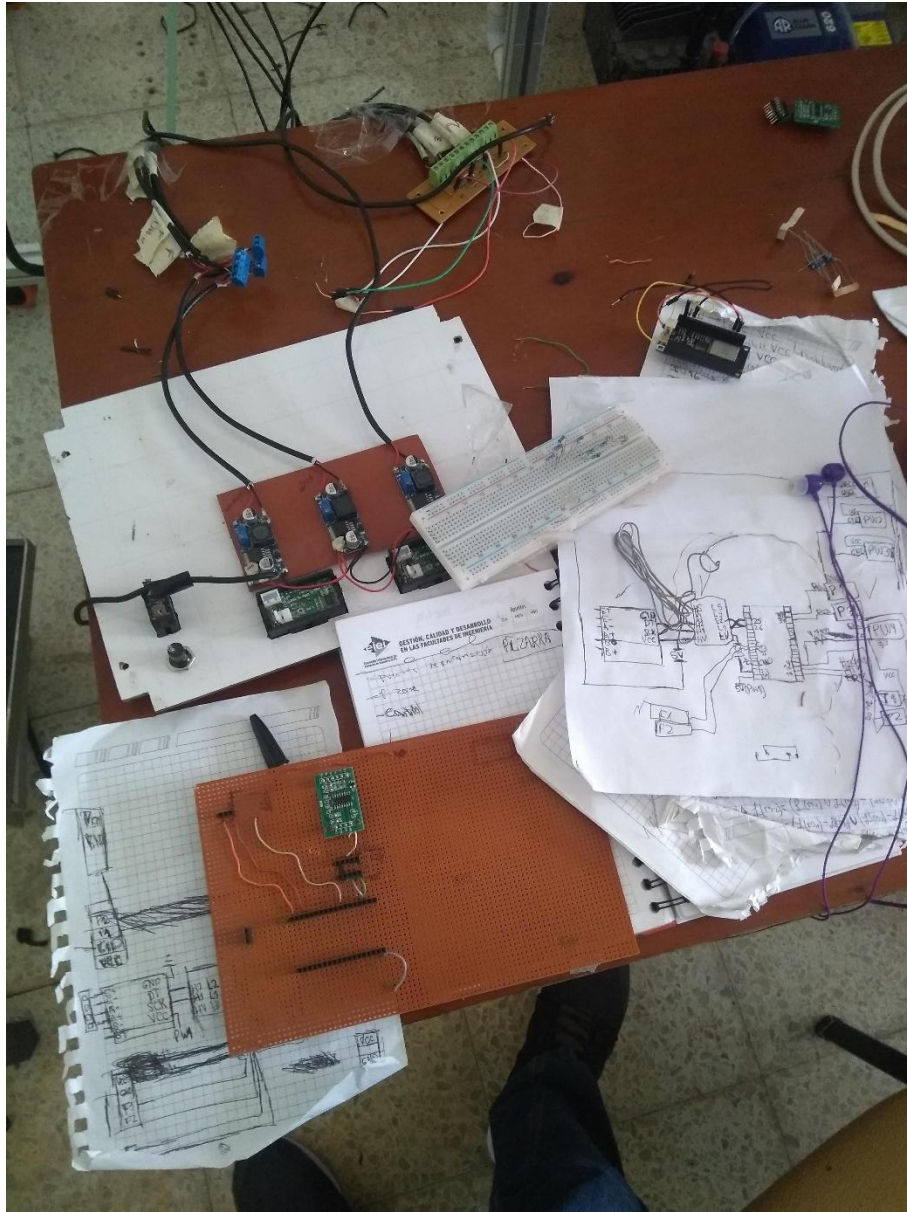


Figura 47: foto de las implementaciones en etapa de finalización del sistema de recolección de datos (en iteración 10) en el laboratorio.

Etapa de actualización del código preliminar, evidencias dadas para entrega a Colciencias (*ver figura 48*).

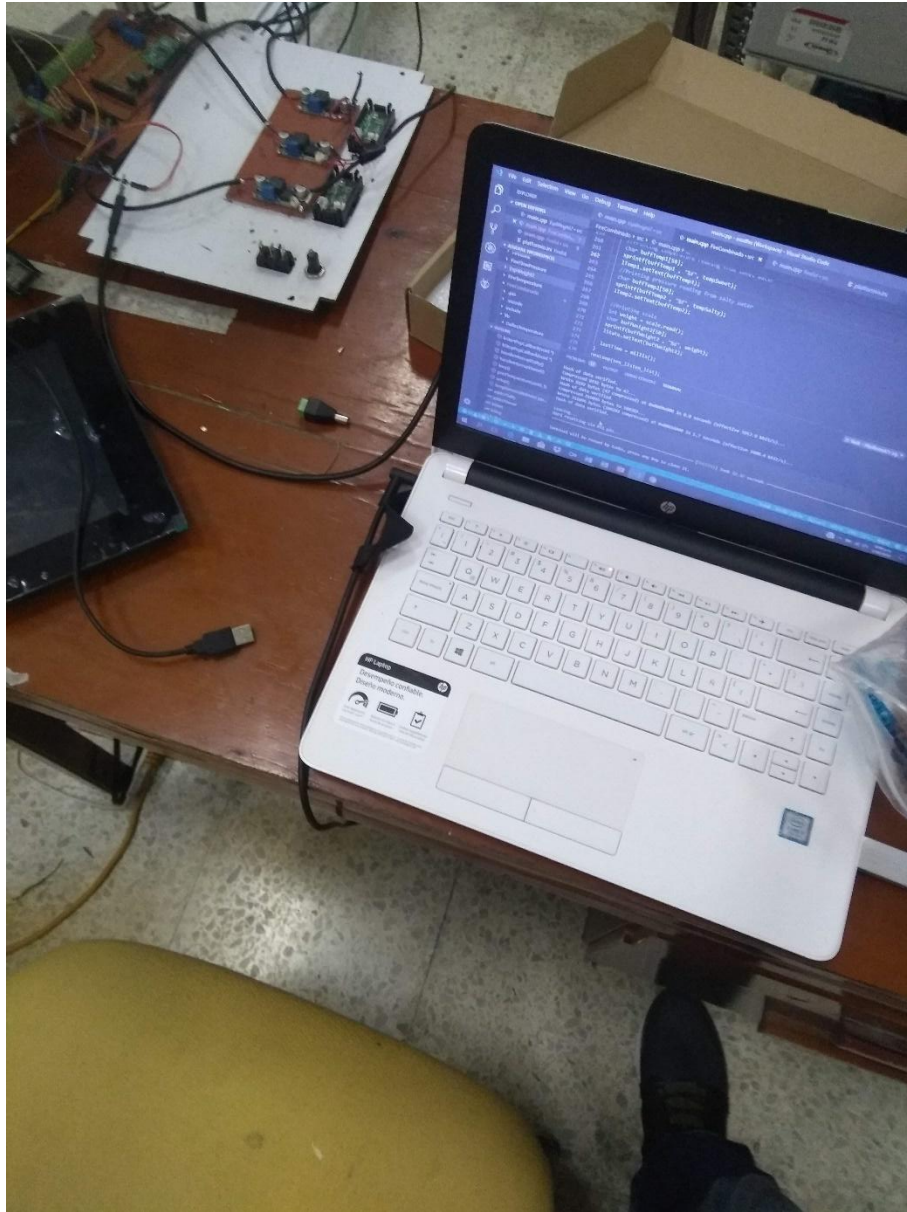


Figura 48: foto de las implementaciones en etapa mediano avanzada del sistema de recolección de datos (en iteración 11) en el laboratorio.

Integración de la pantalla al código y al sistema de alimentación, evidencias dadas para entrega a Colciencias (*ver figura 49*).

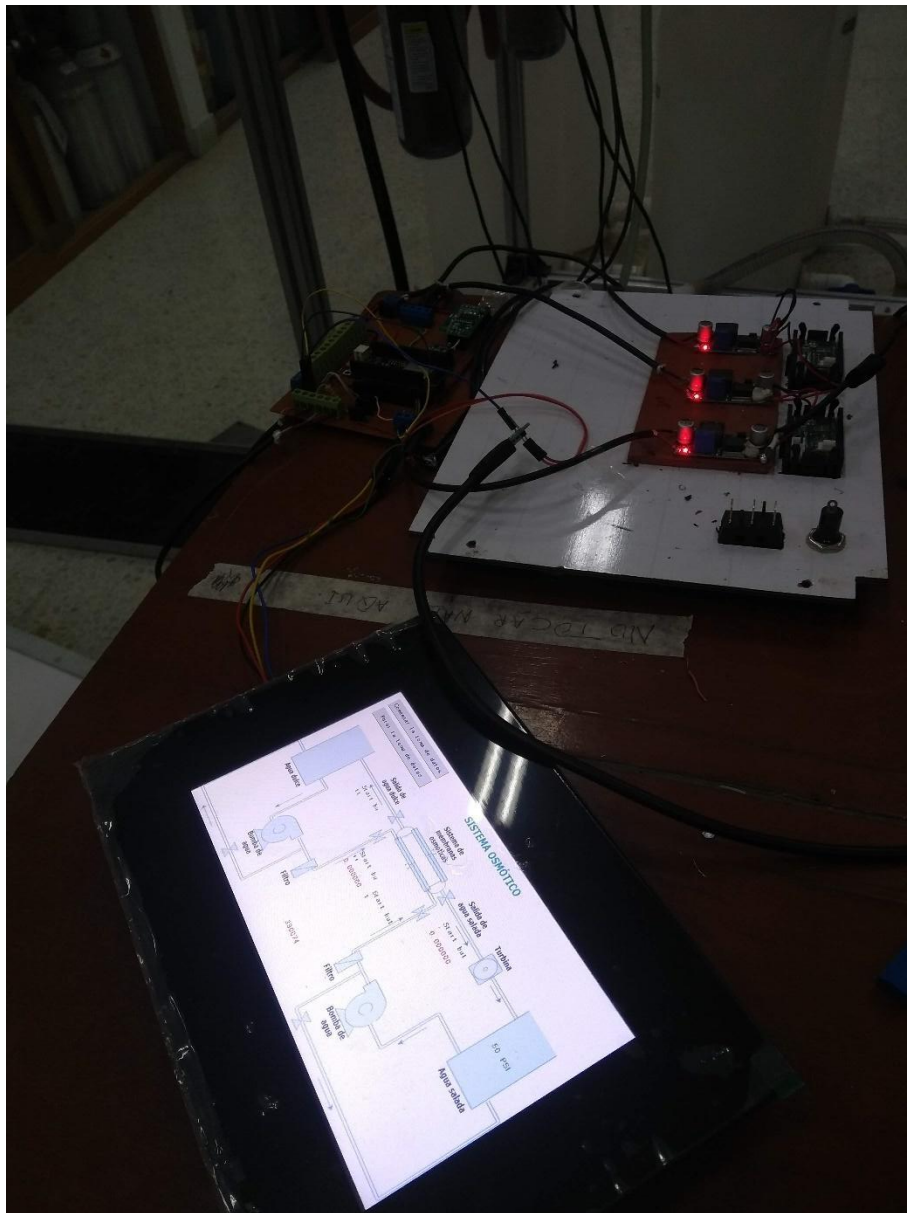


Figura 49: foto de las implementaciones en etapa avanzada del sistema de recolección de datos (en iteración 12) en el laboratorio.

Etapa de organización del sistema electrónico dentro del prototipo osmótico del grupo de investigación GIMA, posicionamiento de la pantalla, evidencias dadas para entrega a Colciencias (*ver figura 50*).



Figura 50: foto de las implementaciones en etapa avanzada del sistema de recolección de datos(iteración 15) en el laboratorio.

Etapa de organización del sistema electrónico dentro del prototipo osmótico del grupo de investigación GIMA, posicionamiento de la caja principal, evidencias dadas para entrega a Colciencias (*ver figura 51*).

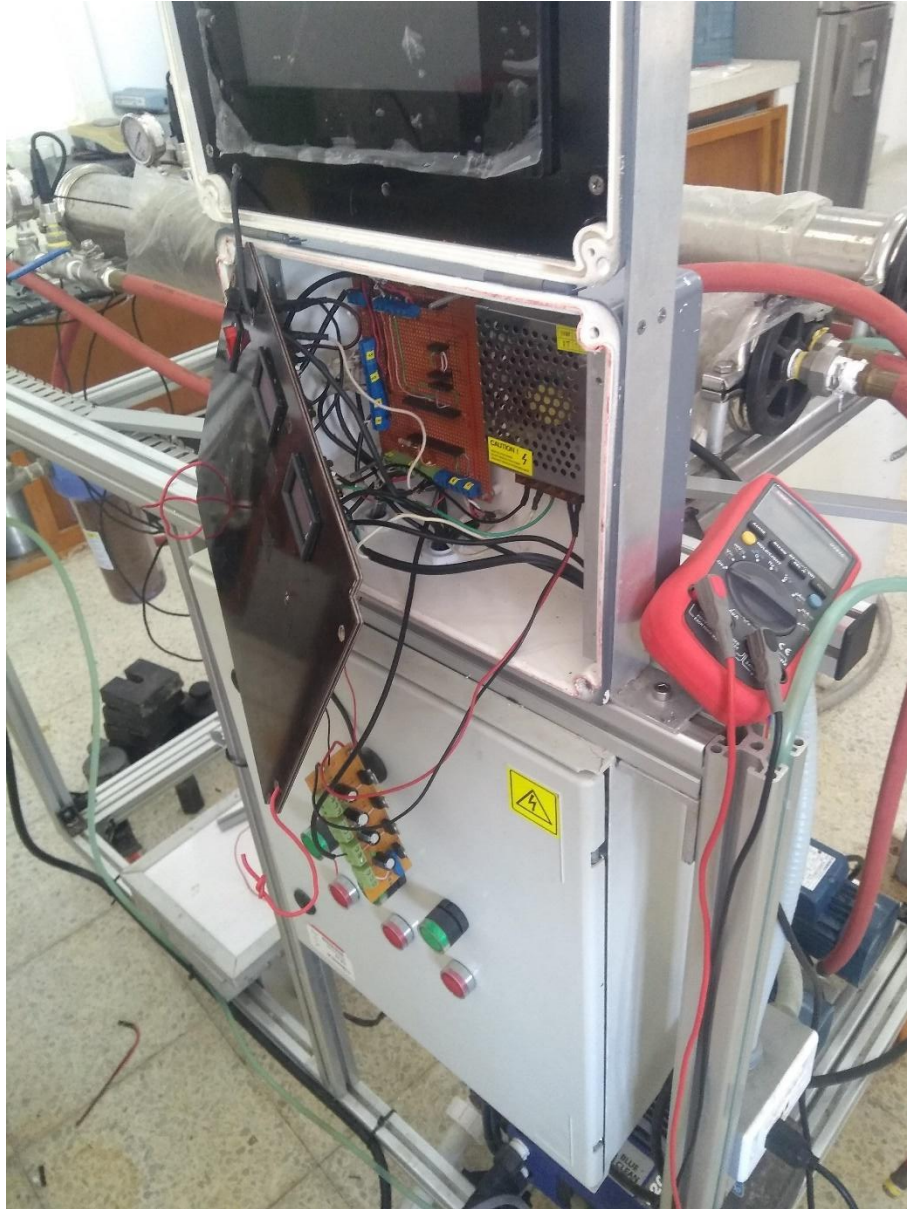


Figura 51: foto de las implementaciones en etapa avanzada del sistema de recolección de datos (en iteración 16) en el laboratorio.

Prueba de encendido del sistema completo luego de su integración, evidencias dadas para entrega a Colciencias (*ver figura 52*).

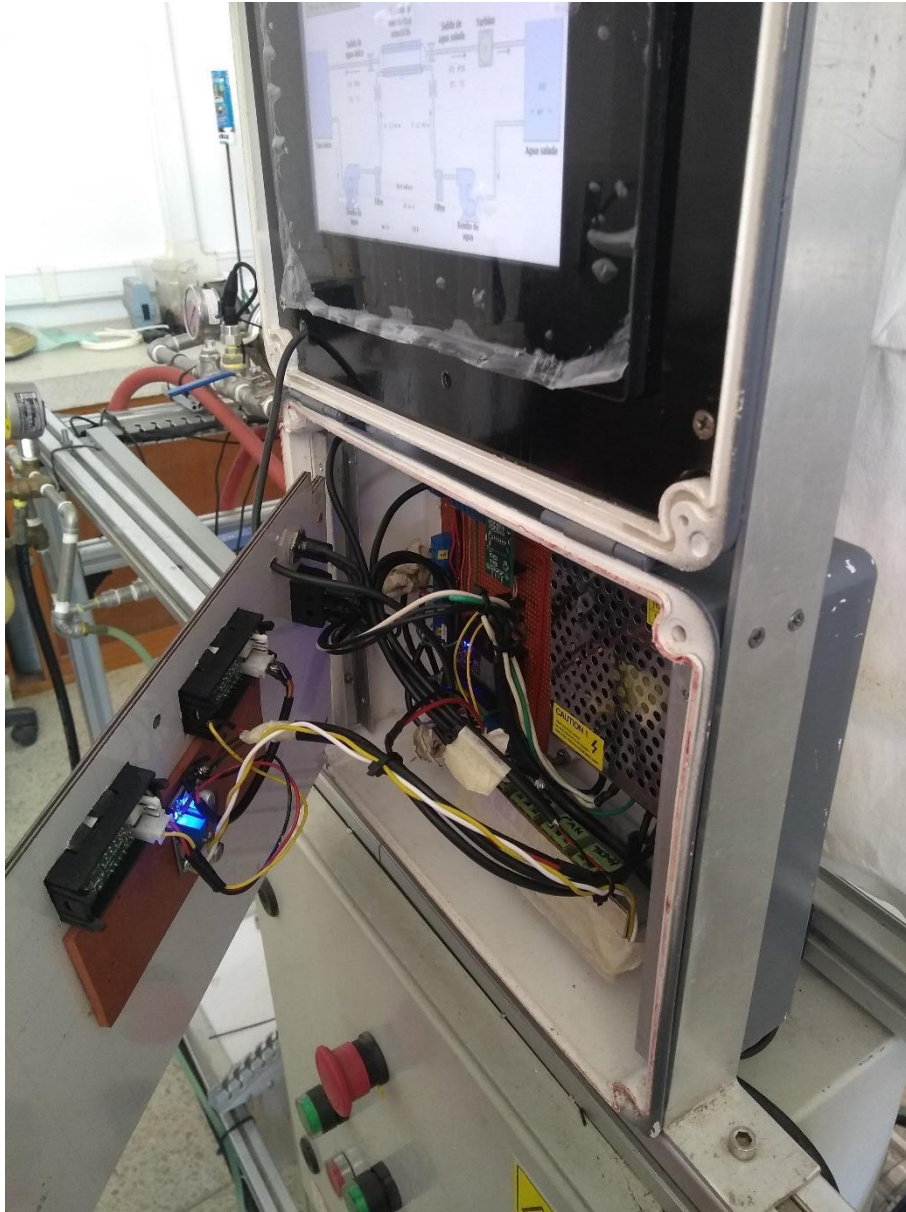


Figura 52: foto de las implementaciones en etapa avanzada del sistema de recolección de datos (en iteración 16) en el laboratorio.

Vista frontal con tapa de la caja principal del sistema abierto evidencias dadas para entrega a Colciencias (*ver figura 53*).



Figura 53: foto de las implementaciones en etapa avanzada del sistema de recolección de datos (en iteración 16) en el laboratorio.

Foto de enfoque al prototipo de balanza final implementado, evidencias dadas para entrega a Colciencias (*ver figura 54*).



Figura 54: foto de las implementaciones en etapa avanzada del sistema de recolección de datos (en iteración 16) en el laboratorio.

Enfoque al cableado organizado que sube a la caja principal del sistema, evidencias dadas para entrega a Colciencias (*ver figura 55*).



Figura 55: foto de las implementaciones en etapa avanzada del sistema de recolección de datos (en iteración 16) en el laboratorio.

Sistema con enfoque a varios sensores y balanza descubierta, evidencias dadas para entrega a Colciencias (*ver figura 56*).



Figura 56: foto de las implementaciones en etapa avanzada del sistema de recolección de datos (en iteración 16) en el laboratorio.

Foto más reciente del sistema encendido, evidencias dadas para entrega a Colciencias (ver figura 57).

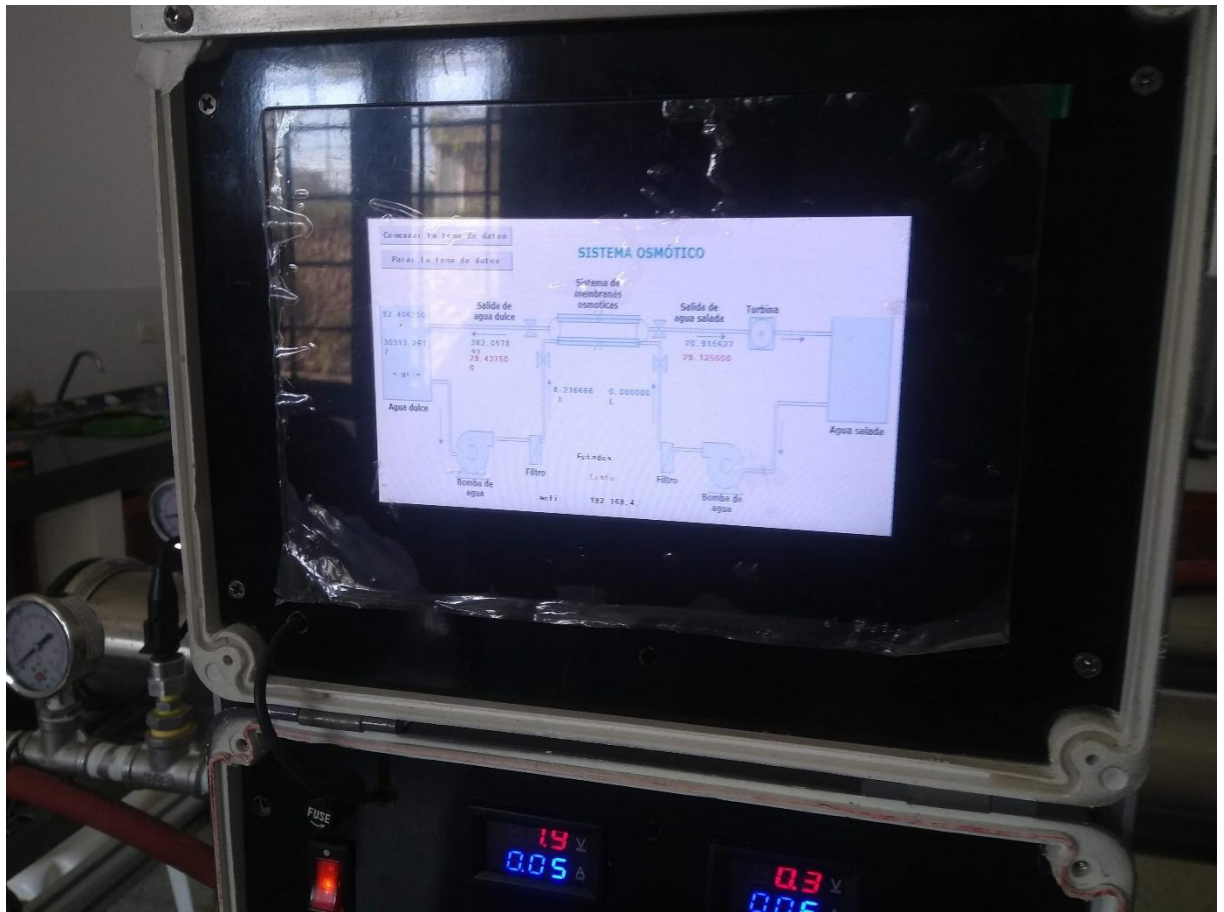


Figura 57: foto de las implementaciones en etapa avanzada del sistema de recolección de datos (en iteración 16) en el laboratorio.

Estas evidencias también fueron dejadas en forma de archivos en la carpeta de anexos de la presente entrega en el **anexo 5** y con su entrega se **cumplió el tercer objetivo**.

Del **cuarto objetivo**, se **cumplió con** la carta del semillero de hidráulica que certifica las pruebas que se hicieron en el sistema físico y el prototipo generador de CSV (**estos archivos generados se pueden ver en la carpeta anexos identificado como anexo 6**), ya que cuando se utilizó siempre estaban presentes ya sea el encargado de los laboratorios (designado por el líder del proyecto) o un encargado de operar la máquina y dándose además a los interesados del proyecto a la plataforma desplegada en el dominio: **osmosis.semard.com.co**, además en la carpeta de anexo se dejan una serie de .txt con los datos copiados de las pruebas unitarias a sensores y totales del sistema en la carpeta **anexos (bajo las carpetas de anexo 6 y 7)**.

Los tipos de pruebas abarcadas incluyen

- Pruebas unitarias: al implementar un nuevo sensor este se calibraba y sus medidas eran cotejadas con las de sensores industriales ya calibrados, lo que llevo a la generación de scripts individuales para probar estos sensores.
- Prueba de integración: antes de enviar el informe a Colciencias se dedicó una noche a tomar medidas bajo distintos tipos de presiones, dichas medidas iban siendo analizadas por el encargado de supervisar el proceso osmótico, quien encontró que como problema había algunos saltos que coincidían al momento de prender la motobomba con mayor poder, se mencionó la espera de un investigador de Canadá que pudiera analizar más a fondo la información obtenida por la máquina.
- End to end test (solo en el sistema remoto): para mostrar que la plataforma funcionaba, se probó manualmente las funciones que ofrecía y se le asignaron credenciales a un encargado para que la revisará y diera un feedback de que más se le podía anexar, el encargado menciona que se agregará el cálculo de la densidad de potencia y graficas relacionadas a la información obtenida directamente en la sección de medidas.

10. CONCLUSIONES

A medida que ha avanzado el tiempo se ha vuelto más accesible y los productos electrónicos tienen mayor disponibilidad a través de distintas tiendas/centros de ventas y en materia de sensores se pueden encontrar a menor costo y más precisión, lo anterior también aplica para el hardware libre, pero se debe mencionar que aunque la grieta de accesibilidad y eficiencia está decreciendo las tecnologías propietarias siguen manteniendo mayores estándares que pueden agilizar el tiempo de implementación de diversos prototipos.

El sistema creado presenta un nivel de sensibilidad de 1gr y velocidad de lectura de poco menos de un segundo en los sensores de pesos instalados al calibrarse con pesas de precisión, los sensores de temperatura con un error de medición del $+1^{\circ}\text{C}$ al compararse con sensores calibrados de temperatura, los sensores de flujo con resultados de flujo muestran un caudal similar al compararse con sensores de flujo calibrados y sensores de presión de error de ± 2 PSI al compararse con sensores de presión calibrados.

La periodicidad de captura de datos es adecuada esto fue evidenciado en algunas de las sesiones de tomas de datos que se hicieron con los encargados de operar la máquina, algunas gráficas deben todavía acordarse y para las falencias encontradas en el sistema obtenido ya se tiene una solución en espera de implementarse pasadas las medidas de la cuarentena.

La cuarentena mencionada también tiene en espera la visita de un investigador especializado en procesos osmóticos que vendrá a evaluar el sistema osmótico construido y el sistema de recolección de datos, para realizar análisis pertinentes y proponer posibles mejoras estructurales a los mismos.

Además, por cada objetivo se puede concluir que para los requisitos el medio más rápido para encontrar asimilar que conocimiento y problemas que estaban teniendo los involucrados del proyecto fueron las reuniones y con ellos las entrevistas y preguntas constante durante el desarrollo, al ser compañeros de grupo dentro del trabajo siempre estuvimos compartiendo información de cómo iban nuestras partes.

Luego, el diseño, aunque más complejo debido la inclusión de dos tipos de sistemas para un solo proyecto, gracias a la versatilidad del lenguaje UML se hizo muy manejable mostrar sus conexiones; es decir, que como fue hecho a nivel estructural un diagrama circuital siempre va a ejemplificar mejor la parte electrónica, requiriendo entonces de ambos enfoques para modelar el sistema final.

En cuanto al enfoque de seguridad y mejoras que pudieran completar el prototipo hacen que aunque se haya ahorrado tiempo, utilizando algunas librerías y frameworks de desarrollo, todas las calibraciones de los sensores, el aislamiento por medio del diseño de condiciones de interferencia en los sectores de toma de medidas (como las vibraciones cerca de los sensores de fuerza), la distribución del ciclo del reloj para sacar provecho a la frecuencia en que corría, los módulos de seguridad para detección de errores, de intrusión de usuarios y logging de acciones de usuario, el sistema de manejo de usuarios y máquinas y el análisis de todo el workflow fomentarán la dificultad de su implementación (y diseño).

La parte de las pruebas entre otras incluyó la correlación con sensores ya calibrados utilizados en la industria y la correlación de los datos con personal experto, estos fueron indispensables para el módulo de recolección de datos.

El producto software final tiene tan pocas diferencias con muchos sistemas automatizados (que son cada vez más frecuentes) al punto que su única similitud son el hecho de automatizar la supervisión y recolección de datos de un proceso, algunos ejemplos de estos son “Planificación automatizada del arranque de generadores para la restauración de sistemas eléctricos de potencia” (Aleagaloaiza et al., 2015), “Sistemas SCADA para la automatización de los procesos productivos del CIGB. Ingeniería Electrónica, Automática y Comunicaciones” (Bárzaga Martell et al., 2016), “Sistema automatizado para la vigilancia epidemiológica de la tuberculosis en la atención primaria en Cuba” (Gelabert Machín & Pérez González, Edilberto. Ochoa Armas, 1995), “Sistema automatizado para la organización racional del proceso cosecha-transporte de cereales (SAORCE)” (Yanoy et al., 2014) y “Diseño de un sistema de automatización para la planta de alimento ensilado “Héctor Molina”” (Silva-Díaz et al., 2017).

Por otra parte, de los sistemas automatizados que se enfocan en la ósmosis como proceso objetivo, pero solo en una de sus variaciones (la ósmosis inversa, un proceso que ya lleva más recorrido a nivel investigativo) algunos de los cuales son La compilación de plantas industriales de desalinización de Kim y su equipo (Kim et al., 2009) y el proyecto acerca del uso de sistemas computacionales para automatizar y controlar procesos de plantas de ósmosis inversa (Zilouchian & Jafar, 2001).

Durante la realización del proyecto ocurrieron algunos resultados inesperados que influyeron en la terminación del presupuestada del proyecto en la terminación a pronta del proyecto:

- La quema de una pantalla durante uno de los mantenimientos del sistema, sucedido, debido a un cambio de polaridades en la corriente de entrada durante un mantenimiento al sistema hidráulico.
- La reutilización de la medida de flujo que, en tuberías llenas de agua a presión fuerte y constante, se convierte en un sistema de detección de cambios de presión de agua antes que un medidor de flujo.

11. RECOMENDACIONES

- En estos casos los sensores de hardware libre proveen una forma de bajo costo para realizar la investigación y con los algoritmos adecuados y un buen nivel de precisión en el ADC (siempre que lo permitan también los sensores) se puede reducir el ruido de las lecturas, pero siempre es mayormente recomendable utilizar sensores de tecnología propietaria que permiten reducir la cantidad de trabajo de calibración y agilizar el proceso de desarrollo, aumentando los costos del proyecto, esto es aplicable a todos los sensores.
- Se recomienda para futuras investigaciones si se va utilizar el módulo HX711 y el microcontrolador ESP32 o ESP8266 en algunas de sus versiones, se recomienda utilizar un oscilador externo que agilice el “sample rate” de las lecturas, lo que permitirá usar el microcontrolador elegido a una mayor frecuencia sin alterar la veracidad de las lecturas del HX711.
- Cuando el sistema incluye o está cerca de elementos industriales que generan un alto campo electromagnético, se recomienda recubrirlos con una jaula de Faraday o crear un sistema de aislamiento para el sistema, aunque permitiendo la conexión con el sistema final.
- El cambio de sensores requiere de calibración, por lo que se recomienda utilizar medidas de sensores ya calibrados para cotejar las medidas que se están obteniendo después de obtener un algoritmo base con las especificaciones del datasheet.
- Se recomienda que las actualizaciones de código o estructura del sistema electrónico involucren conocedores del proceso o una capacitación previa, para obtener conocimiento cimentado que permite tomar decisiones eficientes.
- Dentro de la investigación también se tuvieron limitantes importantes a tener en cuenta, presentadas en subpuntos a continuación.
 - Debido a que cada sensor tenía que ser posicionado en el punto específico de medición y los sistemas hidráulicos son susceptible a escapes de agua (algo que no es bueno para los mismos) sobre todo cuando tienen muy altas presiones como en el presente los reemplazos de sensores fueron altamente limitados durante el proceso de desarrollo.

12. REFERENCIAS BIBLIOGRÁFICAS

- Achilli, A., & Childress, A. E. (2010). Pressure retarded osmosis: From the vision of Sidney Loeb to the first prototype installation - Review. *Desalination*, 261(3), 205–211. <https://doi.org/10.1016/j.desal.2010.06.017>
- Aleagaloaiza, L. F., Morell Pérez, C., & García Sánchez, Z. (2015). Planificación automatizada del arranque de generadores para la restauración de sistemas eléctricos de potencia. *Ingeniería Energética*, 36(2), 168–179. http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1815-59012015000200006&lng=es&tlng=es
- Ant-Design. (n.d.). *Introduction - Ant Design*. Retrieved June 13, 2021, from <https://ant.design/docs/spec/introduce>
- Arnold, J. (2017). *Dumb Components and Smart Components | by Jason Arnold | Medium*. <https://medium.com/@thejasonfile/dumb-components-and-smart-components-e7b33a698d43>
- Bárzaga Martell, L., Mompie Paneque, R., & Valdés Cuesta, B. (2016). Sistemas SCADA para la automatización de los procesos productivos del CIGB. *Ingeniería Electrónica, Automática y Comunicaciones*, 37(1), 20–37. <https://doi.org/10.1234/rielac.v37i1.314>
- Blandin, G., Rosselló, B., Monsalvo, V. M., Batlle-Vilanova, P., Viñas, J. M., Rogalla, F., & Comas, J. (2019). Volatile fatty acids concentration in real wastewater by forward osmosis. *Journal of Membrane Science*, 575(November 2018), 60–70. <https://doi.org/10.1016/j.memsci.2019.01.006>
- Bogler, A., Rice, D., Perreault, F., & Bar-Zeev, E. (2019). Comparing membrane and spacer biofouling by Gram-negative *Pseudomonas aeruginosa* and Gram-positive *Anoxybacillus* sp. in forward osmosis. *Biofouling*, 35(1), 104–116. <https://doi.org/10.1080/08927014.2019.1570502>
- Capan, T. (2017). *Why the Hell Would You Use Node.js | by Node.js | Node.js Collection | Medium*. <https://medium.com/the-node-js-collection/why-the-hell-would-you-use-node-js-4b053b94ab8e>
- Gamma, E., Helm, R., Johnson, R., & Vissides, J. (1994). Creational Patterns. *AdvancED ActionScript 3.0: Design Patterns*, 127–189. https://doi.org/10.1007/978-1-4302-3615-3_6
- Gelabert Machín, A., & Pérez González, Edilberto. Ochoa Armas, L. . (1995). *Sistema*

- automatizado para la vigilancia epidemiológica de la tuberculosis en la atención primaria en Cuba.* http://scielo.sld.cu/scielo.php?pid=S0864-34661995000100002&script=sci_arttext&lng=en
- Hoek, E. M. V., Kim, A. S., & Elimelech, M. (2002). Influence of crossflow membrane filter geometry and shear rate on colloidal fouling in reverse osmosis and nanofiltration separations. *Environmental Engineering Science*, 19(6), 357–372. <https://doi.org/10.1089/109287502320963364>
- li, C. I. M. (2000). *Cim ii fiuba*. 62.
- Javascript.info. (n.d.). *Destructuring assignment*. Retrieved June 14, 2021, from <https://javascript.info/destructuring-assignment>
- Jucá, S. C. S., Carvalho, P. C. M., & Brito, F. T. (2011). A low cost concept for data acquisition systems applied to decentralized renewable energy plants. *Sensors*, 11(1), 743–756. <https://doi.org/10.3390/s110100743>
- Kádár, P. (2014). *Pros and Cons of the Renewable Energy Application P é ter K á d á r Hydro Energy*. 11(4), 211–224.
- Kim, Y. M., Kim, S. J., Kim, Y. S., Lee, S., Kim, I. S., & Kim, J. H. (2009). Overview of systems engineering approaches for a large-scale seawater desalination plant with a reverse osmosis network. *Desalination*, 238(1–3), 312–332. <https://doi.org/10.1016/j.desal.2008.10.004>
- Klaysom, C., Cath, T. Y., Depuydt, T., & Vankelecom, I. F. J. (2013). Forward and pressure retarded osmosis: Potential solutions for global challenges in energy and water supply. *Chemical Society Reviews*, 42(16), 6959–6989. <https://doi.org/10.1039/c3cs60051c>
- Logan, B. E., & Elimelech, M. (2012). Membrane-based processes for sustainable power generation using water. *Nature*, 488(7411), 313–319. <https://doi.org/10.1038/nature11477>
- Maisonneuve, J., Laflamme, C. B., & Pillay, P. (2016). Experimental investigation of pressure retarded osmosis for renewable energy conversion: Towards increased net power. *Applied Energy*, 164, 425–435. <https://doi.org/10.1016/j.apenergy.2015.12.007>
- Ministerio de Minas y Energía. (2016). Boletín Estadístico de Minas y energía 2012 – 2016. *Ministerio de Minas y Energía*, 200. http://www1.upme.gov.co/simco/Documents/Boletin_Estadistico_2012_2016.pdf
- Mozilla-org. (2020a). *Closures - JavaScript | MDN*.

- <https://developer.mozilla.org/es/docs/Web/JavaScript/Closures>
- Mozilla-org. (2020b). *La desestructuración - JavaScript | MDN*.
https://developer.mozilla.org/es/docs/Web/JavaScript/Reference/Operators/Destructuring_assignment
- OpenJS-Foundation. (n.d.). *About | Node.js*. Retrieved June 10, 2021, from <https://nodejs.org/en/about/>
- Oracle, I. (n.d.). *Core J2EE Patterns - Data Access Object*. Retrieved June 14, 2021, from <https://www.oracle.com/java/technologies/dataaccessobject.html>
- Patel, S. (2019). *Worker Threads examples in Nodejs | by Sandeep Patel | NodejsMadeEasy | Medium*. <https://medium.com/nodejsmadeeasy/workers-threads-in-node-js-part2-ec45ac084963>
- Redacción Actualidad. (2015). *Colombia, Brasil y Perú, entre los 10 países que más agua tienen en el mundo | EL ESPECTADOR*. <https://www.elespectador.com/actualidad/colombia-brasil-y-peru-entre-los-10-paises-que-mas-agua-tienen-en-el-mundo-article-547715/>
- Roman, A. (2010). *TEACHING AUTOMATIZACION INDUSTRIAL*. http://www.avid-romangonzalez.com/Teaching/AUTOMATIZACION_INDUSTRIAL.pdf
- Salamanca, J. M., Álvarez-Silva, O., & Tadeo, F. (2019). Potential and analysis of an osmotic power plant in the Magdalena River using experimental field-data. *Energy*, 180, 548–555. <https://doi.org/10.1016/j.energy.2019.05.048>
- Silva-Díaz, L. J., Hernández-Lopez, Y., Vázquez-Peña, A., Pérez-Acosta, O. G., & Pérez-Torres, D. (2017). Diseño de un sistema de automatización para la planta de alimento ensilado “Héctor Molina.” *Revista Ciencias Técnicas Agropecuarias*, 26(4), 109–120. http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S2071-00542017000400010&nrm=iso
- Suwarno, S. R., Huang, W., Chew, Y. M. J., Tan, S. H. H., Trisno, A. E., & Zhou, Y. (2018). On-line biofilm strength detection in cross-flow membrane filtration systems. *Biofouling*, 34(2), 123–131. <https://doi.org/10.1080/08927014.2017.1409892>
- Thelin, W. R., Sivertsen, E., Holt, T., & Brekke, G. (2013). Natural organic matter fouling in pressure retarded osmosis. *Journal of Membrane Science*, 438, 46–56. <https://doi.org/10.1016/j.memsci.2013.03.020>
- Tiraferri, A., Yip, N. Y., Phillip, W. A., Schiffman, J. D., & Elimelech, M. (2011). Relating performance of thin-film composite forward osmosis membranes to support layer

- formation and structure. *Journal of Membrane Science*, 367(1–2), 340–352.
<https://doi.org/10.1016/j.memsci.2010.11.014>
- Valencia, J. (1981). *Teoría, diseño y control de los procesos de clarificación de agua*.
- Vives, L. (2017). *Reconocimiento e interacción sobre una máquina con procesos Osmóticos*.
- Wikipedia. (n.d.). *Data access object - Wikipedia*. Retrieved June 14, 2021, from https://en.wikipedia.org/wiki/Data_access_object
- Yanoy, M. S., Mesa, M., Pedro, I., Abreu, L. C., Yosleidy, I., Alayon, R., & Coronel, C. C. E. I. (2014). *Sistema automatizado para la organización racional del proceso cosecha-transporte de cereales (SAORCE) Automated system to the rational organization of cereals harvest-transport process (SAORCE)*. 23(3), 89–93.
<http://scielo.sld.cu/pdf/rcta/v23n3/rcta14314.pdf>
- Yip, N. Y., Tiraferri, A., Phillip, W. A., Schiffman, J. D., Hoover, L. A., Kim, Y. C., & Elimelech, M. (2011). Thin-film composite pressure retarded osmosis membranes for sustainable power generation from salinity gradients. *Environmental Science and Technology*, 45(10), 4360–4369. <https://doi.org/10.1021/es104325z>
- Yuan, M. (2017). *Conozca MQTT – IBM Developer*.
<https://developer.ibm.com/es/articles/iot-mqtt-why-good-for-iot/>
- Zilouchian, A., & Jafar, M. (2001). Automation and process control of reverse osmosis plants using soft computing methodologies. *Desalination*, 135(1–3), 51–59.
[https://doi.org/10.1016/S0011-9164\(01\)00138-2](https://doi.org/10.1016/S0011-9164(01)00138-2)