

UNIVERSIDAD SANTO TOMÁS

IMPLEMENTACIÓN DE UN LABORATORIO REMOTO PARA LA REALIZACIÓN DE PRUEBAS DE VACÍO Y ROTOR BLOQUEADO EN MOTORES ELÉCTRICOS

Realizado por

Damian Camilo Rojas Gonzales

Juan Carlos Rincón Contreras

Anteproyecto enviado en cumplimiento del requisito parcial
para optar por el grado de Ingeniería Electrónica



Facultad de Ingeniería Electrónica
División de Ingenierías

Diciembre de 2022

IMPLEMENTACIÓN DE UN LABORATORIO REMOTO PARA LA REALIZACIÓN DE PRUEBAS DE VACÍO Y ROTOR BLOQUEADO EN MOTORES ELÉCTRICOS

Realizado por

**Damian Camilo Rojas Gonzales
Juan Carlos Rincón Contreras**

Anteproyecto enviado en cumplimiento del requisito parcial
para optar por el grado de Ingeniería Electrónica

Dirigido por

Dario Alejandro Segura Torres

Codirigido por

Edwin Francisco Forero Garcia

Facultad de Ingeniería Electrónica
División de Ingenierías

Diciembre de 2022

Índice general

1. Introducción	1
2. Planteamiento del problema	2
3. Justificación	4
4. Estado del arte	6
5. Objetivos	9
5.1. Objetivo General	9
5.2. Objetivos Específicos	9
6. Marco Teórico	10
6.1. Aplicaciones Web	10
6.1.1. HTML	11
6.1.2. CSS	12
6.1.3. JavaScript	13
6.2. Prueba de vacío en motores de inducción	14
6.3. Prueba de rotor bloqueado	15
6.4. Requerimientos para la implementación de un laboratorio remoto	15
7. Desarrollo Metodológico	17
7.1. Funcionamiento físico de la planta	17
7.2. Desarrollo de la plataforma web	17
7.3. Estructuración y desarrollo de la base de datos	19
7.4. Desarrollo de la API	19
7.5. Desarrollo de la aplicación para la Raspberry Pi 3	19
8. Resultados	21
8.1. Acondicionamiento de la planta física	21
8.1.1. Voltaje en motores	21
8.2. Estructura del sistema	23
8.3. Base de Datos	23
8.4. Comunicación	24
8.4.1. Tarjeta de desarrollo - Raspberry Pi 3	24

8.4.2.	Comunicación con analizador ElNet LT	25
8.4.3.	Comunicación variador de frecuencia	27
8.5.	API	29
8.5.1.	Métodos API - Analizador ElNet LT	29
8.5.1.1.	CargarImagen - POST	29
8.5.1.2.	DownloadFile - GET	30
8.5.1.3.	GetAccion - GET	31
8.5.1.4.	AccionKey - POST	32
8.5.2.	Métodos API - Variador de frecuencia	32
8.5.2.1.	Run - Post	33
8.5.2.2.	GetRun - Get	33
8.5.2.3.	UpdateVelocidad - Post	34
8.5.2.4.	GetVelocidad - GET	35
8.5.3.	Métodos API - Fuente externa	36
8.5.3.1.	ManagementSource - POST	36
8.5.3.2.	GetAccionSource - GET	37
8.5.4.	Métodos API - Autenticación y validación de datos	38
8.5.4.1.	GetUsuario - GET	38
8.5.4.2.	GetDatos - GET	39
8.5.4.3.	AddUser - POST	39
8.5.5.	Métodos API - Agendamiento	40
8.6.	Aplicación de consola Raspberry APP	42
8.6.1.	Aplicativo Web - UstaLab	43
8.6.1.1.	Vistas	43
8.6.1.2.	Controladores	46
8.7.	Pruebas de funcionalidad	49
8.7.1.	Inicio de sesión	50
8.7.2.	Creación de Usuarios	51
8.7.3.	Agendamiento de Sesiones	52
8.7.4.	Ejecución de la prueba de vacio	55
8.7.5.	Ejecución de la prueba de rotor bloqueado	59
8.8.	Encuesta a estudiantes	62
9.	Análisis de resultados	66
9.1.	Acondicionamiento de la planta física	66
9.2.	Desarrollo de la plataforma Web	66
9.3.	Pruebas de funcionamiento.	67
9.4.	Impacto en los estudiantes	68
10.	Conclusiones	69
11.	Apéndice	70

Capítulo 1

Introducción

Los laboratorios remotos han sido la respuesta a la necesidad de realizar prácticas durante la pandemia causada por el COVID-19, apoyados por los avances tecnológicos que se han logrado durante las últimas décadas en los campos del control y la automatización.

Con el fin de brindarle una nueva herramienta a los estudiantes de la Universidad Santo Tomás de la Facultad de Ingeniería Electrónica, para el desarrollo de sus clases, se ha implementado un laboratorio remoto para realizar pruebas de vacío y rotor bloqueado en motores eléctricos, en este caso, un motor trifásico 220V/440V.

En esta implementación se han integrado elementos de hardware y de software, obteniendo un sistema con el cual el usuario final es capaz de interactuar en tiempo real con el equipo por medio de una plataforma web, la cual entrega los resultados de la experimentación.

Capítulo 2

Planteamiento del problema

En los últimos años debido a la pandemia del COVID-19 muchos países han sufrido secuelas a nivel económico y social. Uno de los sectores más afectados por los cambios ha sido la educación, en donde cientos de países han tenido que tomar la decisión de cerrar sus centros educativos en todos los niveles desde la primaria hasta los estudios superiores. La tecnología ha sido una gran mediadora con la cual han logrado adaptarse y pasar de un modelo presencial a un modelo virtual [1] [2]. Sin embargo, también ha mostrado las numerosas dificultades que este modelo trae, en especial en países en desarrollo, donde la conectividad a internet no es la mejor y no todos los estudiantes cuentan con ella en su hogar, tampoco con los espacios adecuados para estudiar y a veces ni con los recursos tecnológicos (computadores, tablets, celulares) para llevar a cabo las clases [3].

No solo para los estudiantes ha sido un cambio drástico, los docentes también se han encontrado con varias dificultades relacionadas con el manejo de TIC y la educación virtual, llevándolos a tomar capacitaciones, una reestructuración de sus clases y métodos de evaluación, incluso los ha llevado a posponer o pasar a simulación las prácticas experimentales que normalmente llevaban en los laboratorios [4].

Estos cambios han generado un punto de inflexión en la educación y ha mostrado que la educación necesita un cambio y que aunque existan brechas sociales es posible pensar en una educación virtual[2] la cual llega a tener varios puntos a favor tanto para las universidades; en temas de costo en infraestructura, como para los estudiantes; los cuales reducen costos en su desplazamiento e incluso en su manutención, pero hay un aspecto importante que se debe tener en cuenta y el cual en la educación virtual aún no se tiene al ciento por ciento implementado:

la formación experimental. La cual es de gran importancia en los campos relacionados con la ciencia, la ingeniería y las tecnologías [5].

Normalmente la formación experimental se lleva a cabo en los laboratorios, sin embargo, la pandemia que ha causado el COVID-19 ha cerrado esta posibilidad, pero ha abierto las puertas a una nueva metodología que poco a poco empezaba a tomar fuerza y que la pandemia la ha impulsado aún más: Los laboratorios remotos, que permiten la experimentación real por medio de equipos a través de internet [6].

Esta nueva manera de realizar laboratorios fortalece la educación a distancia, pero a su vez ofrece nuevas herramientas a la educación presencial, ya que es importante mencionar que para mantener un laboratorio los costos son altos, ya que se requiere de personal capacitado para su administración, lugares idóneos para el uso de los equipos, y muchas universidades no logran asumir dichos costos limitando así su capacidad de enseñanza. Sin contar que los estudiantes muchas veces se encuentran con problemas de disponibilidad de equipos o de tiempo para realizar sus prácticas dentro de los laboratorios [5].

Aunque se llevan ya varios años trabajando en el campo de los laboratorios remotos como una herramienta para la educación a distancia, como se mencionó anteriormente, la pandemia ha sido un punto de inflexión que ha evidenciado la necesidad de una mayor implementación de estos laboratorios, la opinión de los estudiantes es una gran demostración de esto, siendo una gran herramienta que puede proporcionar ventajas tales como disponibilidad las 24 horas, el poder realizar el mismo experimento varias veces si se llegan a encontrar con resultados extraños, el acceso para realizar los experimentos y que este tipo de laboratorios promueven las competencias tecnológicas. Sin embargo, es importante mencionar que existen aún dificultades en este campo, la conexión a internet y los equipos computacionales de los estudiantes son un factor que influye en la experiencia al usar este tipo de laboratorios. A pesar de esto, en tiempos de pandemia han sido de gran ayuda en las universidades donde se encuentran implementados estos laboratorios, tanto para estudiantes como para docentes, y se estima que pueden llegar a ser de gran importancia en tiempos de post pandemia [6].

Debido a este nuevo interés por los laboratorios remotos nace una pregunta ¿Que impacto a nivel académico puede llegar a tener la implementación de un laboratorio remoto en la facultad de Ingeniería Electrónica de la Universidad Santo Tomás? Teniendo en cuenta los cambios que se han dado en el último año los cuales proyectan a tener una educación mediada por TICs.

Capítulo 3

Justificación

La crisis causada por el COVID-19 ha dejado a millones de estudiantes fuera de las aulas, en el caso latinoamericano este número llega a los 23.4 millones de estudiantes en educación superior, causando cambios en la metodología con la que se llevaba la educación presencial y aunque en los últimos años con la creciente cobertura de la educación superior a distancia se han fortalecido las herramientas tecnológicas para ofrecer una enseñanza de calidad, en muchos países latinoamericanos aún siguen existiendo falencias en aspectos de conectividad y recursos para conseguir cubrir todos los aspectos en la educación, desde lo teórico hasta lo práctico [7].

En los últimos años se han venido realizando trabajos para el desarrollo de laboratorios remotos y su integración a la educación superior tanto para el modelo a distancia como al modelo presencial, en este último con el fin de fortalecer las herramientas disponibles para brindarle al estudiante la posibilidad de desarrollar trabajos de manera independiente [8]. Aunque el desarrollo de estos laboratorios lleva años ya en el ámbito académico, a causa de la pandemia ha aumentado el interés por parte de los centros educativos, con el fin de abordar todas las competencias necesarias en la enseñanza de sus estudiantes, especialmente en quienes se encuentran cursando carreras relacionadas con la ciencia, la ingeniería y la tecnología [9].

Contar con estas herramientas permite a las instituciones llegar a disminuir costos para la gestión y administración de laboratorios especializados [5] además de ofrecerle al estudiante la posibilidad que desde casa logre realizar sus prácticas experimentales, lo cual le trae beneficios tales como ahorrar costos en su desplazamiento y disponibilidad las 24 horas del laboratorio [6].

Teniendo en cuenta lo anterior, se propuso mediante este proyecto implementar un laboratorio remoto, enfocado en el estudio de motores, brindado al estudiantes nuevas herramientas

didácticas con las cuales pueda fortalecer su aprendizaje y desarrollar competencias en el aspecto práctico, respondiendo a las necesidades actuales y futuras en temas de disponibilidad y capacidad de laboratorios, con el fin de que el estudiante pueda realizar experimentos sin la necesidad de tener que desplazarse hasta la universidad.

Capítulo 4

Estado del arte

Hasta el día de hoy el desarrollo tecnológico que se ha dado a través de la historia es producto de cientos de estudios e investigaciones realizadas por personas que han dejado su nombre para la posteridad. Pero existe un gran precedente en la historia: la invención de la electricidad. Este hito marcó un antes y un después en el desarrollo tecnológico, abriendo la posibilidad de nuevos estudios que sentaron las bases en los campos del control y la automatización, que al día de hoy son de suma importancia en la robótica y los laboratorios remotos. Estos últimos han ganado popularidad en las últimas décadas, debido al interés de las universidades por brindar mejores experiencias prácticas en sus carreras [10].

Pero ¿Qué es un laboratorio remoto? En pocas palabras es una integración de hardware y software, con la que el usuario interactúa por medio de una plataforma virtual para el desarrollo de experimentos en tiempo real. En la implementación de estos laboratorios se involucran tres agentes principales: los equipos, con los cuales la práctica se llevará a cabo; una base de datos, en la que se gestiona el acceso a usuarios y el almacenamiento de datos producto de la realización experimental; y por último la plataforma virtual, ya sea una página web o un aplicativo especializado, en donde el usuario controla los equipos y visualiza los datos en tiempo real a partir de las experimentaciones que haga [11].

El desarrollo de los laboratorios remotos a través de la historia es extenso, pasando desde sistemas de maestro esclavo como el desarrollado por Goertz y Thomson en 1954, hasta conseguir sistemas muchos más complejos en donde realizan integraciones con software tales como Lab-View [10], sin embargo hay grandes proyectos que se han desarrollado y han sido referentes para los laboratorios remotos en los campos de la ciencia y la ingeniería, de los más destacables son: Visir, PEARL, MARVEL y iLab Project [5].

VISIR comenzó su desarrollo entre los años 2006 y 2008, liderado por el departamento de Ingeniería Eléctrica del Instituto de Tecnología Blekinge, con el objetivo de brindarle a los estudiantes la posibilidad de realizar experimentos tanto de manera física, como de manera virtual para aquellos que estaban en cursos a distancia y se ha convertido en la arquitectura que ha mejorado los laboratorios prácticos en las áreas de la electrónica y los circuitos, permitiendo realizar montajes de circuitos con resistencias, condensadores, diodos, amplificadores operacionales o transistores, además de poder realizar las mediciones pertinentes sobre estos circuitos. Con estas características ha sido de gran utilidad para ser utilizado en asignaturas de los primeros niveles de carreras como la Ingeniería Eléctrica y Electrónica. Su infraestructura es muy sólida la cual presenta cuatro partes: una interfaz web, un servidor de equipos, una matriz de medición y la matriz de relés de conmutación. Al llevar una experimentación por medio de este laboratorio remoto, el usuario realiza su circuito desde la interfaz web, cuando este decide dar por terminado su circuito, los datos se envían a dos servidores, uno de ellos se encarga de verificar que el experimento se puede realizar sin afectar los equipos, luego de esto, el segundo servidor recibe la información del primero e identifica los componentes que son necesarios para la realización del experimento, una vez ambos servidores validan la información el circuito se realiza sobre la matriz y los resultados son devueltos al usuario [12].

Por otro lado, PEARL fue un proyecto desarrollado por la Open University [13] entre los años 2000 y 2003. Este consta de un sistema modular con el que es posible llevar a cabo varios experimentos a distancia. El principal objetivo de este proyecto desde su inicio era de ofrecer al estudiante experiencias de aprendizaje en los campos de la ciencia y la ingeniería, presentando las ventajas que los laboratorios remotos logran aportar, tales como la flexibilidad en tiempo y ubicación, con todo esto se consiguió impulsar la implementación de más cursos a distancia ya que los estudiantes podían realizar prácticas que normalmente se llevarían a cabo de manera presencial en un laboratorio convencional [14].

De igual manera el proyecto MARVEL se ha enfocado en impactar en la educación superior, en este caso su campo es la Mecatrónica además de la formación profesional, ofreciendo la posibilidad a los usuarios de tener un acceso a través de internet a las instalaciones físicas de los laboratorios desde otros lugares. Este proyecto es financiado por la Comisión Europea a través del programa Leonardo da Vinci, junto a colaboradores de otros países como Alemania y Portugal, en donde se han enfocado en la enseñanza de asignaturas como el control de sistemas, mantenimiento, monitorización de procesos, automatización de plantas y maquinaria. De manera general MARVEL se conforma por 3 módulos: Un servidor de comunicaciones flash, para llevar un aprendizaje colaborativo con videoconferencia; Una aplicación de horario y reserva,

por medio de la cual el estudiante aparta una hora para trabajar con el laboratorio; y por último un paquete de aprendizaje electrónico, el cual es el encargado de integrar tanto los módulos anteriores como los contenidos necesarios para poder realizar el control remoto y llevar a cabo los experimentos [15].

Por último, uno de los proyectos más grandes y el cual ha sido desarrollado durante un largo tiempo ha sido iLab, el cual comenzó a mediados de 1998 en el MIT. Al igual que en los anteriores casos buscaban alternativas para ayudar a fortalecer el aprendizaje de los estudiantes, debido a las dificultades en la disponibilidad de los equipos que en aquella época tenían. Una primera solución fue la implementación de un programa desarrollado en Java, el cual permitía a los estudiantes por medio de una plataforma web enviar subrutinas a un servidor el cual estaba conectado con uno de los dispositivos que solían tener problemas de acceso debido a su baja disponibilidad. El interés por seguir brindando este tipo de herramienta a los estudiantes llevó a realizar varios desarrollos durante los siguientes años para conseguir mejorar la experiencia de uso y permitir el acceso también a más universidades, manteniendo un enfoque en el campo de la microelectrónica. A mediados del 2008 se tenía un sistema complejo que integraba simulaciones, software específico para los instrumentos y herramientas especializadas desarrolladas en LabView, logrando de esta manera una gran arquitectura compartida [16].

En el contexto latinoamericano, se tienen varios desarrollos de laboratorios remotos, tales como el desarrollado por la Universidad de El Salvador, en el campo de la prueba de máquinas eléctricas [17], o el desarrollado en la Universidad Piloto de Colombia, en donde también abordan el tema de los motores eléctricos, integrando una plataforma web y el entorno de programación gráfico LabView [18].

Con todos los desarrollos que se han llevado a cabo en el campo de los laboratorios remotos, con el fin de fortalecer la enseñanza en las instituciones educativas, se ha conseguido desarrollar diferentes metodologías para llevar a cabo su implementación, en la cuales se logra evidenciar varios aspectos en común, tales como: la identificación de las herramientas y equipos requeridos, el diseño, la interacción y el manejo de los equipos, las pruebas de funcionamiento y por último el entorno en donde se realizará la práctica. De esta manera logrando establecer las bases para el desarrollo y la implementación de los laboratorios remotos en la actualidad [19].

Capítulo 5

Objetivos

5.1. Objetivo General

Implementar un laboratorio remoto para realización de pruebas de vacío y rotor bloqueado en motores eléctricos trifásicos a 220/440V y 60Hz mediante el uso de tecnologías TIC, con el fin de evaluar el impacto en los estudiantes de la facultad de Ingeniería Electrónica de la Universidad Santo Tomás.

5.2. Objetivos Específicos

- Integrar los elementos mecánicos, electrónicos y de software para el funcionamiento de la planta.
- Desarrollar una plataforma web por medio de la cual el estudiante tendrá acceso al laboratorio remoto.
- Realizar pruebas de funcionamiento para validar la operación de la planta integrada con la plataforma web.
- Implementar un laboratorio piloto dentro del espacio académico Conversión Electromagnética para los estudiantes y por medio de encuestas medir el impacto y aceptación del laboratorio remoto.

Capítulo 6

Marco Teórico

6.1. Aplicaciones Web

Una aplicación Web, como se define en TechTarget: *“Es un programa de aplicación que almacena un servidor remoto y se entrega a través de la interfaz del navegador”* Para el desarrollo de estos aplicativos es necesario integrar *scripts* tanto del lado del servidor como del lado del cliente. Los primeros se encargan de almacenar y recuperar información, mientras que los segundos se encargan de la parte visual con la que el cliente interactúa. [20] [21]

Los *scripts* utilizados para el desarrollo de estos aplicativos, se encuentran enmarcados dentro de estándares con los que se busca mejorar la accesibilidad a los sitios web. Estos estándares los desarrollan varias organizaciones entre las cuales están: ISO, ANSI, AENOR, BSI entre otros. [22]

Sin embargo, dentro de estas organizaciones una de las más influyentes es el W3C, la cual desde que se formó ha establecido protocolos y estándares para el lenguaje web, en donde se destacan: HTML, URL, HTTP, CSS, XML, XHTML, SOAP, UDDI y WSDL, los cuales hasta el día de hoy son tecnologías que aún son usadas. [23]

Para las comunicaciones en las aplicaciones web se ha establecido el protocolo HTTP, creado en el CERN con el fin de compartir datos de forma internacional a bajo costo, el cual define las normas para enviar peticiones y la respuesta a estas peticiones de las páginas webs. [23]

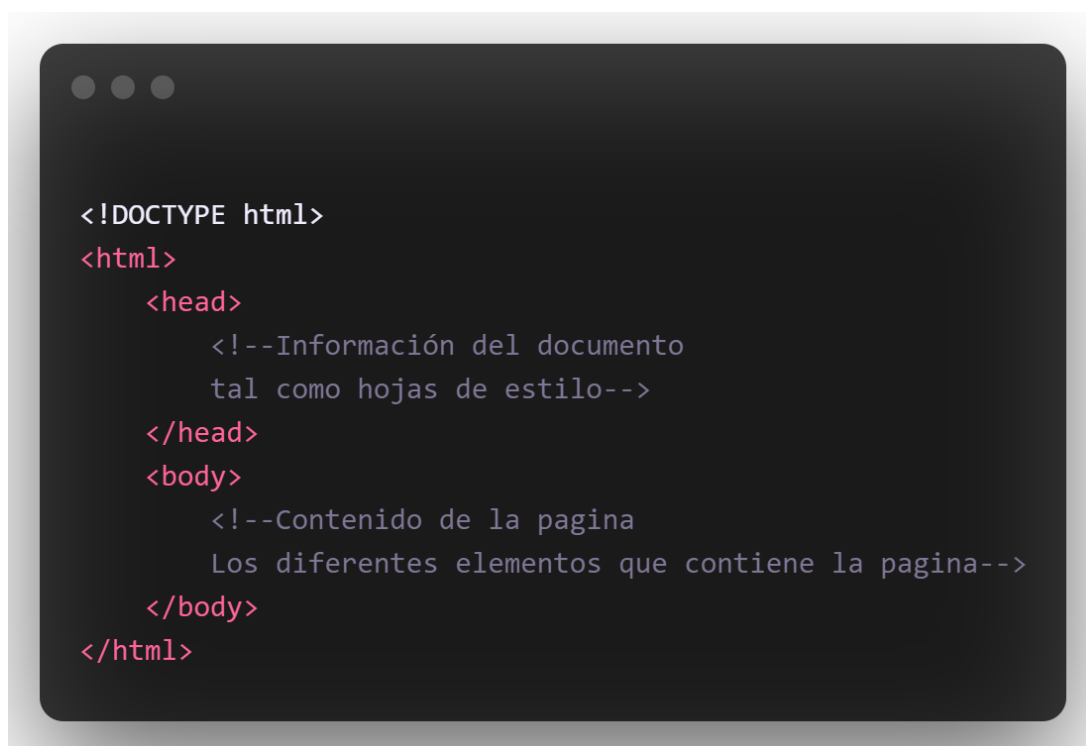
Con el fin de abordar con más profundidad cada una de las temáticas y conceptos mencionados en esta primera parte sobre las aplicaciones web, se desglosan cada una a continuación.

6.1.1. HTML

El HTML comenzó como una alternativa para crear la estructura de las páginas web, con la cual era posible organizar su contenido y compartirlo de manera sencilla. [24]

Debido a la sencillez del HTML nacieron más lenguajes que aunque tuvieron éxito en otros ámbitos del desarrollo, no lograron una integración adecuada con el HTML llevando al nacimiento del JavaScript, que junto al CSS ayudaron al surgimiento de lo que hoy se conoce como el HTML5, que al día de hoy es el encargado de proveer la estructura, estilo y la funcionalidad a las páginas webs.[24]

Aunque el HTML5 se define muchas veces como la combinación de HTML, CSS, JavaScript, cada uno es independiente, y contiene distinciones marcadas. En este caso, tal como lo mencionan en [23] el HTML tiene un esquema definido tal como se ve en la figura 1 en donde se distinguen diferentes partes:



```
<!DOCTYPE html>
<html>
  <head>
    <!--Información del documento
    tal como hojas de estilo-->
  </head>
  <body>
    <!--Contenido de la pagina
    Los diferentes elementos que contiene la pagina-->
  </body>
</html>
```

FIGURA 1: Esquema típico de un documento HTML. Autoría propia

- Doctype: Es la declaración del tipo de documento, en este caso, se define que será de tipo HTML.

- Head: Es el encabezado del documento, en el cual se dispone información o metadatos del documento, como pueden ser hojas de estilos, archivos JavaScript, entre otros, esta cabeza no es visible para el usuario al momento de ver la página web.
- Body: En esta parte se encuentran todos los elementos que contendrá la página, serán visibles y sirven para dar orden y forma a la página.

Como se puede apreciar en la Figura 1 cada una de las secciones que componen un archivo HTML se especifica o diferencia por los símbolos *menor que* (<) y *mayor que* (>) dentro de ellos se especifica un nombre, ya sea para definir el *Head* o el *Body*, a esta pequeña estructura se le define como etiqueta. Existen dos tipos de etiquetas, apertura o inicio y final. Las de apertura o inicio se pueden escribir de la siguiente forma <NombreEtiqueta>por otro lado las de cierre se distinguen por tener una barra diagonal al inicio de esta forma </NombreEtiqueta>. [25]

Algunas de estas etiquetas necesitan más información, un ejemplo de esto puede ser la etiqueta para agregar imágenes al HTML . Esta necesita que se defina la imagen que se quiere mostrar para esto se escribe la etiqueta tal como se muestra en la Figura 2 en donde se define la apertura de la etiqueta *img* seguido de la sigla *src* que hace referencia a *source* o fuente, que determina de dónde proviene la imagen, es decir su ruta, incluyendo el nombre de esta. [23]

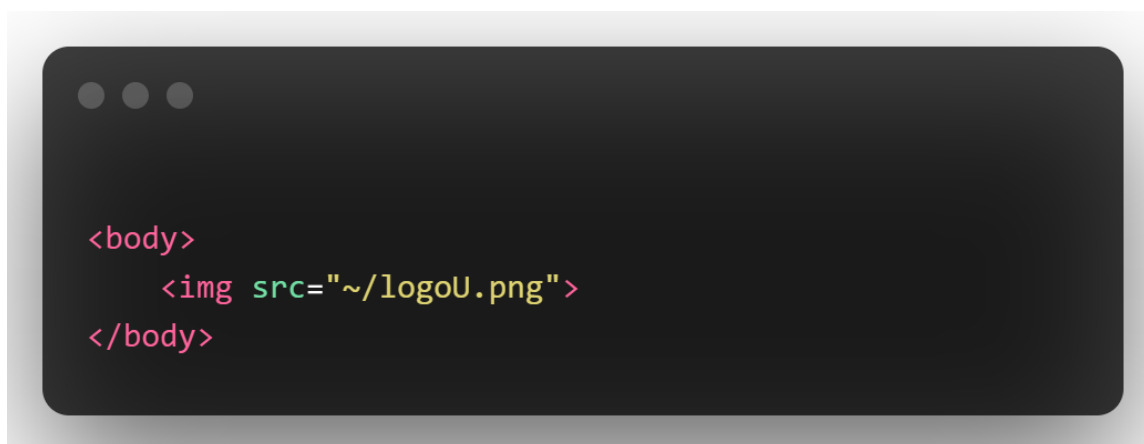


FIGURA 2: Ejemplo de la etiqueta para insertar imágenes en HTML. Autoria propia

6.1.2. CSS

Como se mencionó anteriormente, el CSS (*Cascading Style Sheets*) hace parte del estándar HTML5. Es un lenguaje no es más que un complemento con el que es posible superar los limitantes que tiene el HTML, sin embargo, en la actualidad se habla de su tercera versión, ya que, tal como

el HTML, ha tenido su evolución a lo largo de la historia, en la cual los desarrolladores se han enfocado en llevar al CSS a cumplir con todas las necesidades de diseño que se pueden tener con una página web.[24]

Al ser el CSS un lenguaje enfocado al diseño se puede ver que en este se definen propiedades y sus valores, logrando así una regla de estilo la cual tiene una estructura definida: Selector y bloque de declaraciones. El primero de estos define a qué elementos se les aplica las propiedades que están descritas dentro del segundo, el bloque de declaraciones, en el cual se encuentran las propiedades y los valores que tendrá el elemento en cuestión.[26] Para ilustrar un ejemplo de esto, se muestra la figura 3 en la que a la izquierda se muestra un login básico en la que se ve un mensaje de error, el estilo de este mensaje se ve descrito en la imagen de la derecha en donde se ven cada una de las propiedades que tiene este mensaje.

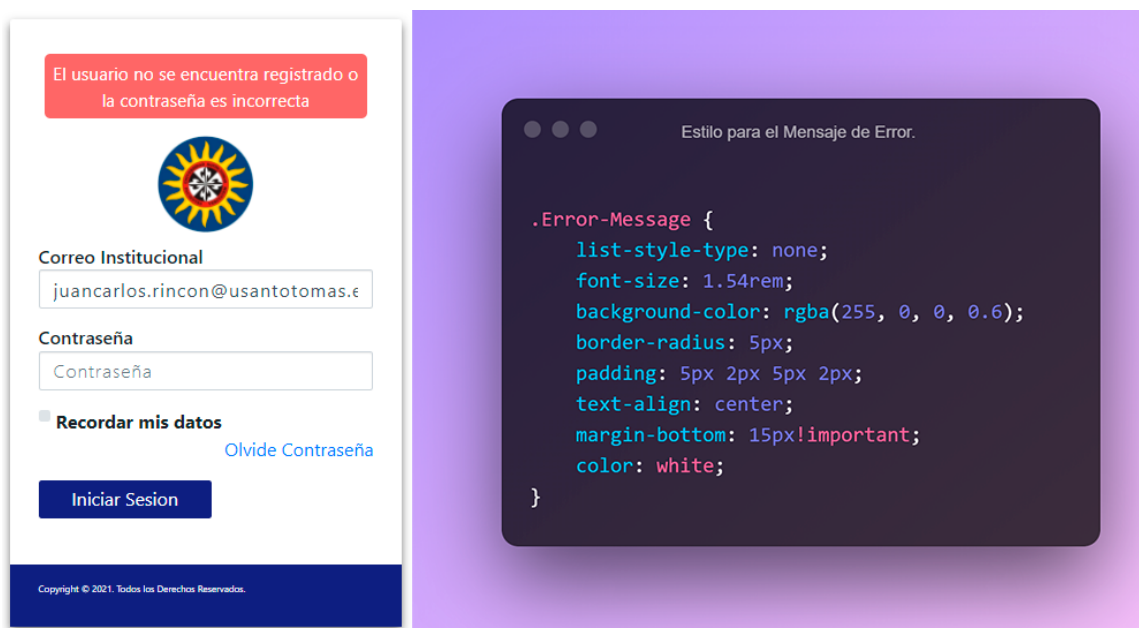


FIGURA 3: Ejemplo del estilo definido para el mensaje de error. Autoría propia

6.1.3. JavaScript

El último de los lenguajes esenciales usados en el diseño y funcionalidad de las páginas web se tiene al Javascript, con el que es posible realizar desde animaciones hasta funcionalidades complejas. Sus inicios parten de aquel lenguaje con el que se comparte nombre Java, el cual fue la primera respuesta para cubrir las necesidades que se tenían con el HTML en sus inicios, sin embargo, era tedioso en aquel entonces, lo que llevó al desarrollo de una nueva alternativa, que

en un principio nació bajo el nombre de LiveScript, el cual, luego de una alianza con el creador de Java quedó bajo el nombre que hoy en día se conoce. [27]

Este lenguaje ha sido esencial desde el nacimiento de la denominada web 2.0 en donde la experiencia de usuario se mejoró y se volvió más dinámica, ayudando a validaciones de formularios, aperturas de ventanas emergentes, animaciones y más funcionalidades que se pueden encontrar en cualquier página web en la actualidad. [27]

6.2. Prueba de vacío en motores de inducción

La implementación de la prueba de vacío en motores de inducción se realiza con el fin de establecer ciertas características físicas del motor que permite conocer su comportamiento ante diferentes condiciones de operación. Durante la prueba, se energiza el motor de inducción de CA a una fuente variable de voltaje sin aplicar carga, dejando correr el motor libremente. Esta condición provoca que el rotor gire aproximadamente a la velocidad de sincronismo, y esto ocasiona que exista un deslizamiento mínimo entre la velocidad de sincronismo y la velocidad mecánica del rotor.[28]

El desarrollo de esta prueba suele asemejarse a la prueba de corto que se realiza con transformadores, ya que el eje del motor está bloqueado de manera que no se puede mover y el devanado del rotor está en cortocircuito. El esquema empleado en esta prueba suele representarse de la siguiente forma.

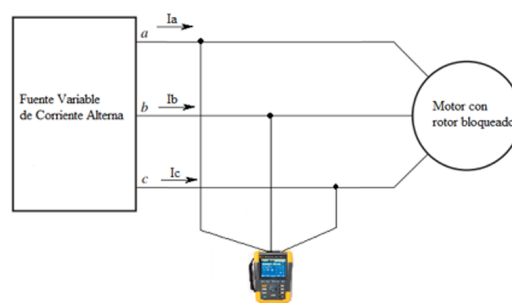


FIGURA 4: Esquema típico rotor bloqueado

Esta prueba se lleva a cabo con el fin de hallar el valor de la resistencia y la reactancia de rotor referidas al estator. El valor de esta resistencia está descrito por la siguiente ecuación.

$$Z_{eq} = \frac{V_{rb1}}{(I_{rb1}\phi)} \quad (6.1)$$

6.3. Prueba de rotor bloqueado

La prueba de rotor bloqueado o rotor enclavado se corresponde a la prueba de cortocircuito de un transformador. Como su nombre lo indica el objetivo de esta prueba es que el rotor no se pueda mover

Para esta prueba es necesario bloquear el rotor del motor para evitar su giro de forma eficiente. Al tener bloqueado el rotor, se procede a inyectar potencia trifásica al motor hasta llegar a la corriente nominal de placa. Se obtiene el voltaje y potencia a corriente nominal con el rotor bloqueado, así como el factor de potencia, los cuales se corroboran con los valores medidos. Para llevar a cabo esta prueba es necesario contar con una fuente que inyecte DC deteniendo de forma segura y controlada el rotor [28]

6.4. Requerimientos para la implementación de un laboratorio remoto

Para la implementación pedagógica de un laboratorio remoto es necesario hacer varias consideraciones en cuanto a lo que es un laboratorio remoto y la forma en que interactúa con la educación a distancia dado que este recurso no cabe estrictamente dentro del modelo de educación a distancia ya que se propone que sea usado como un complemento a las prácticas tradicionales presentando una gran flexibilidad en su implementación académica. Según lo especifica [24] el enlace entre la educación a distancia y el uso de laboratorios remotos arroja que: en este caso se podría hablar de educación constructivista donde se construye conocimiento a través de la aplicación de conceptos teóricos en el laboratorio remoto(LR). Ahora bien, para proporcionar un aprendizaje efectivo mediante el uso de laboratorios remotos es necesario seguir una serie de pautas generales: Aprendizaje en contexto, aprendizaje orientado a objetivos, retos y capacidades coherentes, experiencia significativa, aprendizaje exploratorio y retroalimentación.

- Aprendizaje en contexto: El aprendizaje en un Laboratorio remoto debe estar contextualizado de tal forma para que éste tenga sentido para los estudiantes remotos.
- Aprendizaje orientado a objetivos: Toda experimentación en el laboratorio remoto debe tener claramente definidos los objetivos de aprendizaje, representando de manera correcta los modelos teóricos estudiados previamente por los estudiantes remotos.

- Retos y capacidades coherentes: Los retos o niveles de complejidad en las experimentaciones de los LR deben ser coherentes con las capacidades de los estudiantes remotos; entendiendo como retos, las distintas situaciones que se pueden presentar, como: seguimiento adecuado de instrucciones, interacción y manipulación de equipos de laboratorio, trabajo colaborativo con otros estudiantes e instructor y análisis e interpretación de resultados.
- Experiencia significativa: Todo el entorno y acciones dentro un laboratorio remoto deben tener un efecto significativo sobre las experiencias realizadas por el estudiante remoto.
- Aprendizaje exploratorio: La implementación de un laboratorio remoto donde el estudiante remoto deba tomar decisiones de una cierta variedad de opciones posibles y enfrentar consecuencias de la decisión tomada, genera un modo de aprendizaje exploratorio.
- Retroalimentación: En el desarrollo de un laboratorio remoto por parte de un estudiante con retroalimentación por el trabajo realizado, se obtiene motivación y compromiso generando una mejor experiencia mejorando de esta manera el desempeño del estudiante.

Capítulo 7

Desarrollo Metodológico

7.1. Funcionamiento físico de la planta

Para la implementación de los laboratorios remotos es necesario el acople y comunicación entre los elementos físicos (hardware) y remotos (software) por lo tanto el desarrollo del proyecto contempla el funcionamiento completo de la planta, esta se compone de dos motores trifásicos en conexión delta donde uno actúa como generador y cuyas variables son medidas mediante un circutor.

Para validar el correcto funcionamiento se realizaran pruebas de forma local para comprobar el estado de las conexiones y las variables que se pueden medir por medio de los dispositivos que cuenta la planta, de ser necesario se ajustarán para obtener las mediciones que se requieren para llevar las pruebas a cabo.

Además de esto se comprobará la posibilidad de comunicación de los dispositivos de medida y control de la planta, con el fin de que conocer los requerimientos para realizar una comunicación efectiva con estos y conseguir obtener los datos deseados de la planta y a su vez controlar todo lo que sea posible de manera remota.

7.2. Desarrollo de la plataforma web

En el desarrollo de la plataforma web es necesario dividir el proceso, debido a los diferentes componentes que engloban su estructura. El primer proceso para llevar a cabo es el diseño de la capa de presentación. Las diferentes capas ayudan a la organización del proyecto al momento

de desarrollarlo, en esta primera capa se incluye todo lo relación con la interfaz de usuario. En la arquitectura que se eligió para llevar a cabo el desarrollo de la plataforma web, la interfaz de usuario se desarrolla con la implementación de tres componentes básicos.

- Las vistas: Encargadas de contener todo el código HTML con el que se realiza la estructura de la página web, en donde se establece la ubicación de cada elemento que la conforma.
- Hojas de estilo: Se describe cada una de las clases que tendrá cada uno de los elementos que conforman la estructura de la vista. En cada una de las clases se añaden propiedades que se encargan de establecer el cómo verá el usuario el elemento, es decir, establecen aspectos como el color de la letra, tamaño, posición, ancho, alto, etc. de un elemento HTML.
- JavaScript: En este componente se implementan múltiples funciones para el funcionamiento tanto visual como funcional de la página web, en las que se pueden encontrar las animaciones, validaciones de datos ingresados por el usuario, la comunicación con el controlador, entre otros.

Teniendo en cuenta los diferentes componentes que se necesitan para realizar la capa de presentación, es necesario definir el número de vistas que va a tener el aplicativo web. El desarrollo del proyecto se dividió en tres vistas.

- Vista Login: Encargada de tomar los datos de inicio de sesión del usuario, en donde se implementa una función en JavaScript encargada de realizar el llamado ajax post para enviar los datos ingresados por el usuario al controlador y realizar las consultas con los datos alojados en la base de datos
- Vista Principal: En esta vista se mostrará toda la interfaz con la que el usuario interactúa para controlar y realizar el laboratorio remoto. De la misma manera que la vista login, en esta se implementarán funciones para realizar llamados ajax tanto post como get con el fin de comunicarse con los controladores que a su vez se comunicarán con el servidor al que está conectada la planta física y de esta manera poder enviar y recibir datos.
- Vista Información: En esta vista se presenta una breve descripción del proyecto, información sobre cómo usar la plataforma web e información sobre los autores. Esta vista no incluirá demasiadas funcionalidades ya que solo serán de información para que el usuario la consulte.

7.3. Estructuración y desarrollo de la base de datos

En el desarrollo de la base de datos se uso el software SQL Mangement Studio el cual permite una gestión completa y fácil de bases de datos. Debido a la naturaleza de información que se maneja en la base de datos, solo se han logrado identificar dos tablas con relación entre ellas, las cuales correspondería a las tablas de usuarios y agenda, ya que estarían relacionadas por el id único de los usuarios.

Las demás tablas se desarrollaron con el propósito de almacenar la información necesaria para el control de los dispositivos de la planta y la información que estos nos pueden entregar para mostrar al usuario en la plata forma web.

7.4. Desarrollo de la API

Para el desarrollo de la API al igual que la plataforma Web sera desde el IDE Visual Studio 2022 usando el framework de desarrollo .NET Core. La API es el puente entre la plataforma web y la planta física, ya que por medio de esta se guardarán y recuperaran los datos de la base de datos.

Al ser la API el punto central por donde pasara toda la información también se desarrollara en esta la lógica necesaria para el correcto funcionamiento del aplicativo desde el control de inicio de sesión hasta el agendamiento de sesiones.

Esto con el fin de hacer un poco más simple el aplicativo web y centralizar toda la lógica de negocio en un solo lugar, así al momento de realizar actualización solo es necesario modificar la lógica dentro de la API y de ser necesario habría que aplicar cambios menores dentro de los demás aplicativos que la consuman.

7.5. Desarrollo de la aplicación para la Rasberry PI 3

La aplicación para la Rasberry PI se plantea como una aplicación de consola, la cual se encargará de consumir la API tanto para dar como recibir información nueva continuamente, al igual que se encargará de controlar los dispositivos con los cuales tenga conexión, tales como el analizador Elnet LT (circutor) y el variador de frecuencia.

Al ser un aplicativo que va a correr siempre en segundo plano, no requiere mayor complejidad, pero si es necesario que los métodos dentro de este aplicativo se realicen de manera paralela para que de esta forma en ningún momento choquen procesos que puedan hacer lenta la comunicación y la transferencia de información.

Capítulo 8

Resultados

8.1. Acondicionamiento de la planta física

8.1.1. Voltaje en motores

La planta inicialmente estaba configurada como se muestra en la Figura 5 en donde los motores estaban conectados a dispositivos determinados. el motor de la izquierda de la imagen, estaba conectado al analizador ElNet LT, el cual se encarga de realizar las mediciones, mientras que el otro motor estaba conectado al variador de frecuencia.



FIGURA 5: Planta en su configuración inicial. Autoría propia

Esta configuración causaba que al momento de poner en marcha la planta el analizador EINet LT no registraré una medida de voltaje, ni de corriente y por ende de potencia. Por lo que fue necesario modificar las conexiones de la plata tal como se muestra en la figura 6.

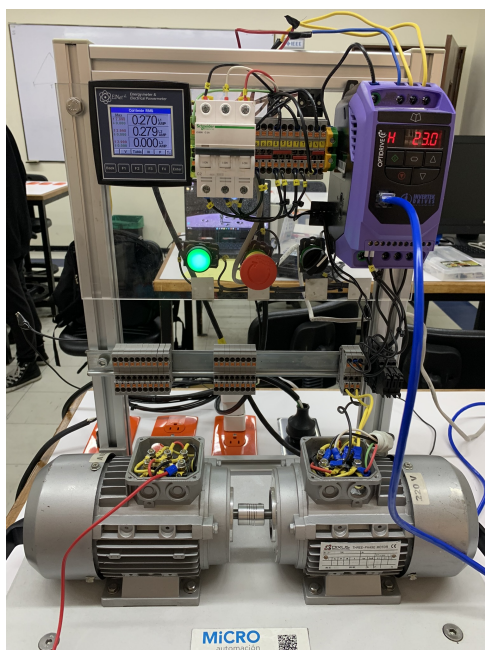


FIGURA 6: Planta adaptada. Autoría propia

De esta manera se mide correctamente el voltaje, corriente y potencia y además permite dejar disponible el segundo motor para que sea el que bloquee el rotor al momento de realizar la prueba de rotor bloqueado cuando se le induzca una corriente directa a sus devanados.

8.2. Estructura del sistema

Con el objetivo de establecer un punto de partida para el desarrollo de los diferentes aplicativos se estableció la estructura base que tendría el proyecto. Para lo cual se identificaron cada uno de los dispositivos y agentes involucrados para llevar a cabo el laboratorio, dando así el diagrama de la figura 7,

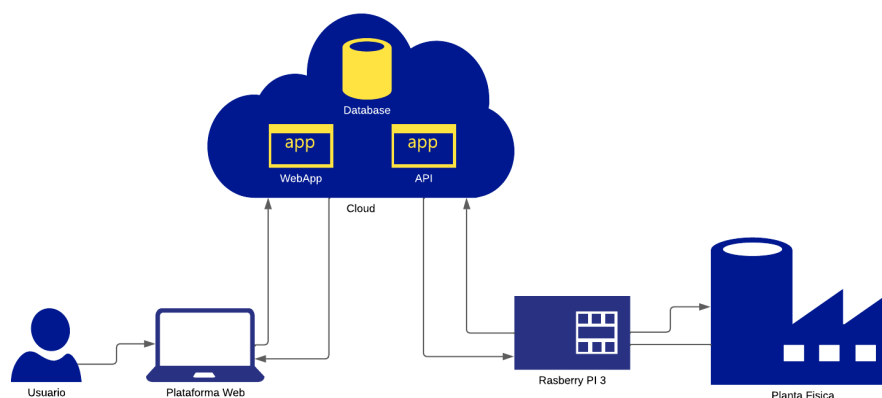


FIGURA 7: Estructura del aplicativo web. Autoría propia

Con el diagrama es posible identificar cada una de las aplicaciones que se deben desarrollar: API, Aplicación Web y Aplicación para Raspberry PI 3. Además de la implementación una base de datos en la cual se almacenaron los datos necesarios para la gestión de información.

8.3. Base de Datos

Se implementó una base de datos no relacional con ocho tablas para poder gestionar tanto la información útil para el usuario al momento de realizar el laboratorio, como la necesaria para el control remoto de cada uno de los dispositivos. La estructura de la base de datos se muestra a continuación.

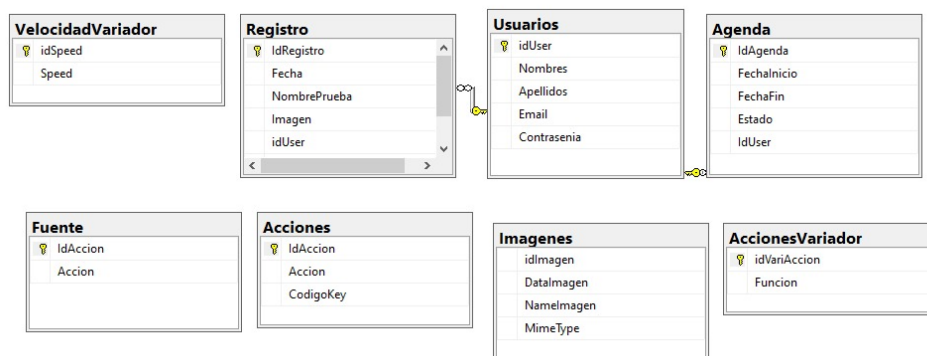


FIGURA 8: Diagrama de la Base de datos. Autoría propia

Como se ve en la figura 8 las únicas tablas que cuentan con una relación son las tabla de usuarios y agenda, las demás tablas aunque es posible establecer relación con la tabla usuarios, no se hizo ya que se consideró innecesario debido a que la información de las demás tablas siempre se actualizará y no dependerá si un usuario está usando o no el dispositivo.

8.4. Comunicación

Para que la planta se pueda controlar de forma remota es necesario comunicar dicha planta a internet. Para lo cual se identificaron los elementos necesarios para lograr controlarla y realizar las pruebas planteadas.

Dichos elementos corresponden al analizador Elnet LT el cual es el encargado de realizar las mediciones de voltaje, corriente y potencia al motor, y el segundo elemento es el variador de frecuencia, con el cual se controla la velocidad de giro del generador.

Ambos elementos descritos cuentan con puertos de comunicación que permiten ser conectados a un computador o tarjeta de desarrollo sin mayor inconveniente, con lo que se logra leer los datos de dichos dispositivos e incluso controlarlos sin interactuar directamente con ellos.

8.4.1. Tarjeta de desarrollo - Raspberry Pi 3

Para la tarjeta de desarrollo se seleccionó una Raspberry Pi 3, debido a que permite tener dentro de ella un sistema operativo, es compatible con el desarrollo de aplicaciones .Net y permite ejecutarlas allí sin ningún inconveniente. Adicional a esto, cuenta con cuatro puertos USB y un puerto Ethernet que permiten conectar otros dispositivos a la tarjeta.

8.4.2. Comunicación con analizador ElNet LT

El analizador ElNet LT permite tres métodos de comunicación, los cuales son: Modbus, Ethernet y con UniArt Software. De estos tres métodos, el más práctico es la comunicación por Ethernet.

Como lo especifica el fabricante, para la comunicación por Ethernet es necesario configurar la dirección IP del dispositivo, la entrada, la máscara de red y el puerto Http como se muestra en la figura 9.

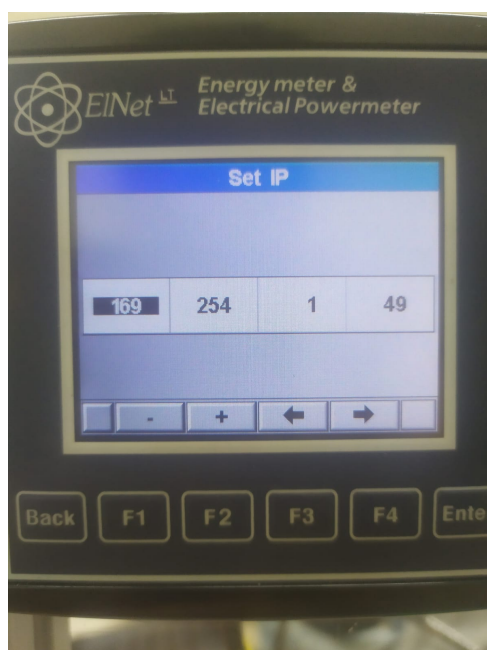


FIGURA 9: Configuración de ethernet en el circutor. Autoría propia

Al tener estos tres parámetros configurados, es posible acceder a la dirección IP junto con el puerto, por medio del navegador, logrando de esta manera visualizar la página que se muestra en la figura 10.

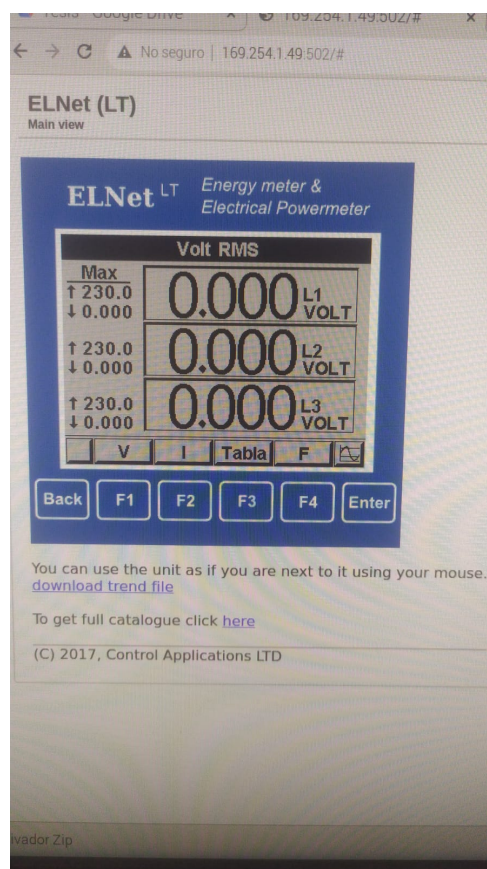


FIGURA 10: Imagen del circutor desde el navegador. Autoría propia

Desde el navegador se puede no solo visualizar lo está actualmente mostrando el analizador, si no que también es posible interactuar por medio de los botones que se encuentran en la parte baja de la imagen, por lo que se obtiene un control completo del dispositivo por medio de dicha página.

Al analizar el funcionamiento de esta página, se logró identificar las diferentes peticiones que se hacían hacia el dispositivo para poder interactuar en tiempo real, y las respuestas que este mismo entregaba para poder visualizar lo que este está mostrando en pantalla.

Para la visualización de la imagen, el dispositivo actualizaba cada segundo la imagen mostrada, haciendo una solicitud GET a la URL `http://169.254.1.49:502/tft.bmp?` seguido una cadena de números que corresponde a la fecha en la que se hizo esta solicitud. Al ingresar dicha dirección al navegador, se obtiene tan solo la imagen actual del analizador, que se actualiza cada segundo, tal como se muestra en la figura 11.

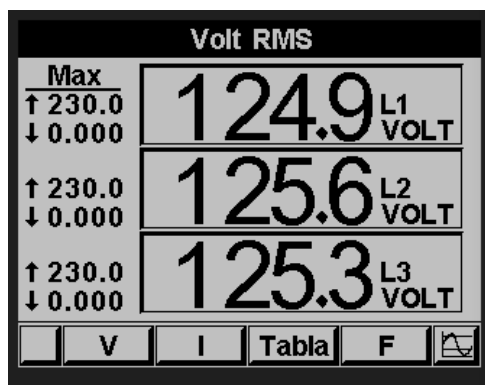


FIGURA 11: Imagen del circutor al momento de ingresar solo la URL de la IP.
Autoría propia

Para la interacción con los botones, la página realiza una solicitud GET a la URL <http://169.254.1.49:502/key.htm?F=3> en donde 3 es la tecla que se oprime, por lo que de esta manera el dispositivo reconoce que tecla se ha oprimido y qué acción debe realizar. Al poner esta URL en un navegador, el analizador ejecuta la acción correspondiente.

Al tener estas dos solicitudes en cuenta es posible plantear las diferentes solicitudes que debe realizar la aplicación de la Raspberry PI para acceder a estas URL tanto para extraer la información como para hacer uso de los botones del circutor.

8.4.3. Comunicación variador de frecuencia

El variador de frecuencia inverttek optidrive e2 cuenta con un puerto RJ45 por el cual se realiza la comunicación usando el protocolo ModBus. En primera instancia se realizaron pruebas desde el PC usando el software Modbus Poll en donde se pudo realizar la comunicación de una manera sencilla, con el fin de establecer los diferentes registros a los cuales era necesario hacer escritura para controlar el variador para ejecutar las funciones necesarias para el desarrollo de las pruebas, para esto también se tuvo apoyo de los manuales del dispositivo. [29]

Partiendo de los resultados de las pruebas, se planteó el esquema mostrado en la figura 12 en el cual se establece el flujo del programa a desarrollar para la Raspberry PI.

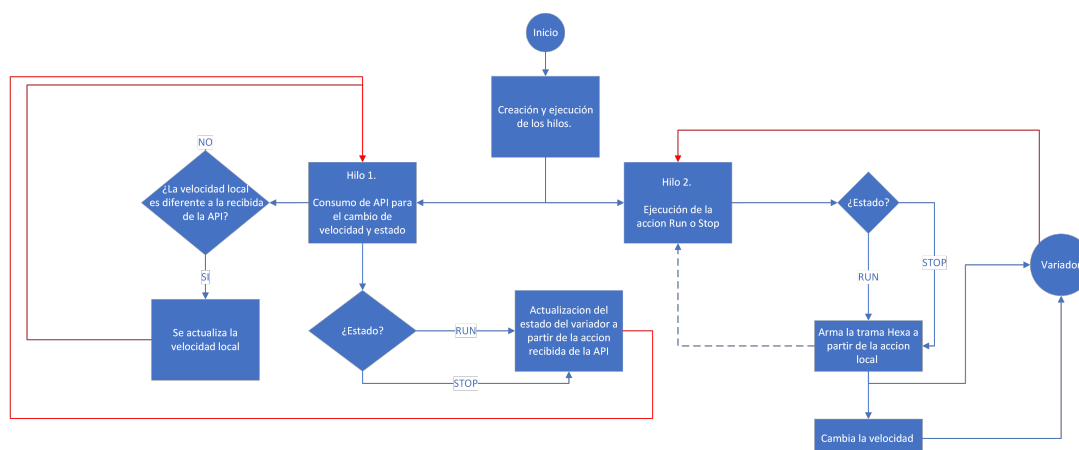


FIGURA 12: Flujo del programa para la comunicación y control de variador de frecuencia. Autoría propia

Como se observa en la figura 12 se plantearon dos hilos para realizar los procesos: Consumo de Api y Comunicación de variador. Ambos hilos se ejecutan en un ciclo esto debido a que es necesario leer continuamente la API y comunicarse con el variador para que este no se pare en ningún momento.

El primer hilo se encarga de consumir el API que trae la información de la velocidad y la acción, esta última hace referencia a si el variador debe estar en marcha o paro. Con las variables actualizadas el segundo hilo ejecuta en primera parte la acción que debe realizar el variador, ya sea ponerse en marcha o parar, para esto se arma la trama hexadecimal y se envía al variador para luego cambiar la velocidad, armando una nueva trama con la nueva velocidad y finalmente se envía.

Es necesario hacer el cambio de velocidad en el mismo hilo y seguido del cambio de acción debido al manejo de los puertos COM ya que desde dos hilos diferentes no se puede acceder al mismo puerto al mismo tiempo sin que se generen conflictos en la ejecución del programa, razón por la cual se optó por el flujo de programa descrito anteriormente.

A partir del flujo de programa realizado, se desarrolló el programa para la Raspberry PI, siendo éste un aplicativo de consola en .NET CORE. Se validó su funcionamiento desde el PC, para luego probarlo directamente en la tarjeta de desarrollo.

8.5. API

Para el desarrollo de la API se establecieron las funcionalidades que se necesitaban tanto para la comunicación con la tarjeta de desarrollo como con el Aplicativo Web. Dichas funcionalidades se agruparon en los diferentes dispositivos de comunicación dentro de la planta: Variador y Analizador (CIRCUTOR). Cada uno de estos dispositivos realizan acciones diferentes y consumen información de diferentes fuentes de la base de datos, por lo que esta distinción era necesaria.

Como se mencionó en la sección 7.5 las tablas creadas en la base de datos almacenan información que se necesita entregar a los equipos, del mismo modo que es necesario almacenar la información entregada por estos, por lo que los métodos de la API fueron desarrollados para estas funcionalidades.

8.5.1. Métodos API - Analizador ElNet LT

En el caso del Analizador o también denominado circutor se crearon seis métodos para acceder a la información o guardarla en la base de datos dependiendo del caso, los cuales son:

- CargarImagen (Método tipo Post).
- DescargarImagen (Método tipo Get)
- GetAccion (Método tipo Get)
- AccionKey (Método tipo Post)

8.5.1.1. CargarImagen - POST

Como se mencionó en la sección 8.2.2 se logró obtener la imagen en tiempo real de lo que muestra este dispositivo en pantalla, para el almacenamiento de esta imagen se desarrolló el flujo mostrado en la figura 13 para el método encargado de guardar esta imagen en la base de datos.

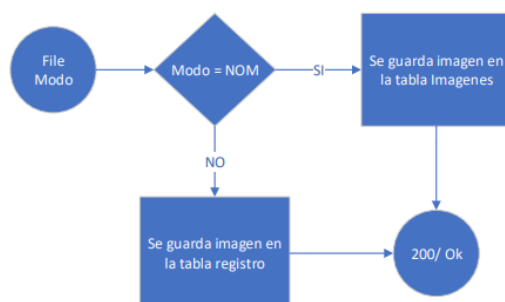


FIGURA 13: Flujo de la API para almacenar la imagen del circutor. Autoría propia

Al saber que la planta será usada por una persona a la vez, no se vio necesario guardar múltiples registros en la tabla, tan solo la última imagen tomada del analizador, de esta manera se facilita la obtención de esta imagen y se asegura que la imagen guardada en la base de datos es la más actual.

Al momento de realizar la prueba de roto bloqueado y teniendo la limitación de tiempo no superior a 8 segundos por temas de seguridad y cuidado de los diferentes dispositivos de la planta, se implementó la tabla registro en donde al momento de ejecutar la prueba de rotor bloqueado la imagen se guarda en esta tabla y de esta manera dar la posibilidad de ser consultada posteriormente.

8.5.1.2. DownloadFile - GET

Con el fin de mostrar la imagen en la plataforma web se desarrolló el flujo de la figura 14 en donde se establece la lectura de la base de datos y la conversión de los datos de la imagen a formato base64 con la cual es posible recuperar el archivo para posteriormente mostrarlo en el aplicativo web.

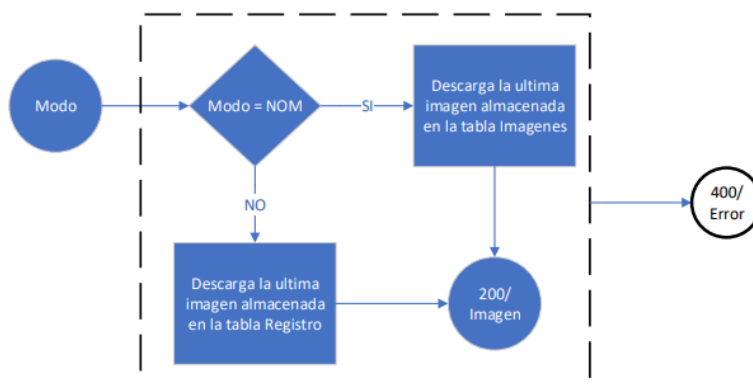


FIGURA 14: Flujo de la API para obtener la imagen almacenada en la base de datos. Autoría propia

Además, se hace la distinción para recuperar la imagen tanto la almacenada en tiempo real como la que se guarda al momento de realizar la prueba rotor bloqueado, de esta manera se puede recuperar la medición que se realizó durante esta prueba.

8.5.1.3. GetAccion - GET

Este método es consumido por la aplicación de la Raspberry Pi 3, en el cual se recupera la acción almacenada en la base de datos para poder enviársela al analizador e interactuar con los diferentes botones que este dispositivo cuenta.

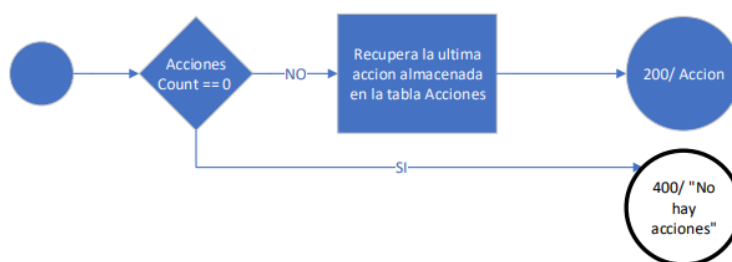


FIGURA 15: Flujo de la API para obtener la acción actual del circutor. Autoría propia

De esa manera se consigue utilizar de manera completa las diferentes opciones que nos ofrece el analizador y visualizar las variables al completo que este dispositivo mide.

8.5.1.4. AccionKey - POST

En la figura 16 se muestra el flujo que desarrolla este método, en donde se realiza una homologación de los diferentes códigos de las teclas que recibe desde el aplicativo web con el fin de almacenarlo en la base de datos de tal manera que al momento de recuperarlos por la Raspberry PI y enviar este dato al analizador, lo pueda comprender y no exista fallas en la comunicación.

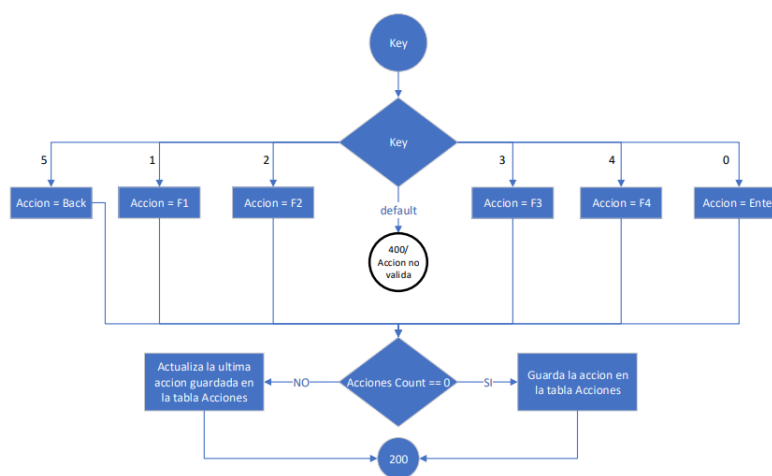


FIGURA 16: Flujo de la API para guardar la acción a ejecutar en el circutor. Autoría propia

Al igual que en el caso del almacenamiento de la imagen tomada del analizador, se guarda tan solo la última acción enviada desde el aplicativo web, de esta forma se facilita la obtención de este dato y se disminuye el consumo de capacidad de la base de datos.

8.5.2. Métodos API - Variador de frecuencia

Para controlar la frecuencia del variador se desarrollaron cuatro métodos los cuales permitirían ponerlo en marcha o detenerlo, conocer si está en marcha o detenido, cambiar la velocidad y recuperar el valor actual de la velocidad, dando como resultado los siguientes métodos:

- Run (Método tipo Post).
- GetRun (Método tipo Get)

- UpdateVelocidad (Método tipo Post)
- GetVelocidad (Método tipo Get)

8.5.2.1. Run - Post

El método *Run* permite almacenar la acción a realizar con el variador en la tabla AccionesVariador. Como lo muestra el flujo de la figura 17 se hace una homologación para guardar en la base de datos un *Run* o *Stop* dependiendo del caso, se valida si ya existe una función registrada en la tabla, en el caso de que no exista se guarda directamente en la tabla, si ya existe alguna otra función se hace la actualización.

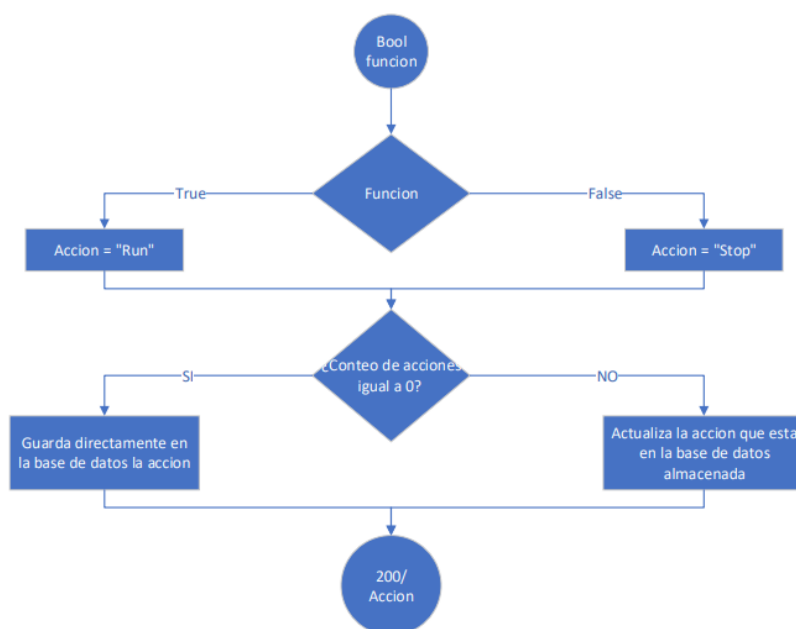


FIGURA 17: Flujo de la API para guardar la acción a ejecutar en el variador. Autoría propia

8.5.2.2. GetRun - Get

Para poder comunicarle al variador que función debe ejecutar se recupera la acción actual que debe ejecutar el variador, para esto se implementó el flujo mostrado en la figura 18.

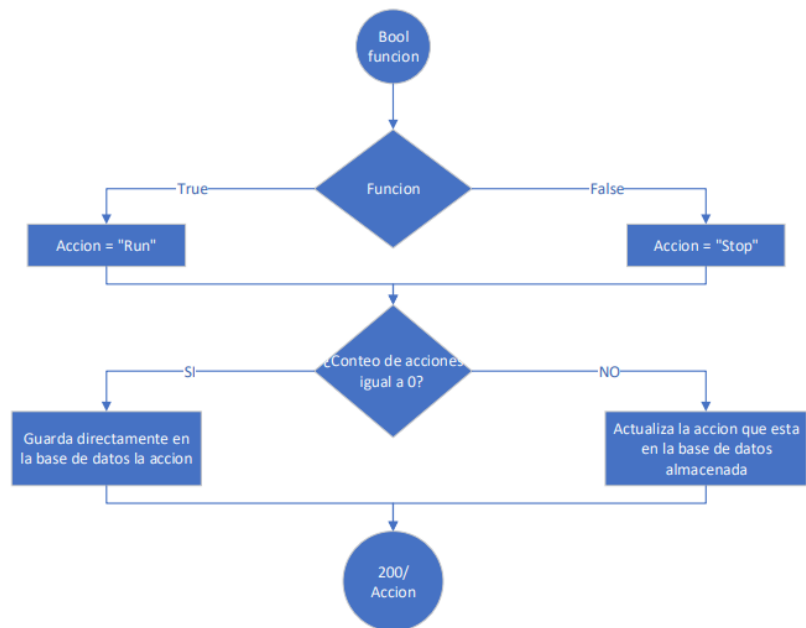


FIGURA 18: Flujo de la API para recuperar la acción a ejecutar en el variador.
Autoría propia

8.5.2.3. UpdateVelocidad - Post

Para modificar la velocidad del variador se implementó un flujo muy similar al flujo del método *Run* en donde se recibe la velocidad que debe tener el variador, se verifica si existe o no ya una velocidad almacenada en la tabla y dependiendo del caso se guarda directamente o se realiza un *update* (actualización del dato) al registro ya existente.

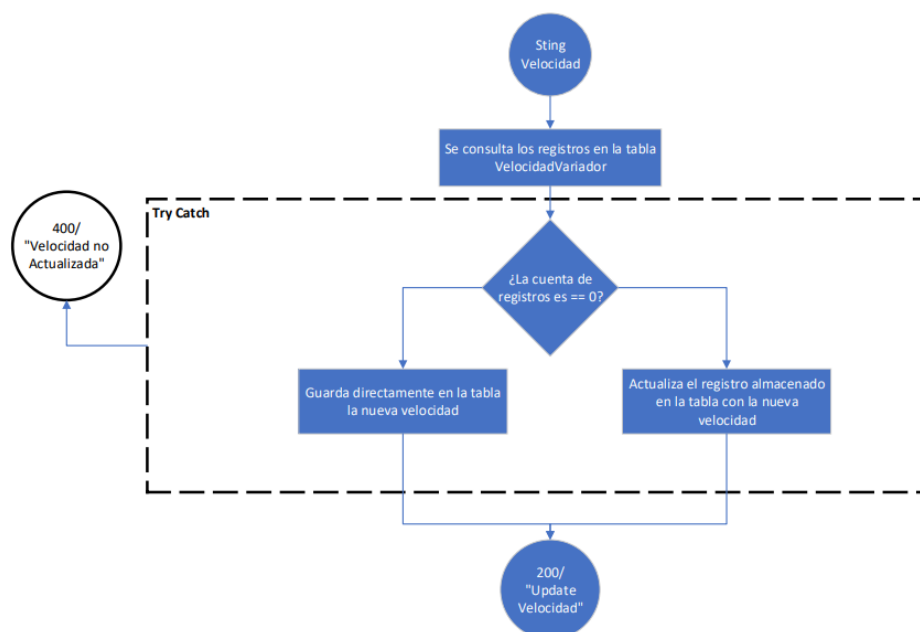


FIGURA 19: Flujo de la API para actualiza la velocidad en el variador. Autoría propia

De esta manera se asegura que solo existe una velocidad en el sistema y facilita el manejo y control de esta variable para evitar inconsistencias y errores.

8.5.2.4. GetVelocidad - GET

El flujo desarrollado para este método se muestra en la figura 20 en donde se hace una consulta a la tabla VelocidadVariador para recuperar la velocidad actual que debe tener el variador.



FIGURA 20: Flujo de la API para recuperar la velocidad del variador. Autoría propia

8.5.3. Métodos API - Fuente externa

Como se menciona en la sección 6.3 en donde se habla de la teoría de la prueba de rotor bloqueado, para esta prueba es necesario inyectar corriente DC para poder bloquear el rotor, por lo que era necesario implementar un sistema en el que sea posible realizar esta conexión de manera automática.

Para esto se hizo uso de un relé el cual se controla desde los puertos GPIO de la Raspberry PI, de tal manera que cuando se da un voltaje de salida por medio de este puerto, el relé se activa y hace la conexión entre la fuente DC y los devanados del rotor.

Debido a esta automatización implementaron dos métodos sencillos para controlar esta conexión:

- ManagementSource (Método tipo Post).
- GetAccionesSource (Método tipo Get)

8.5.3.1. ManagementSource - POST

En la figura 21 se muestra el flujo que se implementó para este método, al igual que los mostrados anteriormente para el control del variador, es un flujo sencillo en donde se hace una homologación para posteriormente validar si se debe actualizar o guardar directamente en la tabla si se desea activar o desactivar la conexión de la fuente.

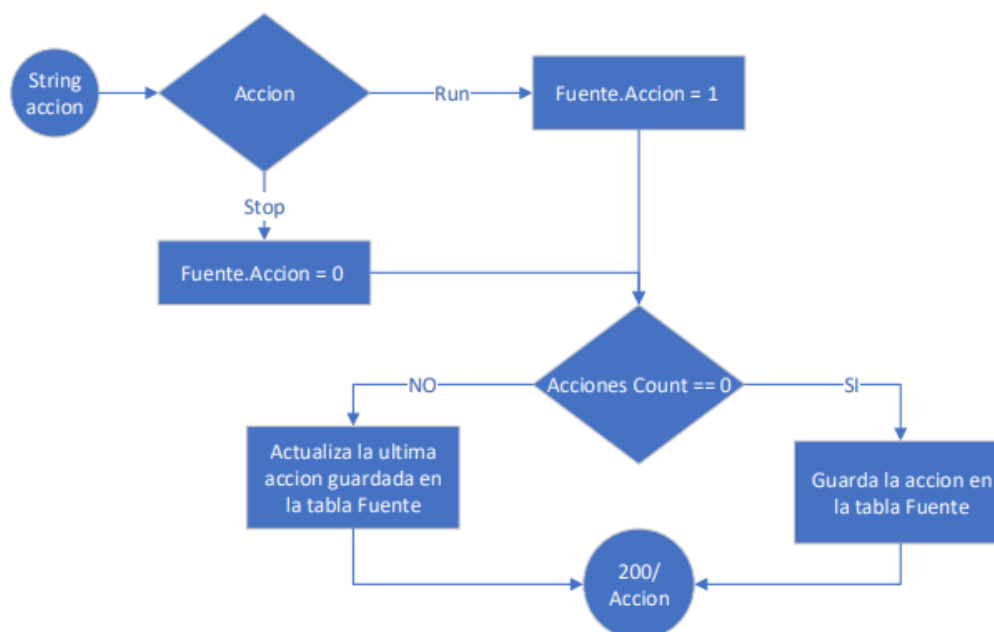


FIGURA 21: Flujo de la API para actualiza la acción de la fuente externa. Autoría propia

8.5.3.2. GetAccionSource - GET

Para obtener la acción actual de la fuente se desarrolla el flujo de la figura 22 de esta manera al momento de consumir este método desde la Raspberry PI es posible realizar la conexión o desconexión de la fuente, dependiendo de la acción almacenada en la base de datos.

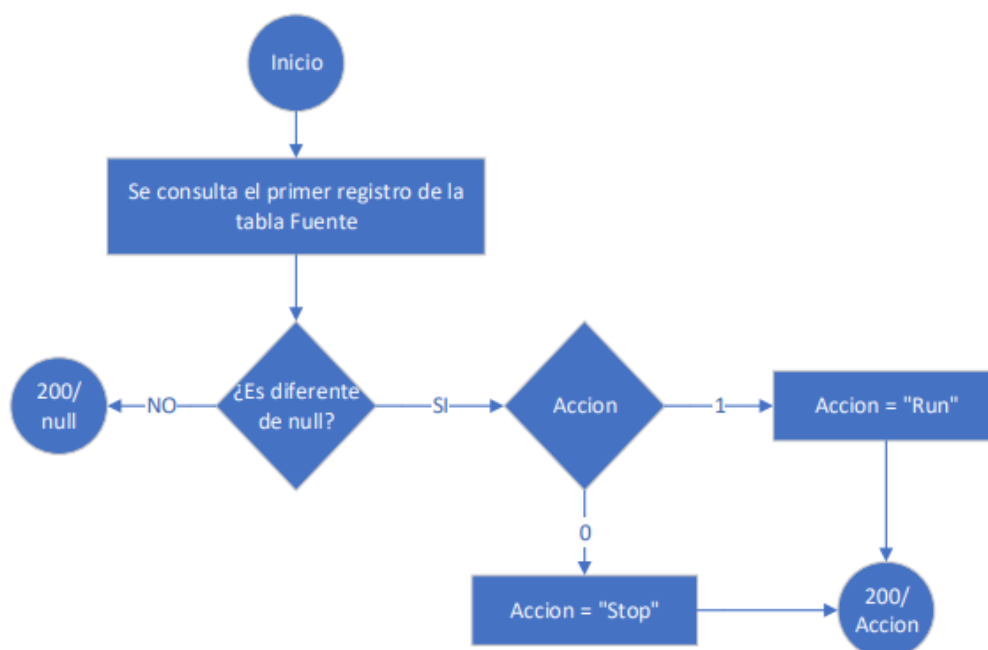


FIGURA 22: Flujo de la API para recuperar la acción de la fuente externa. Autoría propia

8.5.4. Métodos API - Autenticación y validación de datos

Para las funcionalidades de login y registro de usuarios del aplicativo web se desarrollaron tres métodos dentro de la API para realizar estos procesos y almacenar o recuperar la información de la base de datos.

- GetUsuario (Método tipo Get).
- GetDatos (Método tipo Get).
- AddUser (Método tipo Post).

8.5.4.1. GetUsuario - GET

Este método de desarrollo para realizar la función de login y verificar que el usuario que trata de ingresar al aplicativo esta registrado en la base de datos y su contraseña es la correcta. El flujo implementado de muestra en la figura 23

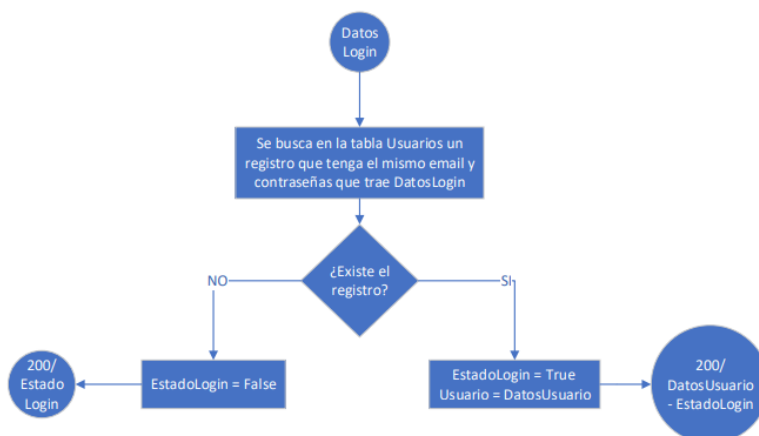


FIGURA 23: Flujo de la API para recuperar la información del usuario que se logea en el aplicativo. Autoría propia

8.5.4.2. GetDatos - GET

Aunque muy similar al anterior método, este tal solo recibe el correo electrónico y se implemento para validar la existencia del usuario dentro de la base de datos al momento de realizar el agendamiento.

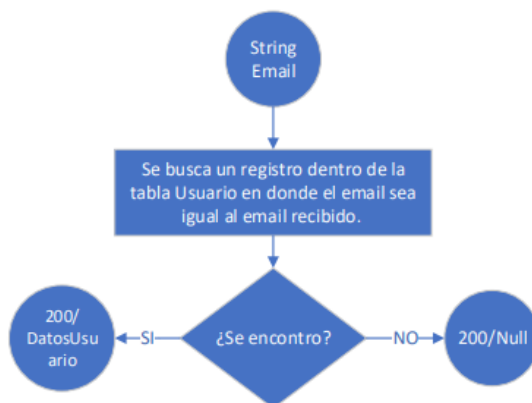


FIGURA 24: Flujo de la API para recuperar la información del usuario por su email. Autoría propia

8.5.4.3. AddUser - POST

Para el registro de usuario se implemento el flujo mostrado en la figura 25 en donde se muestra los datos básicos que se reciben del usuario y se valida si este ya esta registrado, en el caso de

que no exista se continua con su registro y se guarda en la tabla Usuarios, si por el contrario ya existe en la base de datos, la API responde con un error informando que ya existe este usuario.

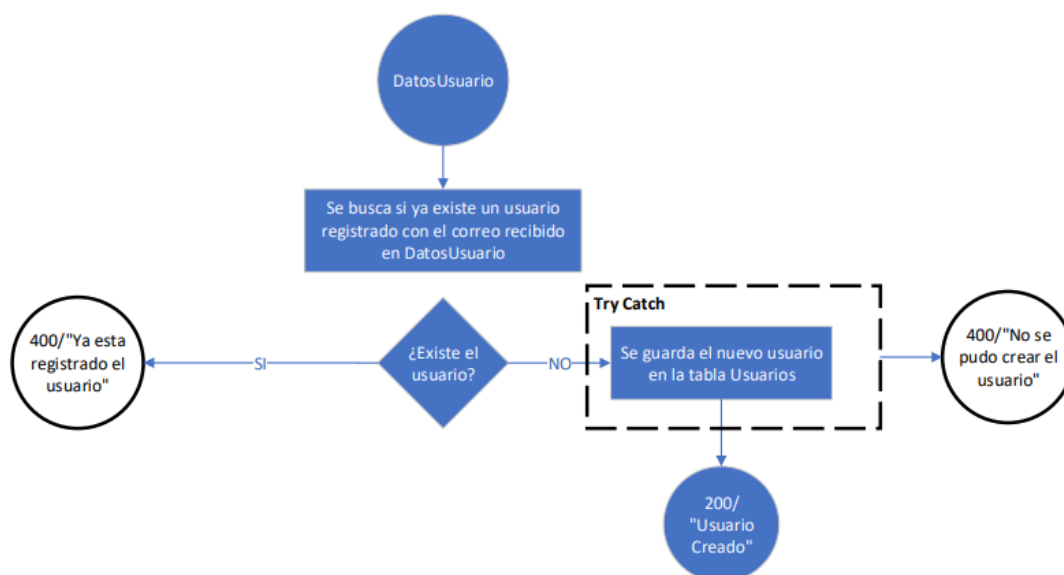


FIGURA 25: Flujo de la API para guardar un nuevo usuario en la base de datos.
Autoría propia

8.5.5. Métodos API - Agendamiento

Dada la necesidad de mantener un orden y un control sobre los usuarios que entrar al aplicativo se implementó un sistema de agendamiento en donde es necesario apartar un horario específico para poder ingresar y usar el laboratorio remoto. Para esto se desarrollaron los siguientes métodos.

- CreateAgenda (Método tipo Post).
- GetAgenda (Método tipo Get).

En este caso no se puede dar un flujo por separado de ambas funciones ya que entre estas se combinan para el correcto funcionamiento del proceso, tal como se muestra en la figura 26.

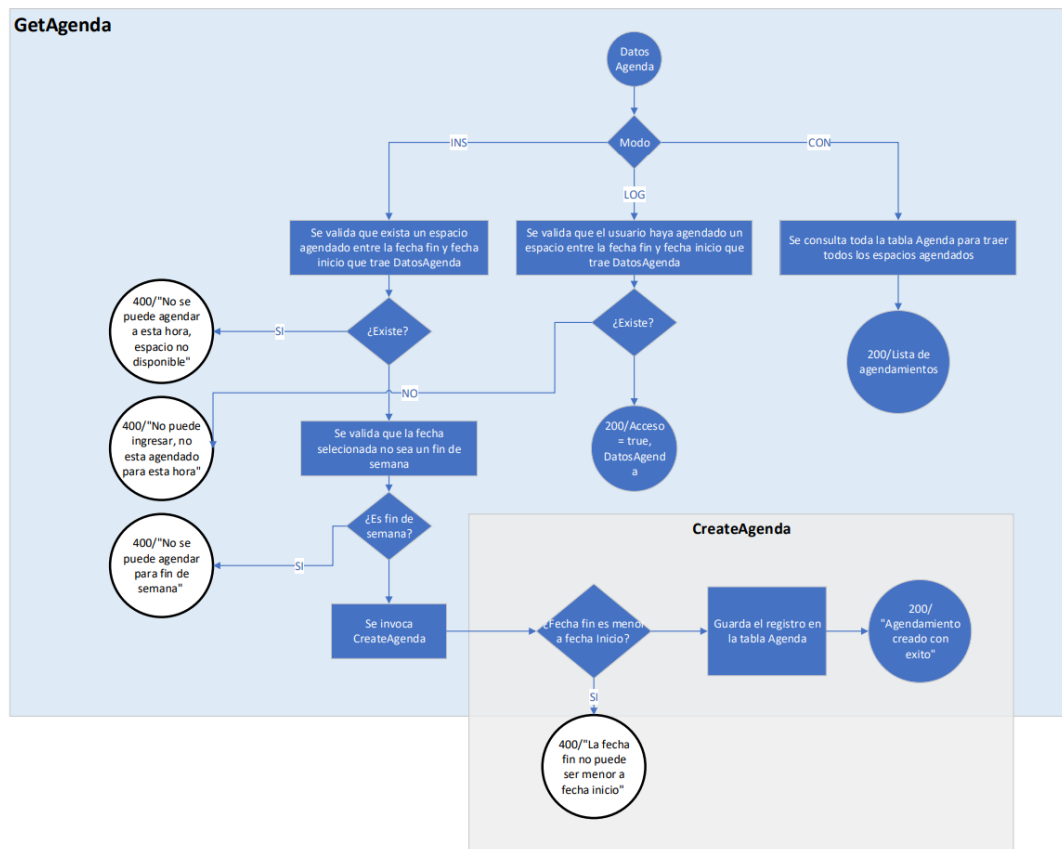


FIGURA 26: Flujo de la API para la creación y consulta de agendaíentos. Autoría propia

El método *GetAgenda* es el primer método consultado y se puede usar para tres casos diferentes, al momento de ingresar en el aplicativo, para ingresar un nuevo agendamiento o para consultar todos los agendamientos existentes.

En cada uno de estos casos se realizan validaciones para comprobar la existencia de un agendamiento en la base de datos, para evitar crear espacios que se crucen, o en el caso del ingreso al aplicativo solo permitir ingresar al usuario que tiene agendado un espacio en ese espacio de tiempo.

Luego de estas validaciones se usa el método *CreateAgenda* en el cual solo se hace una validación de fechas para luego guardar directamente en la tabla el espacio agendado por el usuario.

8.6. Aplicación de consola Raspberry APP

Para el desarrollo del aplicativo que se ejecuta en la Raspberry App se tuvieron en cuenta los siguientes factores que eran críticos para conseguir una comunicación adecuada y un envío de información rápida desde la Raspberry PI 3 hacia la API.

- Ejecución continua: El aplicativo siempre debe estar ejecutándose para el funcionamiento de la comunicación entre dispositivos y API, por lo que los procesos a realizar debían estar en un ciclo continuo sin detenerse.
- Numero de procesos a realizar: Se identificaron cuatro procesos que debían ejecutarse paralelamente con el fin de tener la menor exactitud y latencia de comunicación entre dispositivos y API, los cuales son: Consulta de API, comunicación con variador, comunicación con analizador y activación de relé para el bloqueo de rotor.
- Actualización de variables: Se identifico la necesidad de mantener de forma local un registro de determinadas variables para poder realizar una comparación continua frente al estado recibido de la API y evitar comunicarse con los dispositivos cuando no era necesario. Dichas variables corresponden a la frecuencia del variador y al estado actual del variador, es decir, si está corriendo o por el contrario se encuentra en paso. En el caso de que las variables recibidas sean diferentes a las almacenadas localmente se hará la comunicación con el variador de frecuencia con los nuevos datos, si por el contrario no hay cambios, se mantiene la comunicación sin variación con este dispositivo.

Con estos puntos críticos identificados se realizó el flujo que el programa debía tener, tal como se muestra en la figura 27 en donde es posible identificar cinco hilos que se ejecutan de manera paralela cada uno de estos ejecutando una función establecida para el control de los dispositivos y la comunicación con la API.

El primer hilo como se muestra se encarga de consultar la API y actualizar variables locales si es necesario, el segundo se encarga de mantener una comunicación continua con el variador para mantenerlo en marcha cuando corresponda, ya que es necesario enviarle continuamente el comando que hace que este dispositivo se mantenga en este estado. El tercer hilo verifica continuamente si es necesario pausar el rotor, por otro lado el cuarto hilo es el que valida que botón ha pulsado el usuario desde el aplicativo web, para mandar la acción al analizador. Por último, el quinto hilo se encarga de tomar la imagen en pantalla del analizador y enviarla hacia la API.

8.6.1. Aplicativo Web - UstaLab

En el desarrollo del aplicativo Web se eligió el patrón de arquitectura MVC esto debido a que es una arquitectura bastante fácil de entender y de la cual se tenía mayor conocimiento practico. Desde un principio se había establecido al menos la necesidad de implementar tres vistas para el aplicativo web, cada una de estas iban a corresponder a la vista de login, vista home y la vista de información o about. Lo que ayudo a establecer los primeros controladores: LoginController, HomeController y WikiController.

Cada uno de estos controladores son quienes devolverán las vistas a mostrar al usuario al momento de navegar en el aplicativo web. Para el diseño de estas vistas se tomó de referencia la plantilla denominada "SB-Admin 2"[30] la cual a su vez hace uso de Bootstrap una biblioteca de

hojas de estilo, archivos HTML y scripts en JavaScript con los cuales se diseñan la parte gráfica de las paginas web.

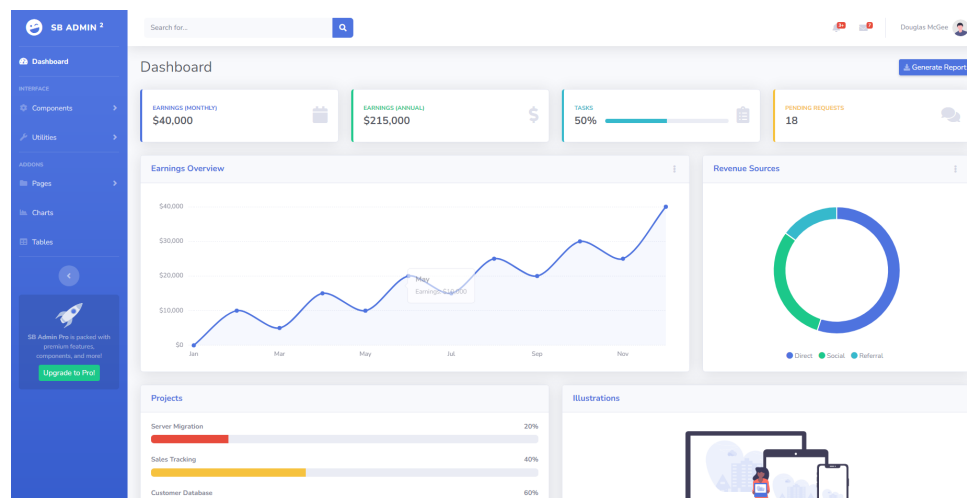


FIGURA 28: Plantilla SB-Admin[30]

Partiendo de esta base inicial se diseñaron las vistas de Home y Wiki. En la primera se muestra una imagen de referencia de la planta que van a controlar, junto a dos botones para iniciar cualquiera de las dos pruebas que se pueden realizar por medio del aplicativo. Figura 29

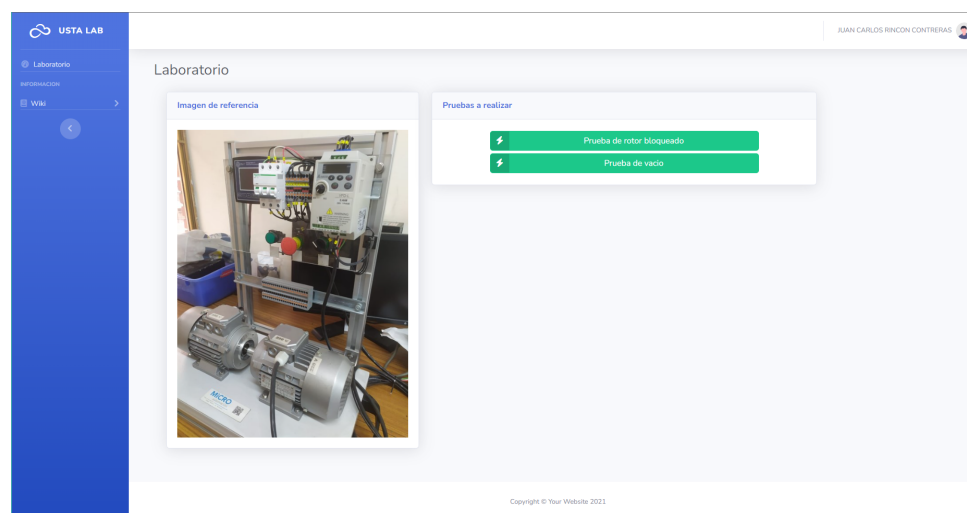


FIGURA 29: Vista Home del aplicativo Web. Autoría propia

Al seleccionar cualquiera de las dos pruebas se muestran los diferentes controles para cada una de las pruebas, a su vez se muestra la imagen de las medidas actuales que está tomando el analizador como se aprecia en la figura 30 y 31. El manejo de estos controles se explica a profundidad en el Anexo 1.

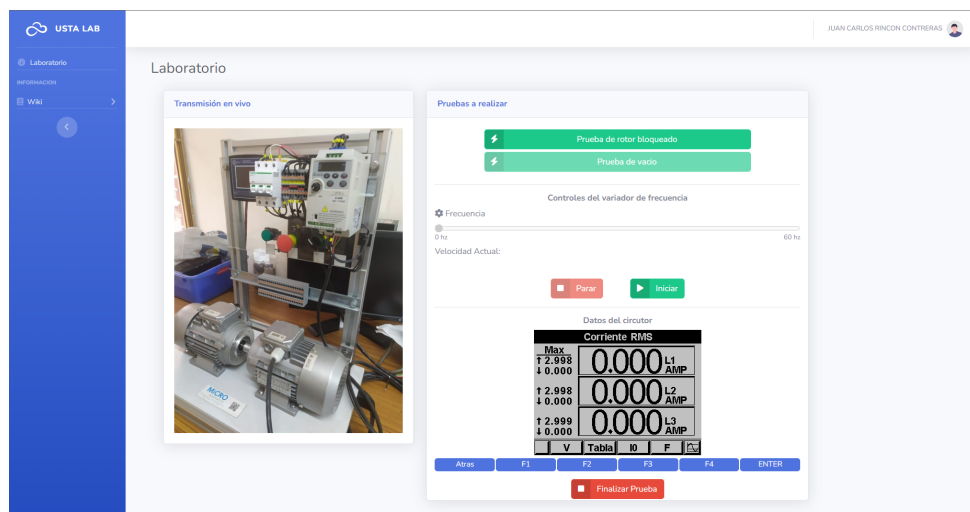


FIGURA 30: Vista Home del aplicativo Web con los controles de la prueba de vacío. Autoría propia

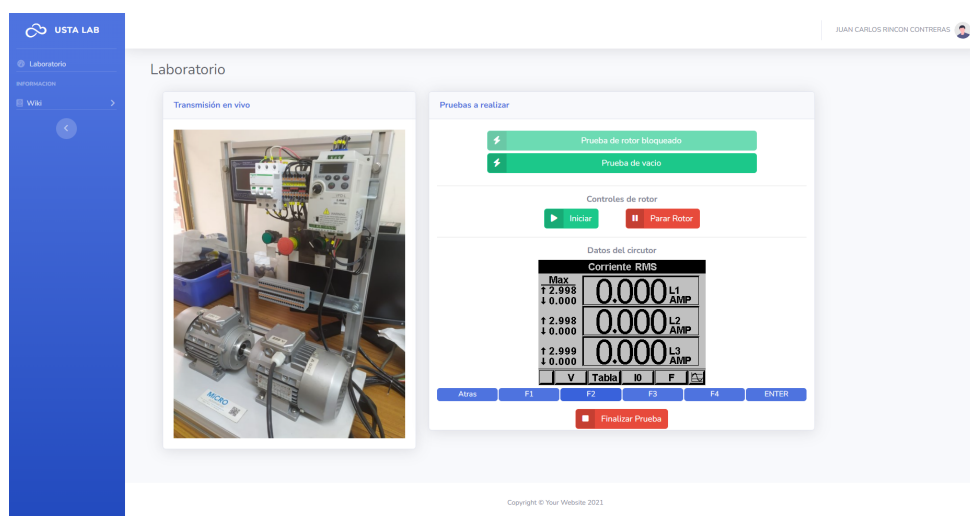


FIGURA 31: Vista Home del aplicativo Web con los controles de la prueba de rotor bloqueado. Autoría propia

Por otro lado la vista Wiki es un poco mas sencilla, se muestra información a profundidad de las pruebas que se pueden realizar en el aplicativo para dar al usuario un soporte teórico de ambas pruebas.

Por último se tiene la vista de Login figura 32 , en la cual no se usó la plantilla si no que se hizo un diseño basado en lo que se tiene en las hojas de estilo que ofrece Bootstrap de manera gratuita.

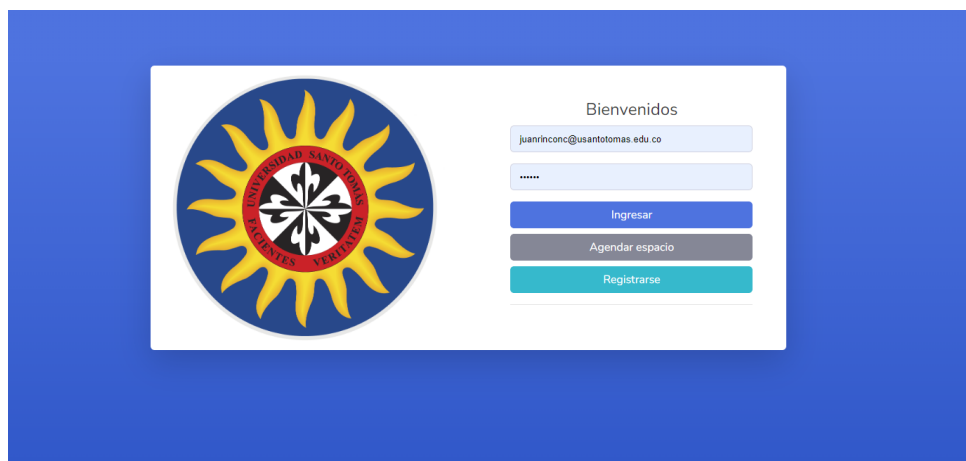


FIGURA 32: Vista Login del aplicativo web. Autoría propia

8.6.1.2. Controladores

En el modelo MVC los controlares es en donde se desarrolla la lógica del aplicativo y en donde se gestionan como llegar hasta las vistas ya diseñadas. Como se mencionó anteriormente se desarrollaron tres controladores LoginController, HomeController y WikiController de forma inicial para que mostraran las vistas, sin embargo, se desarrollaron otros dos controladores más para la lógica de las pruebas a los motores: FuenteController y PruebaVacioController.

Para la funcionalidad de login se implementó el flujo mostrado en la figura 33 en el cual se puede evidenciar el consumo de la API desarrollada para realizar validaciones tales como la existencia del usuario en la base de datos y la fecha que el usuario tiene apartada para usar la plataforma. En el caso de que alguna de las dos validaciones restrinja el ingreso al usuario se mostrara en la pantalla de login un mensaje con la razón por la cual no puede ingresar.

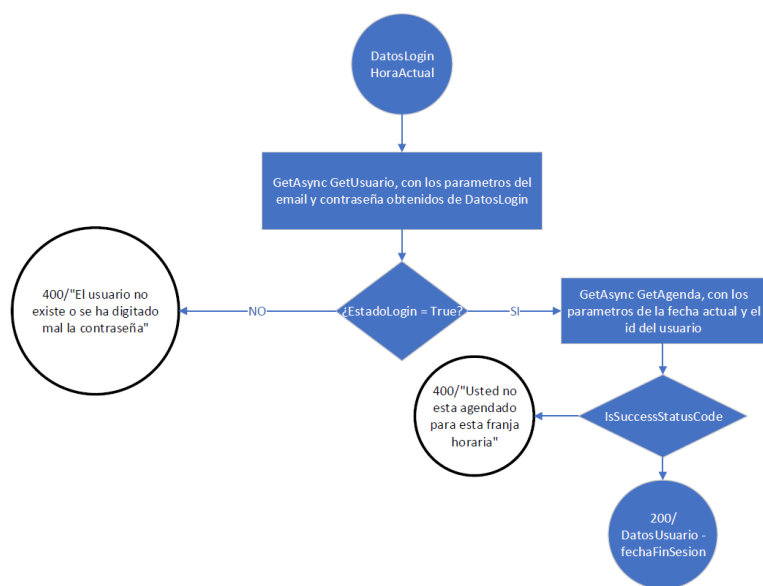


FIGURA 33: Flujo para el Login desde el aplicativo. Autoría propia

Además, en este controlador también se implementaron la lógica para la función de agendar horario y registro de usuarios, los flujos que se siguieron para esto se muestran en la figura 34 y en la figura 35 en cada uno de los flujos se evidencia que la lógica del aplicativo no se lleva completamente en el aplicativo si no que se realizan en la API y dependiendo de la respuesta de la API se comporta el aplicativo web.

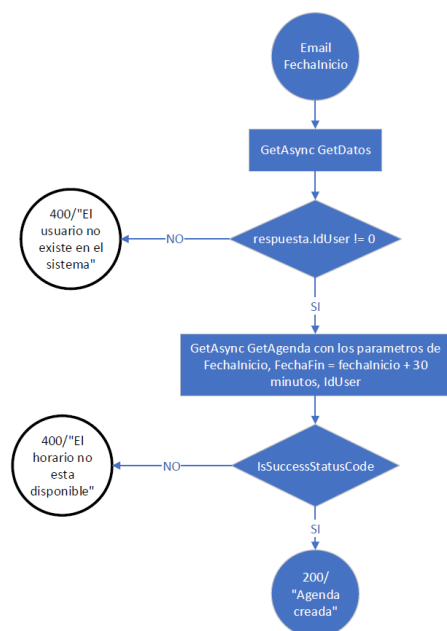


FIGURA 34: Flujo para el agendamiento desde el aplicativo. Autoría propia

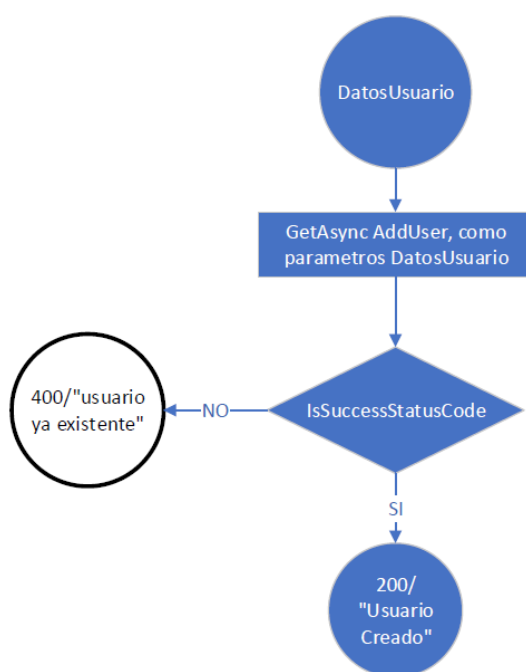


FIGURA 35: Flujo para el registro de usuarios dentro del aplicativo. Autoría propia

En el caso de los controladores FuenteController y PruebaVacioController de implementa la lógica necesaria consumir las funciones de la API que se encargan de traer la imagen del circutor,

modificar velocidad de variador, detener rotor y controlar el circutor. Diagramas como el de la figura 36 muestra como desde el controlador se envía la información a la API para poder cambiar la velocidad el variador.

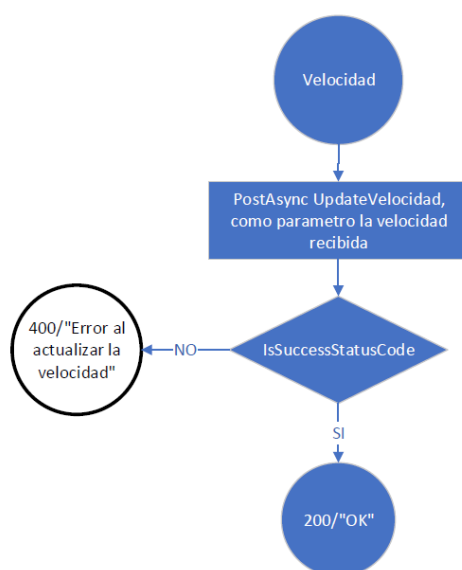


FIGURA 36: Flujo para el cambio de velocidad dentro del aplicativo. Autoría propia

8.7. Pruebas de funcionalidad

Con el desarrollo de los tres aplicativos necesarios para el funcionamiento adecuado del laboratorio remoto se realizaron varias pruebas de funcionamiento para validar la integración completa entre el aplicativo web y la planta física. En las cuales se comprobaron los siguientes puntos:

- Inicio de sesión.
- Creación de usuarios.
- Agendamiento de sesiones.
- Ejecución de la prueba de vacío.

- Ejecución de la prueba de rotor bloqueado.

De las pruebas realizadas se lograron obtener los siguientes resultados:

8.7.1. Inicio de sesión

Para comprobar el correcto funcionamiento del inicio de sesión se hicieron pruebas en donde se inició sesión con correos no registrados, con una contraseña incorrecta, cuando no se ha agendado el espacio para usar el aplicativo. Cuando se tienen todos los requerimientos correctos para ingresar el aplicativo muestra directamente la vista de home que se muestra en la figura 29. Los resultados de cada una de estas pruebas se muestran en las figuras 37, 38, 39



FIGURA 37: Intento de ingreso al aplicativo con un correo no registrado. Autoría propia



FIGURA 38: Intento de ingreso al aplicativo con una contraseña errónea. Autoría propia



FIGURA 39: Intento de ingreso al aplicativo sin agenda apartada. Autoría propia

8.7.2. Creación de Usuarios

Los resultados de esta prueba se muestran en las figuras 40, 41 en donde se hace el intento de crear un usuario con un correo ya registrado y una segunda prueba con un correo nuevo. De esta manera se valida que los flujos implementados para este proceso funcionan correctamente.

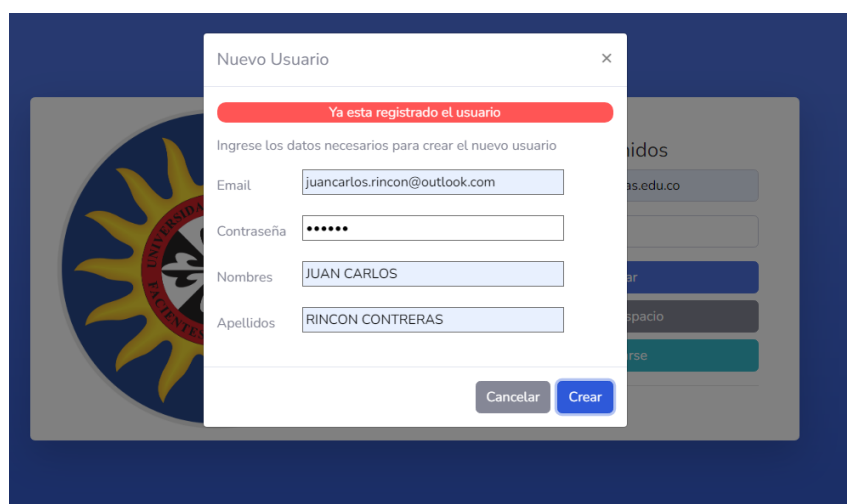


FIGURA 40: Creación de usuario ya registrado en el sistema. Autoría propia

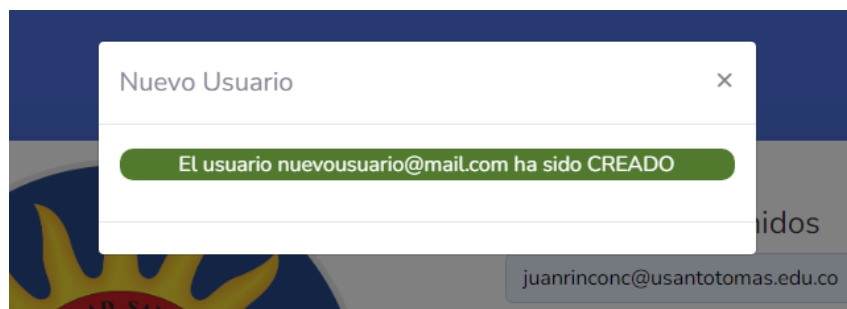


FIGURA 41: Creación de usuario. Autoría propia

8.7.3. Agendamiento de Sesiones

En el agendamiento de sesiones se realizan pruebas tanto con correo no registrados como correos ya registrados y de la misma forma programando sesiones los fines de semana. Esto con para comprobar que el sistema realiza las validaciones necesarias antes de crear el agendamiento.

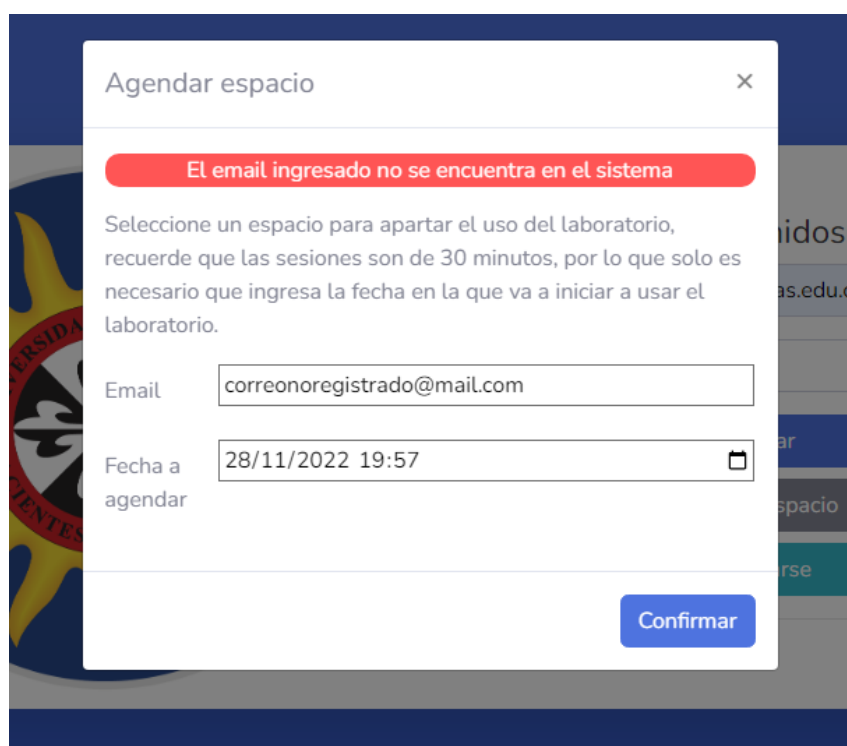
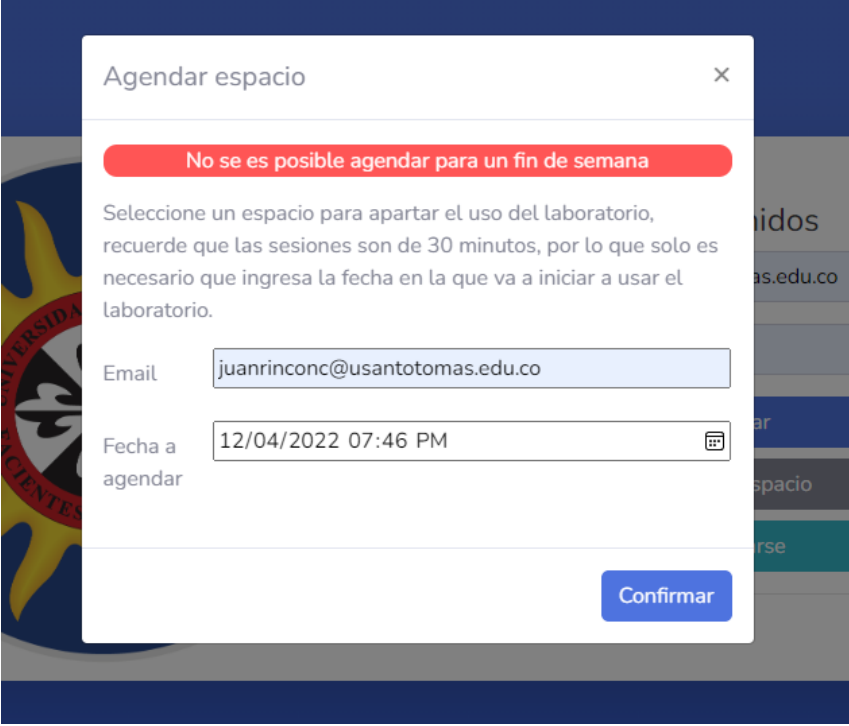


FIGURA 42: Agendamiento con un correo no existente. Autoría propia



The image shows a web browser window with a modal form titled "Agendar espacio" (Book space). The form has a red error message at the top: "No se es posible agendar para un fin de semana" (It is not possible to book for a weekend). Below the message, there is a text area with instructions: "Seleccione un espacio para apartar el uso del laboratorio, recuerde que las sesiones son de 30 minutos, por lo que solo es necesario que ingresa la fecha en la que va a iniciar a usar el laboratorio." (Select a space to book the use of the laboratory, remember that the sessions are 30 minutes, so it is only necessary that you enter the date when you are going to start using the laboratory). The form contains two input fields: "Email" with the value "juanrinconc@usantotomas.edu.co" and "Fecha a agendar" (Date to book) with the value "12/04/2022 07:46 PM". A blue "Confirmar" (Confirm) button is located at the bottom right of the form. The background of the browser window shows a university logo and some text.

FIGURA 43: Agendamiento en un fin de semana. Autoría propia

En los casos mostrados en las figuras 42, 43 se ve como el sistema en una primea ocasión valida que el correo no se encuentra registrado y en una segunda ocasión evita crear un agendamiento para un fin de semana.

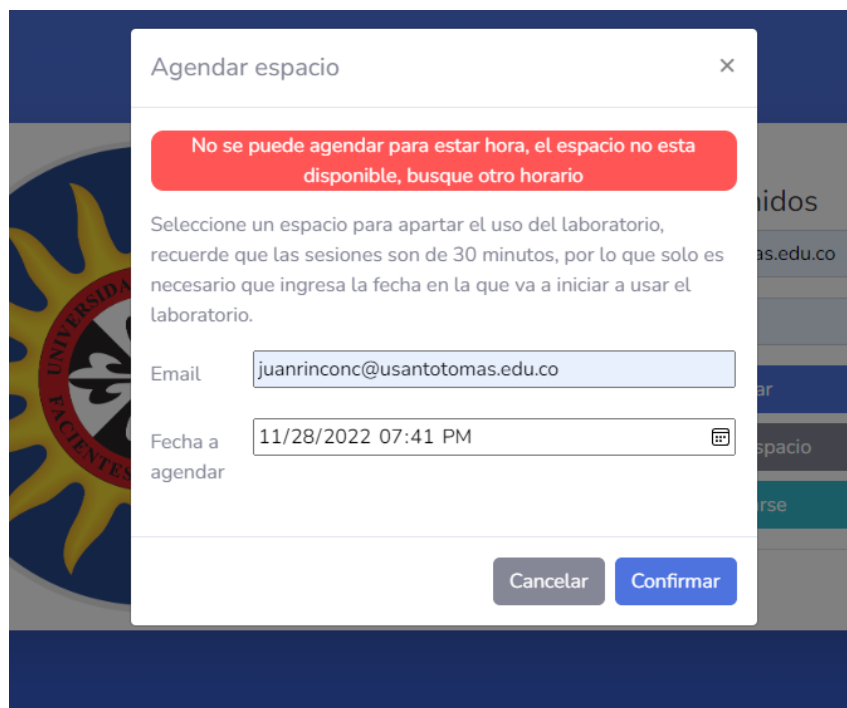


FIGURA 44: Agendamiento en una fecha ya programada. Autoría propia

De igual forma se hace una prueba programando la sesión en una hora en donde ya hay otra programada tal como se muestra en la figura 44 el sistema responde con un mensaje indicando que este espacio ya ha sido apartado por otro usuario.



FIGURA 45: Paso 1 para agendar una sesión. Autoría propia



FIGURA 46: Mensaje de confirmación del agendamiento. Autoría propia

Por último se hace la prueba con todos los datos válidos para apartar el espacio, tal como se muestra en las figuras 45, 46 el sistema mediante un mensaje confirma la creación del espacio.

8.7.4. Ejecución de la prueba de vacío

Se realizó la ejecución de la prueba de vacío para validar su correcto funcionamiento y que los datos mostrados en pantalla en el aplicativo corresponden a los que se tienen realmente en la planta física.

Para esto se inició con la prueba desde el aplicativo en donde se validó primero la puesta en marcha y el paro del variador de frecuencia, desde el aplicativo se pulsa el botón de iniciar y a los pocos segundos el variador pasa de stop a una frecuencia mínima. Al momento de dar el botón de pausa el variador baja su frecuencia hasta estar nuevamente en stop.

Laboratorio

Transmisión en vivo



Pruebas a realizar

⚡ Prueba de rotor bloqueado

⚡ Prueba de vacío

Controles del variador de frecuencia

⚙ Frecuencia

10 hz 60 hz

Velocidad Actual: 10Hz

■ Parar ▶ Iniciar

Datos del circutor

Volt RMS	
Max ↑ 230.0 ↓ 0.000	27.09 L1 VOLT
↑ 230.0 ↓ 0.000	25.93 L2 VOLT
↑ 230.0 ↓ 0.000	26.96 L3 VOLT

V I Tabla F

Atras F1 F2 F3 F4 ENTER

■ Finalizar Prueba

FIGURA 47: Ejecución de la prueba desde el aplicativo web. Autoría propia

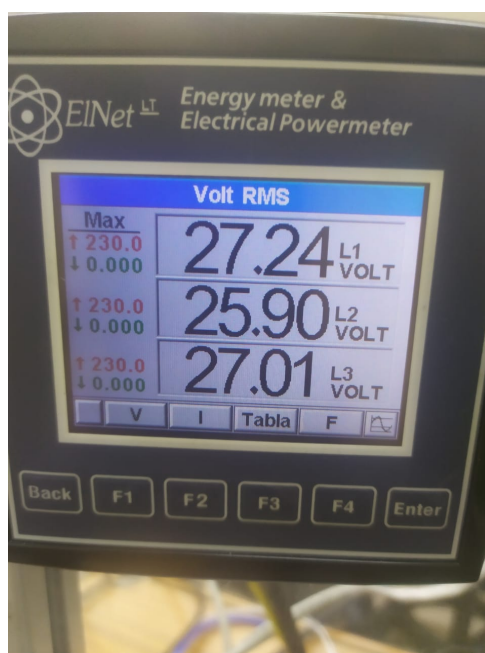


FIGURA 48: Valores de voltaje mostrados desde el circutor. Autoría propia



FIGURA 49: Variador puesto en marcha con 10Hz. Autoría propia

La segunda prueba es el cambio de frecuencia en el variador, al modificar la frecuencia desde el aplicativo web se evidencia este cambio en el variador tal como se muestra en las figuras 50 y 52.

Laboratorio

Transmisión en vivo



Pruebas a realizar

- Prueba de rotor bloqueado
- Prueba de vacío

Controles del variador de frecuencia

Frecuencia

10 hz ————— 60 hz

Velocidad Actual: 60Hz

Datos del circuito

Corriente RMS	
Max ↑ 2.998 ↓ 0.000	0.459 L1 AMP
↑ 2.998 ↓ 0.000	0.299 L2 AMP
↑ 2.999 ↓ 0.000	0.453 L3 AMP

FIGURA 50: Ejecución de la prueba desde el aplicativo web, mostrando datos de corriente. Autoría propia

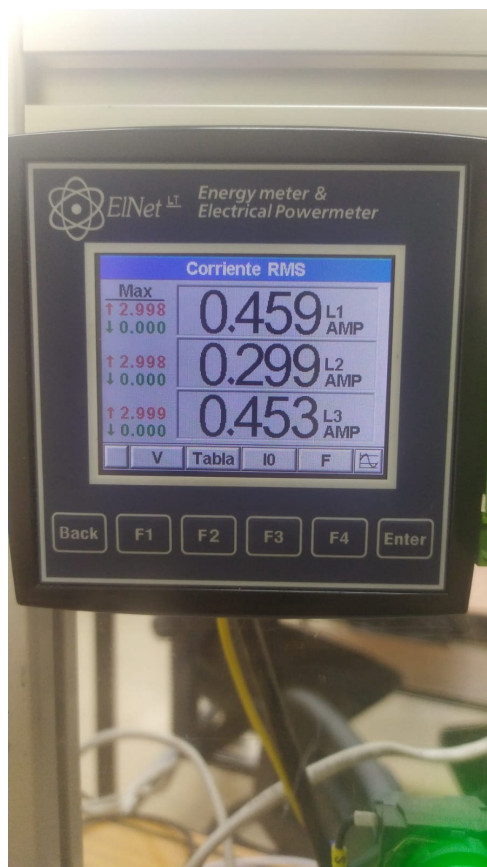


FIGURA 51: Valores de corriente mostrados desde el circutor. Autoría propia



FIGURA 52: Variador puesto en marcha con 60Hz. Autoría propia

De la misma forma se verifica que la imagen del circutor corresponde con la imagen que se muestra en el aplicativo, con las figuras 48 y 47 se puede comprobar que ambas imágenes tienen valores cercanos y su diferencia se debe al retraso que se generan en este tipo de comunicaciones.

8.7.5. Ejecución de la prueba de rotor bloqueado

Al igual que para la anterior prueba, se hizo una validación del funcionamiento de esta prueba, al momento de pulsar el botón de iniciar se pone en marcha el variador a una velocidad fija de 10Hz esto para que al momento de bloquear el rotor no se necesite demasiada corriente, ya que podría causar un corto.

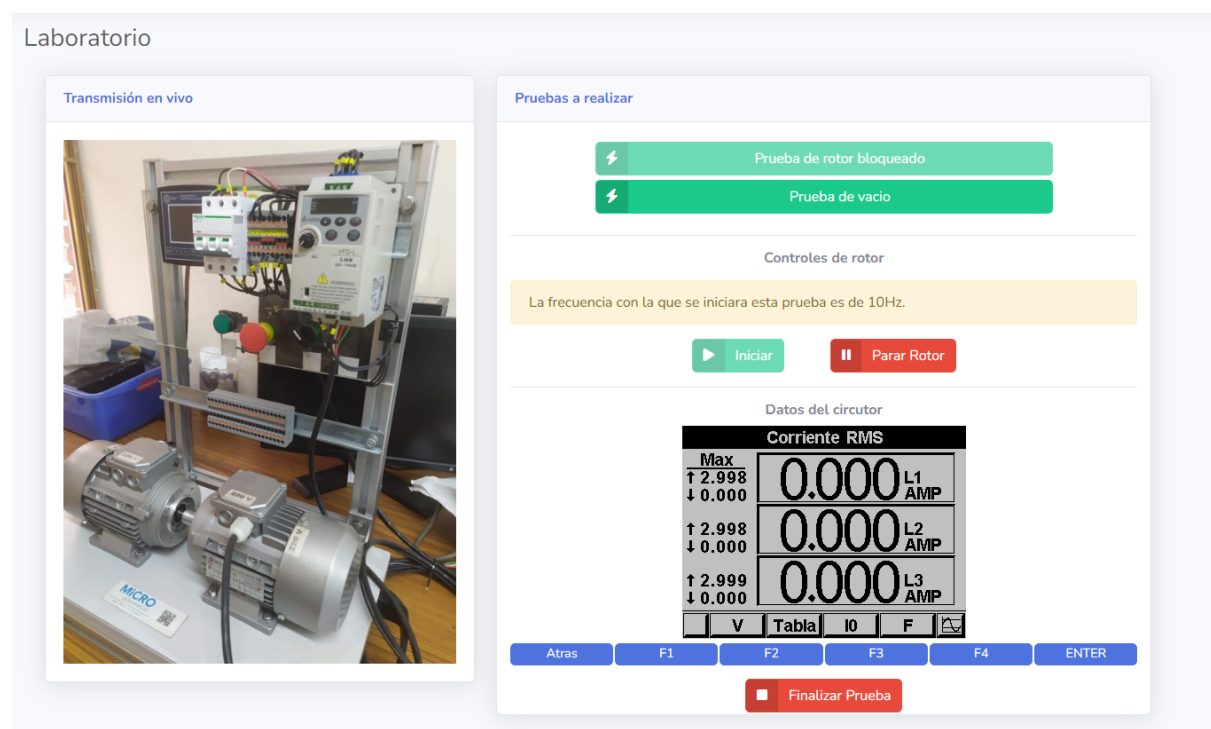


FIGURA 53: Inicio de prueba de rotor bloqueado. Autoría propia

Posterior a esto se puede activar el botón de bloquear rotor, lo que activa inmediatamente la conexión entre la fuente y los devanados del rotor, y detiene el rotor por 8 segundos, durante ese tiempo como se muestra en la figura 54 la medición corriente aumenta, comportamiento que se esperaba al bloquear el rotor.

Laboratorio

Transmisión en vivo



Pruebas a realizar

- ⚡ Prueba de rotor bloqueado
- ⚡ Prueba de vacío

Controles de rotor

La frecuencia con la que se iniciará esta prueba es de 10Hz.

▶ Iniciar ⏸ Parar Rotor

El rotor será bloqueado por 8 segundos, tiempo restante: 2

Datos del circuito

Corriente RMS	
Max ↑ 2.998 ↓ 0.000	0.494 L1 AMP
↑ 2.998 ↓ 0.000	0.329 L2 AMP
↑ 2.999 ↓ 0.000	0.488 L3 AMP

⏪ V Tabla IO F ⏩

Atras F1 F2 F3 F4 ENTER

■ Finalizar Prueba

FIGURA 54: Corriente mostrada en el aplicativo al momento de bloquear el rotor.
Autoría propia



FIGURA 55: Corriente mostrada en el circutor al momento de bloquear el rotor.
Autoría propia

8.8. Encuesta a estudiantes

Para validar el impacto y aceptación de la implementación de un laboratorio remoto para pruebas de vacío y rotor bloqueado se realizaron pruebas con estudiantes de pregrado de la Facultad de ingeniería electrónica de la Universidad Santo Tomás, concretamente con aquellos estudiantes que estuvieran cursado o hubiesen cursado la asignatura de Conversión electromagnética, esto con la finalidad de garantizar que los usuarios estuvieran familiarizados con el desarrollo de la prueba de vacío y rotor bloqueado tradicional y poder contrastar su experiencia del desarrollo de una práctica presencial con la de una realizada mediante un laboratorio remoto. Se hicieron dos pruebas piloto, la primera, totalmente asistida en donde se explicó a los alumnos el funcionamiento completo de la planta y del aplicativo web. Posterior a esto se programó una segunda prueba en la cual los participantes interactuaron por su cuenta con el laboratorio remoto agendando el espacio de 30 minutos dentro del aplicativo y realizando cada una de las pruebas disponibles.

La encuesta realizada consta de diez preguntas en donde se buscó medir el impacto y la viabilidad de este tipo de herramientas en el desarrollo del aprendizaje de los estudiantes. Los resultados se presentan en las figuras de la 56 a la 65.

1. El aplicativo web utilizado para el desarrollo de las pruebas de vacío y rotor bloqueado es intuitivo y de fácil manejo.

7 respuestas

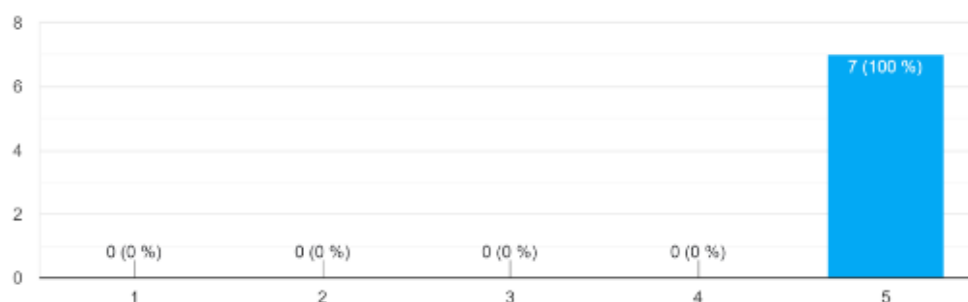


FIGURA 56: Resultados encuesta pregunta 1. Autoría propia

2. La disposición del aplicativo web es organizada y clara para el usuario.

7 respuestas

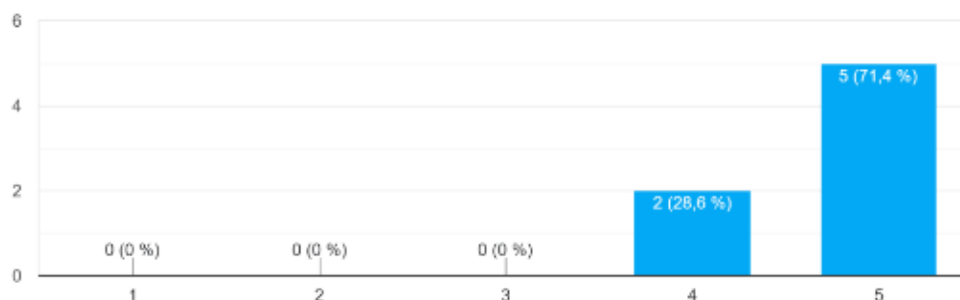


FIGURA 57: Resultados encuesta pregunta 2. Autoría propia

3. El funcionamiento del aplicativo web responde de manera correcta a lo largo de su ejecución, permitiendo la realización de agendamiento de hora...ción de las pruebas de vacío y rotor bloqueado.

7 respuestas

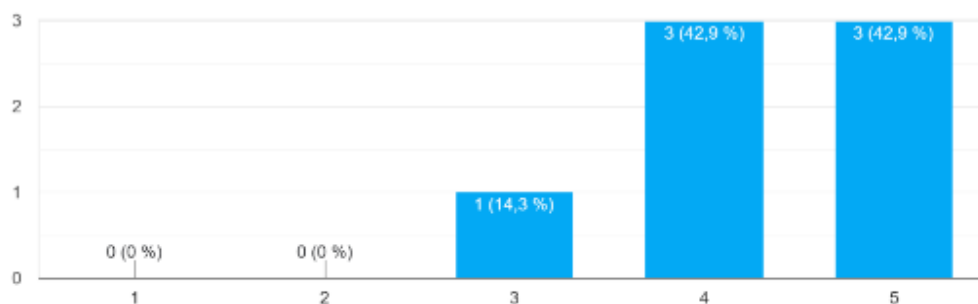


FIGURA 58: Resultados encuesta pregunta 3. Autoría propia

4. El laboratorio remoto permite medir las características principales del motor involucrado en las pruebas de vacío y rotor bloqueado.

7 respuestas

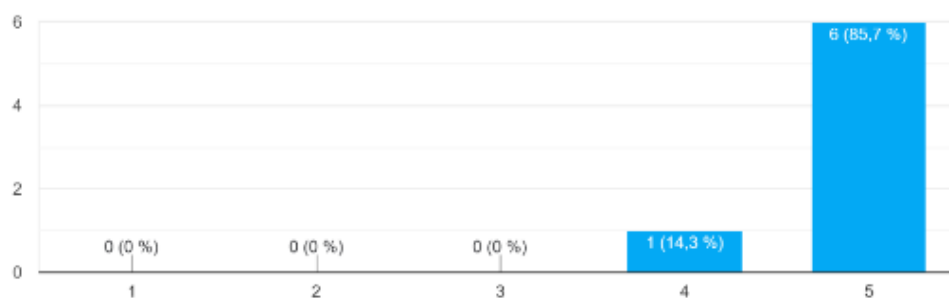


FIGURA 59: Resultados encuesta pregunta 4. Autoría propia

5. El laboratorio remoto permite la comprensión del desarrollo de las pruebas de vacío y rotor bloqueado.

7 respuestas

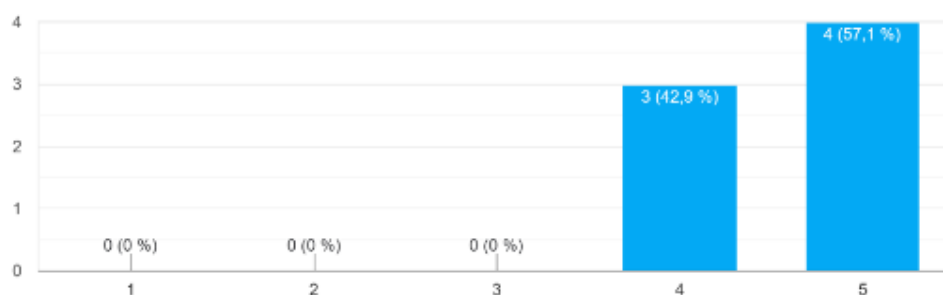


FIGURA 60: Resultados encuesta pregunta 5

6. El laboratorio remoto refuerza el aprendizaje de las temáticas relacionadas a la asignatura Conversión Electromagnética.

7 respuestas

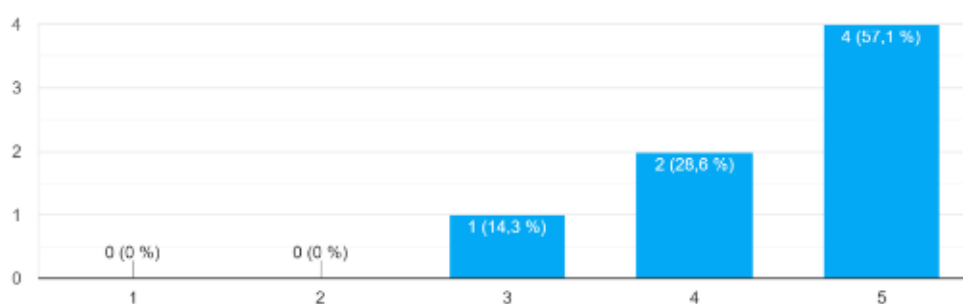


FIGURA 61: Resultados encuesta pregunta 6

7. Utilizaría el aplicativo web para realizar pruebas de vacío y rotor bloqueado en futuras experiencias de aprendizaje.

7 respuestas

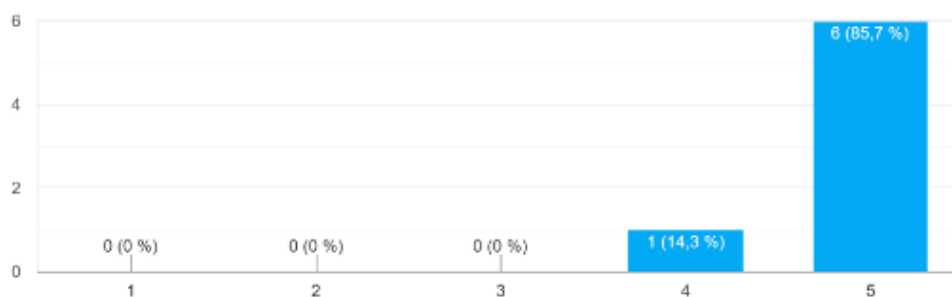


FIGURA 62: Resultados encuesta pregunta 7

8. Considero el laboratorio remoto como un facilitador de mi proceso de aprendizaje universitario.

7 respuestas

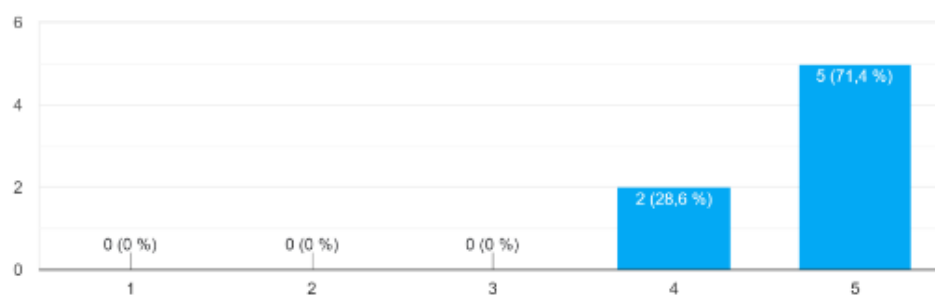


FIGURA 63: Resultados encuesta pregunta 8. Autoría propia

9. Considero que el laboratorio remoto mejora el acceso al aprendizaje, eliminando obstáculos referentes a desplazamiento, falta de equipos de laboratorio, tiempo, entre otros.

7 respuestas

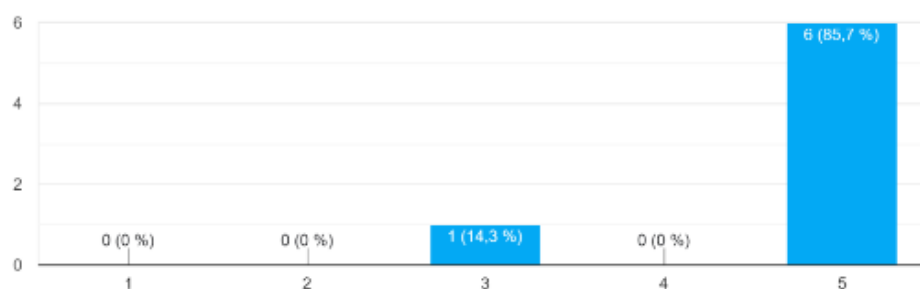


FIGURA 64: Resultados encuesta pregunta 9. Autoría propia

10. Recomendaría a otros estudiantes la utilización del laboratorio remoto para la comprensión de las temáticas que estos engloban.

7 respuestas

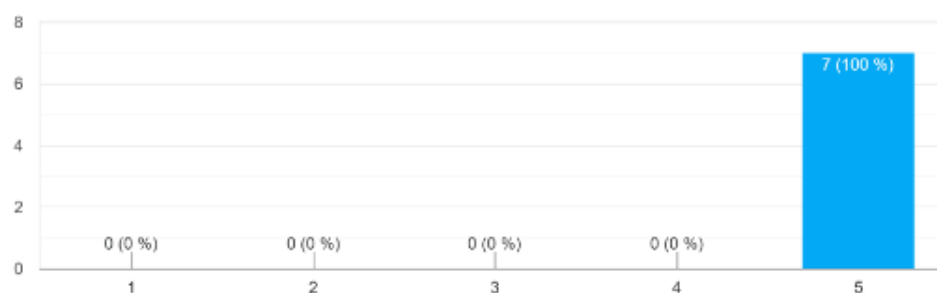


FIGURA 65: Resultados encuesta pregunta 10. Autoría propia

Capítulo 9

Análisis de resultados

A partir de los resultados obtenidos y teniendo en cuenta los objetivos a cumplir con este proyecto se puede evidenciar que todos los procesos realizados durante el desarrollo de este mismo han llevado al cumplimiento de cada uno de ellos de manera satisfactoria. Siendo así, se enumeran a continuación para su análisis.

9.1. Acondicionamiento de la planta física

Tal como se mostró en la sección 8.1.1 la planta física fue adecuada con el fin de poder realizar las medidas correspondientes a voltaje y corriente para la realización de las pruebas de rotor bloqueado y vacío. Además de la adecuación de conexiones fue necesario la integración con la Raspberry PI la cual es el centro de comunicación con cada uno de los dispositivos de medición en la planta.

La correcta adecuación de esta planta se ve reflejada en los resultados obtenidos en las pruebas de funcionamiento, en donde se puede evidenciar una medición correcta de las variables de interés al momento de realizar las pruebas al motor.

9.2. Desarrollo de la plataforma Web

El uso correcto de este laboratorio remoto parte desde la experiencia que se le puede presentar al usuario desde el aplicativo web. Desde la página de login se puede evidenciar lo simple y clara que es la plataforma, ofreciendo mensajes al usuario cada vez que se presenta un

error o cuando logra realizar un proceso con éxito, tal como se muestra en las pruebas de funcionamiento de la sección 8.7 al momento de crear usuarios o agendar espacios para uso del laboratorio.

De igual forma al momento de realizar las pruebas la plataforma es clara con las posibilidades que ofrece al usuario y la información que se le muestra. Los botones del aplicativo están configurados para evitar el mal uso por parte de usuario, activándose o desactivándose solo en momentos necesarios, tal como se muestra en la figura 66 de la prueba de rotor bloqueado en donde el botón de parar rotor solo es seleccionable al momento de iniciar la prueba, no antes.



FIGURA 66: Botón deshabilitado hasta comenzar prueba de rotor bloqueado.

La implementación de un API que funcione como intermediario entre el aplicativo web y la aplicación de la Raspberry PI ofrece velocidad y paralelismo al aplicativo, lo que permite tener una respuesta rápida al momento de modificar velocidades, parar o poner en marcha el variador y obtener imágenes de las medidas actuales que se tienen desde el circutor.

9.3. Pruebas de funcionamiento.

Las pruebas de funcionamiento son el resultado más importante para el proyecto, son la muestra de que el sistema integrado funciona y logra realizar las tareas para las cuales fue diseñado. Con los datos que se le proporcionan al estudiante de voltaje, corriente y potencia es capaz de completar los cálculos pertinentes para estas pruebas.

Con los resultados mostrados en la sección 8.7 en donde se valida diferentes funcionalidades tanto del aplicativo web como de la ejecución de las pruebas en los motores se valida que es posible realizar estas pruebas sin problemas desde el aplicativo, ofréndole al estudiante una herramienta con la cual puede realizar este tipo de prácticas de laboratorio desde un navegador web.

9.4. Impacto en los estudiantes

Al observar el resultado de las encuestas se evidencia un impacto positivo en los estudiantes al poder desarrollar practicas de una forma alterna a la convencional. Según el resultado de las preguntas 1 y 2 el aplicativo web es amigable con el usuario, intuitivo y fácil de usar con una oportunidad de mejora en cuanto a la organización de los ítems.

Con los resultados de las preguntas 3, 4 y 5 se evidencia la percepción de los estudiantes en cuanto al funcionamiento del laboratorio remoto, mostrando un resultado muy satisfactorio en cuando a la medición de variables y ejecución de pruebas.

Finalmente las preguntas 6, 7, 8,9 y 10 se enfocaron en la relación del laboratorio remoto con el proceso de aprendizaje de los estudiantes dando resultados muy positivos en cuanto a la aceptación y recomendación a otros usuario para el desarrollo de prácticas remotas sin embargo la propuesta de esta herramienta se planteo desde el principio como un recurso adicional al proceso de aprendizaje, reforzando las temáticas relacionadas con la asignatura y no como un remplazo a las practicas de laboratorio convencionales.

Capítulo 10

Conclusiones

- La integración de la tarjeta de desarrollo Raspberry PI a la planta física ha funcionado como puente de comunicación entre la plataforma web y los dispositivos de medición y control de los motores, logrando de esta manera un control preciso de la velocidad de los motores y la puesta en marcha y paro de estos mismos, al igual que una medición acertada de voltajes, corrientes y potencias por medio del analizador ElNet LT.
- La plataforma web permite a los estudiantes de la Universidad Santo Tomas hacer uso del laboratorio remoto fácilmente, realizar las pruebas de vacío y rotor bloqueado en motores de inducción en horarios cómodos para ellos, con una interfaz intuitiva y fácil de usar, ayudando de esta forma a reducir los tiempos que normalmente toma realizar este tipo de prácticas.
- Los resultados presentados de las pruebas de funcionamiento son evidencia de la correcta operación del laboratorio remoto. Demostrando así la integración entre aplicativo web y planta física, en donde es posible observar casi a tiempo real las mediciones tomadas por el analizador ElNet LT y controlar la puesta en marcha, el paro y la velocidad del variador de frecuencia de forma remota.
- Por medio de las encuestas realizadas se pudo evidenciar la opinión positiva de los estudiantes frente a este tipo de herramientas, al considerarla una alternativa viable para el desarrollo de este tipo de practicas de forma rápida y segura, reforzando las temáticas vistas en clases.

Capítulo 11

Apéndice

- Anexo 1. Manual de uso aplicativo web.

Guia de laboratorio remoto para pruebas de vacío y rotor bloqueado

Facultad de ingeniería electrónica

Asignatura: Conversión electromagnética

Objetivo General:

Por medio del aplicativo web damian216-001-site1.ctempuri.com podrá realizar de manera remota pruebas de vacío y rotor bloqueado en motores de inducción. (Se recomienda el uso de Chrome como explorador)

Procedimiento:

1. **Registro:** Para acceder a la plataforma web es necesario registrar un usuario,, dando click en el botón de **Registrarse** en la pantalla de inicio.



Bienvenidos

Ingresar

Agendar espacio

Registrarse

Para el registro se solicita el correo institucional y contraseña junto con el nombre y apellido del estudiante.

Nuevo Usuario

×

Ingrese los datos necesarios para crear el nuevo usuario

Email

Contraseña

Nombres

Apellidos

Cancelar

Crear

2. **Agendamiento:** Una vez generado el usuario es necesario agendar el horario en que se desea acceder a la plataforma para desarrollar las pruebas dando click en el botón de **Agendar espacio** en la pantalla de inicio.

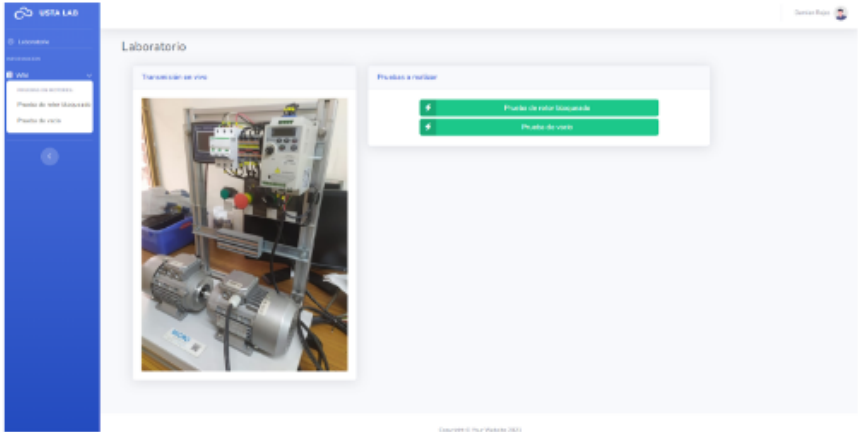
Agendar espacio ×

Seleccione un espacio para apartar el uso del laboratorio, recuerde que las sesiones son de 30 minutos, por lo que solo es necesario que ingresa la fecha en la que va a iniciar a usar el laboratorio.

Email

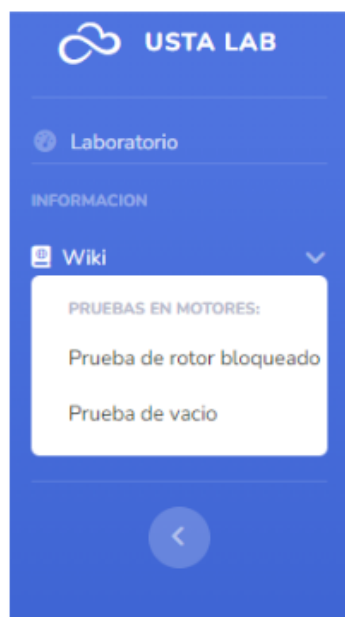
Fecha a agendar

3. Ingreso: Al dar click en el botón **Ingreso** podrá acceder al aplicativo web (tenga en cuenta ingresar en el horario previamente agendado) en donde encontrará la siguiente vista.



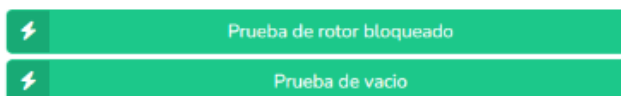
The screenshot shows the VISTA LAB web application. On the left is a blue sidebar menu with the following items: 'Inicio', 'Laboratorio', 'Proyectos', 'Procesos', 'Procedimientos', 'Problemas de mantenimiento', 'Problemas de seguridad', and 'Problemas de calidad'. The main content area is titled 'Laboratorio'. It features a 'Transferencia de datos' section with a photograph of a laboratory setup involving a motor and various electronic components. To the right of this is a 'Proyectos a realizar' section containing a table with two rows of tasks, each with a green progress bar and a status indicator.

En esta sección encontrará un menú desplegable con información teórica de las pruebas a su izquierda(Wiki)



También encontrará el menú con las pruebas disponibles en el aplicativo Web

Pruebas a realizar



4. **Inicio de pruebas:** Una vez seleccionada la prueba se proporciona la opción de acceder a un resumen teórico del funcionamiento y desarrollo de la respectiva prueba.

Prueba de rotor bloqueado



La prueba de rotor bloqueado consiste en bloquear el rotor del motor para que no se mueva, para posteriormente medir el voltaje, la corriente y la potencia. Si deseas mas informacion sobre esta prueba dar click en "Mas Informacion", si deseas continuar con el desarrollo de la prueba dar click en "Iniciar Prueba"

Cancelar

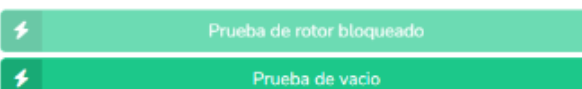
Mas informacion

Iniciar prueba

5. **Prueba de rotor bloqueado:** Al acceder a la prueba de rotor bloqueado observará la

siguiente pantalla, en donde encontrará los botones para bloquear el motor mediante inyección de DC realizar esta acción colocara en corto el motor, por motivos de seguridad el tiempo de bloqueo está limitado a 8 segundos. Deberá esperar este tiempo entre cada acción de bloqueo de rotor, los datos se mostrarán en tiempo real en la imagen que corresponde al circutor encargado de realizar las mediciones correspondientes.

Pruebas a realizar



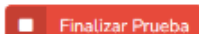
Controles de rotor

La frecuencia con la que se iniciara esta prueba es de 20Hz.



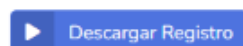
Datos del circutor

Corriente RMS		
Max		
↑ 2.998	0.000	L1 AMP
↓ 0.000		
↑ 2.998	0.000	L2 AMP
↓ 0.000		
↑ 2.999	0.000	L3 AMP
↓ 0.000		
	V	Tabla IO F



Después de pulsar el botón de **Parar Rotor** y transcurrido el tiempo de bloqueo se habilitará un botón de **Descargar Registro** mediante el cual podrá descargar la imagen del circutor con los datos capturados al momento de ejecutar la prueba.

Prueba Finalizada, pulse el boton de "Descargar Imagen" para obtener una imagen del circutor tomada al momento de hacer la prueba.



Datos del circutor

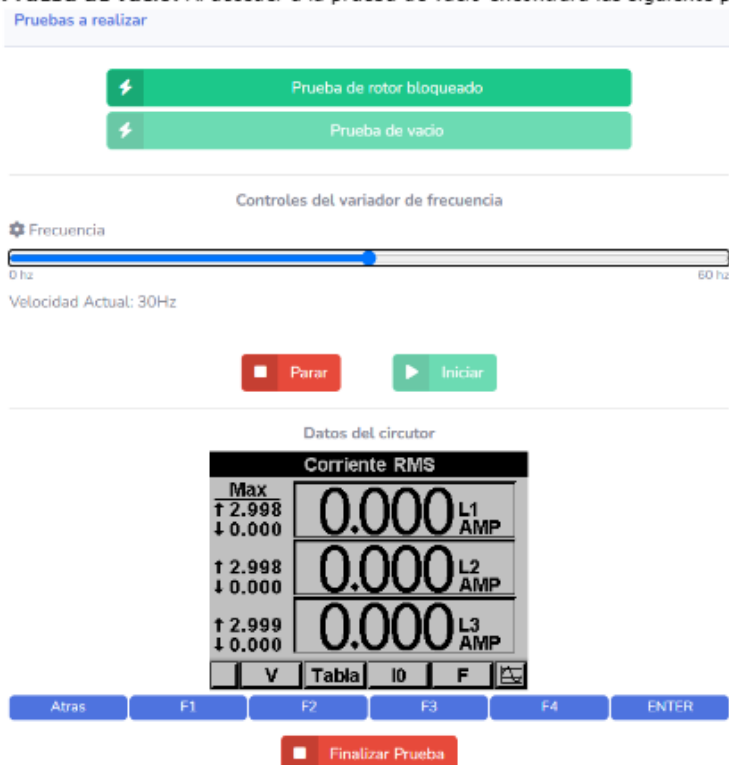
Corriente RMS		
Max		
↑ 2.998	0.000	L1 AMP
↓ 0.000		
↑ 2.998	0.000	L2 AMP
↓ 0.000		
↑ 2.999	0.000	L3 AMP
↓ 0.000		
	V	Tabla IO F



En la parte inferior de la imagen encontrará los botones **Atras**, **F1**, **F2**, **F3**, **F4**, **ENTER** con los

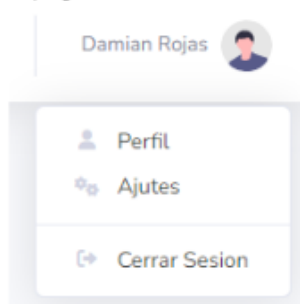
cuales podrá ejecutar acciones en el circutor en tiempo real, observando gráficas, cambiando las variables de medición o cualquier otra acción que permita realizar el circutor.

6. Prueba de vacío: Al acceder a la prueba de vacío encontrará las siguiente pantalla



en donde podrá variar la frecuencia del motor entre 0Hz y 60Hz observando en todo momento los datos en tiempo real entregados por el circutor. De igual manera podrá manipular mediante los botones **Atrás, F1, F2, F3, F4, ENTER** el circutor en tiempo real.

7. Cierre de sesión: Una vez finalizadas las pruebas encontrará en la parte superior derecha junto con su nombre un menú desplegable



en donde podrá salir del aplicativo web cerrando su sesión. Recuerde que pasado los 30 minutos correspondientes a cada franja agendada se cerrará su sesión automáticamente.

■ Anexo 2. Guía de laboratorio remoto.

LABORATORIO DE CONVERSIÓN ELECTROMAGNÉTICA

Prueba remota de vacío y rotor bloqueado

Objetivos

- Realizar la prueba de vacío a un motor de inducción trifásico.
- Realizar la prueba de rotor bloqueado (o prueba de cortocircuito) al motor de inducción trifásico.

Actividades previas

- Leer manual de usuario para el funcionamiento de la plataforma web.

Equipo necesario

La presente guía de laboratorio describe el procedimiento para realizar la práctica de rotor bloqueado y vacío en motores de inducción mediante la implementación de un aplicativo web de manera remota. Por lo cual únicamente requiere del link de acceso al aplicativo para llevar a cabo la práctica.

Así, en la sección de **Wiki** del aplicativo web encontrará todo el fundamento teórico para el desarrollo de las pruebas.

Prueba de rotor bloqueado

Para el desarrollo de una prueba convencional se incrementa cuidadosamente la tensión DC aplicada al motor, hasta que la corriente que toma sea la nominal. El rotor de la máquina debe estar bloqueado, de tal forma que no se le permite girar. Para el caso de la práctica de manera remota la tensión DC aplicada se encuentra fijada en el valor capaz de detener el rotor y en el circutor podrá observar todos los datos relevantes de corriente, potencia, factor de potencia y tensiones. Por seguridad y para evitar daños en los equipos de laboratorio el bloqueo del rotor está limitado a unos segundos, posterior a esto podrá descargar los datos registrados por el circutor.

Prueba de vacío

Se alimenta el motor a frecuencia nominal y se aumenta cuidadosamente la tensión hasta alcanzar su valor nominal. La máquina debe girar libremente, sin carga conectada a su eje. Para el desarrollo de esta prueba podrá variar la tensión del motor mediante el variador de frecuencia, usando el controlador del aplicativo y registrando la corriente de línea y potencia del motor en la imagen del circutor.

Actividades de laboratorio

Accediendo a link damian216-001-site1.ctempurl.com ingrese al aplicativo según indica el manual de usuario. Una vez dentro desarrollé tanto la prueba de vacío y rotor bloqueado, tomando datos de corriente, voltaje y potencia según corresponda.

En un informe de laboratorio detalle el proceso de ejecución y resultados de las pruebas, adjuntando imágenes del circutor, tenga en cuenta la información suministrada en la **Wiki** del aplicativo web para entender el funcionamiento de la planta.

Escriba en detalle el procedimiento seguido en cada una de las actividades realizadas en esta práctica y finalmente presente el circuito equivalente del motor de inducción bajo prueba.

Bibliografía

- [1] CEPAL - UNESCO. «La educación en tiempos de pandemia de COVID-19». En: (2020), pág. 1.
- [2] Jesús Aguilar y col. *Educación y pandemia. Una visión Académica*. First. Universidad Nacional Autónoma de México, 2020, págs. 10-13, 66. ISBN: 978-607-30-3220-9.
- [3] Jesús Wiliam Huanca-Arohuanca y col. «El problema social de la educación virtual universitaria en pandemia, Perú». En: *Innovaciones Educativas* (2020), págs. 125-126.
- [4] Gonzalo Aiassa Martínez y col. *Nuevas Prácticas de Enseñanza y Evaluación Virtual en Ingeniería. Innovando desde la experiencia en tiempos de pandemia*. Primera. Universidad FASTA Ediciones, 2020, págs. 30-41. ISBN: 978-987-1312-93-1.
- [5] D.C. Herrera, K. Triana y W. Mesa. «Importancia de los laboratorios remotos y virtuales en la educación superior». En: (2020), pág. 2.
- [6] Diana Herreo Villareal, Carlos Arguedas Matarrita y Evelyn Gutiérrez Soto. «Laboratorios remotos: recursos educativos para la experimentación a distancia en tiempos de pandemia desde la percepción de estudiantes». En: *Revista de Enseñanza de la Física* 32 (2020), pág. 182.
- [7] Francesc Pedró. «Covid-19 y educación superior en América latina y el Caribe: efectos, impactos y recomendaciones políticas». En: *Análisis Carolina* (2020), págs. 1-5.
- [8] Wilmar Yovany Rojas y Andrea Liliana Fagua. «Laboratorios remotos y Virtuales: una herramienta para el desarrollo de practicas en ingeniería». En: *Ciencia Tecnología e Investigación* (2013), págs. 72-74.
- [9] Ignacio Julio Idoyaga y col. «El Laboratorio Remoto: una alternativa para extender la actividad experimental». En: *Campo Universitario* (2020), págs. 1-6.
- [10] J. Machotka, A.Nafalski y Z. Nedié. «The history of developments of remote experiments». En: *2nd World Conference on Technology and Engineering Education* (2011), págs. 96-97.

-
- [11] Magalí Grisolio y col. «Laboratorios Remotos para Prácticas de Ingeniería en Carreras de Grado». En: *EST, Concurso de Trabajos Estudiantiles* (2018), págs. 56-61.
- [12] Dominik May y col. «The remote laboratory VISIR - Introducing online laboratory equipment in electrical engineering classes». En: *IEEE* (2020), págs. 1-4.
- [13] Martyn Cooper, Alexis Donnelly y José M. M. Ferreira. «Remote controlled experiments for teaching over the internet: a comparison of approaches developed in the pearl project». En: (2002), págs. 3-4.
- [14] Martyn Cooper y José M. M. Ferreira. «Remote Laboratories Extending Access to Science and Engineering Curricular». En: *IEEE* (2009), pág. 343.
- [15] José M. Ferreira y Dieter Mueller. «The MARVEL EU project: A social constructivist approach to remote experimentation». En: (2004), págs. 6-7.
- [16] V. Judson Harward y col. «The iLab Shared Architecture: A Web Services Infrastructure to Build Communities of Internet Accessible Laboratories». En: *IEEE* (2004), págs. 932-935.
- [17] Claudia Maria Gonzalez Aquino, Noel Antonio Ramos Ayala y Henry David Treminio. «Propuesta de laboratorio de pruebas de maquinas electricas». En: (2016), pág. 121.
- [18] John Anderson Galvis Niño y John Sebastian Bustamante Vargas. «Laboratorio remoto de motores eléctricos basado en una plataforma web y el software de procesos Labview para el acompañamiento de asignaturas virtuales en Tecnoparque». En: *Repositorio Universidad Piloto* (2016), pág. 93.
- [19] Jairo Antonio Contreras-Mendieta y col. «Implementación de un Laboratorio Remoto (LR), como recurso de apoyo en un sistema de Educación a Distancia». En: *Iberian Journal of Information Systems and Technologies* (2018), págs. 926-933.
- [20] TechTarget Contributor. *Web application (Web app)*. Disponible en [//searchsoftwarequality.techtarget.com/definition/Web-application-Web-app](https://searchsoftwarequality.techtarget.com/definition/Web-application-Web-app) (2019/08).
- [21] Vanessa Marisol Chenaz Paspuezán. «Desarrollo de una aplicación web basa en MVC para la gestión de solicitudes en la carrera de ingeniería en electrónica y redes de información». En: (2020), pág. 3.
- [22] Carmen Valera y col. «Estándares y legislación sobre accesibilidad web». En: *ATICA* (2012), págs. 47-48.
- [23] Raúl V. Lerma-Blasco, José Alfredo Murcia Andrés y Elvira Mifsud Talón. *Aplicaciones Web*. McGraw-Hill/Interamericana de España, S.L., 2013. ISBN: 978-84-481-8570-1.

-
- [24] Juan Diego Gauchat. *El gran libro de HTML5 CSS3 Y Javascript*. MARCOMBO, S.A. 2012, 2012. ISBN: 978-84-267-1782-5.
- [25] Jorge Ferrer, Victor García y Rodrigo García. *Curso completo de HTML*.
- [26] Jesús Arias Fisteus y Norberto Fernández García. «Introducción a HTML, XHTML y CSS». En: (), págs. 15-17.
- [27] Miguel Angel Alvarez y Manu Gutierrez. *Manual de Javascript*. 2016.
- [28] Betsabé Pastrana Gutiérrez, ATomás Ignacio Asiaín Olivares y Daniel Ruiz Vega. «Determinación de Parámetros de Motores de Inducción a Partir de Datos del Fabricante». En: *ATICA* (2018), págs. 3-4.
- [29] Anonimo. *Optidrive E2 - User Guide*.
- [30] Autores Varios. *Start Bootstrap - SB Admin 2*. Disponible en [//github.com/startbootstrap/startbootstrap-sb-admin-2](https://github.com/startbootstrap/startbootstrap-sb-admin-2) (2021/04).