



**UNIVERSIDAD TÉCNICA DE AMBATO**

**FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E  
INDUSTRIAL**

**CARRERA DE INGENIERÍA INDUSTRIAL EN PROCESOS DE  
AUTOMATIZACIÓN**

**Tema:**

---

**“PROTOTIPO DE UNA ESTACIÓN DE AUTO-LAVADO UTILIZANDO  
SISTEMAS DE CONTROL INTELIGENTES”**

---

Trabajo de Titulación Modalidad: Proyecto de Investigación, presentado previo a la obtención del título de Ingeniero Industrial en Procesos de Automatización.

**ÁREA:** Electrónica

**LÍNEA DE INVESTIGACIÓN:** Tecnologías de la Información y Sistemas de Control.

**AUTOR:** Dario Javier Quilumba Muso

**TUTOR:** Ing. Franklin Wilfrido Salazar Logroño, Mg.

**Ambato - Ecuador**

**Agosto - 2020**

## **APROBACIÓN DEL TUTOR**

En calidad de tutor del Trabajo de Titulación con el tema: “PROTOTIPO DE UNA ESTACIÓN DE AUTO-LAVADO UTILIZANDO SISTEMAS DE CONTROL INTELIGENTES”, desarrollado bajo la modalidad Proyecto de Investigación por el señor Dario Javier Quilumba Muso, estudiante de la Carrera de Ingeniería Industrial en Procesos de Automatización, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que el estudiante ha sido tutorado durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 15 del Reglamento para obtener el Título de Tercer Nivel, de Grado de la Universidad Técnica de Ambato, y el numeral 7.4 del respectivo instructivo.

Ambato, agosto 2020.

Digitally signed by FRANKLIN  
WILFRIDO SALAZAR LOGRONO  
Date: 2020.07.28 17:44:06 COT  
Reason: TUTOR DE TESIS

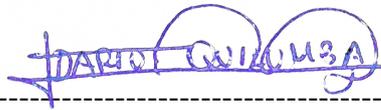
-----  
Ing. Franklin Wilfrido Salazar Logroño, Mg.

**TUTOR**

## AUTORÍA

El presente proyecto de investigación titulado: “PROTOTIPO DE UNA ESTACIÓN DE AUTO-LAVADO UTILIZANDO SISTEMAS DE CONTROL INTELIGENTES”, es absolutamente original, autentico y personal. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, agosto 2020.



Dario Javier Quilumba Muso

C.C. 0503315988

AUTOR

## APROBACIÓN TRIBUNAL DE GRADO

En calidad de par calificador del Informe Final del Trabajo de Titulación presentado por el señor Dario Javier Quilumba Muso, estudiante de la Carrera de Ingeniería Industrial en Procesos de Automatización, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado: “PROTOTIPO DE UNA ESTACIÓN DE AUTO-LAVADO UTILIZANDO SISTEMAS DE CONTROL INTELIGENTES”, nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 17 del Reglamento para obtener el Título de Tercer Nivel, de Grado de la Universidad Técnica de Ambato, y al numeral 7.6 del respectivo instructivo. Para cuya constancia suscribimos, conjuntamente con la señora Presidenta del Tribunal.

Ambato, agosto 2020.



Firmado electrónicamente por:  
**ELSA PILAR  
URRUTIA**

-----  
Ing. Pilar Urrutia, Mg.  
PRESIDENTA DEL TRIBUNAL



Firmado electrónicamente por:  
**SANTIAGO MAURICIO  
ALTAMIRANO  
MELENDEZ**

-----  
Ing. Santiago Altamirano, Mg.  
DOCENTE CALIFICADOR



Firmado electrónicamente por:  
**FRANKLIN  
GEOVANNY TIGRE  
ORTEGA**

-----  
Ing. Franklin Tigre, Mg.  
DOCENTE CALIFICADOR

## **DERECHOS DEL AUTOR**

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi Trabajo de Titulación en favor de la Universidad Técnica de Ambato, con fines de difusión pública. Además, autorizo su reproducción total o parcial dentro de las regulaciones de la institución.

Ambato, agosto 2020.



Dario Javier Quilumba Muso

C.C. 050315988

AUTOR

## **DEDICATORIA**

*Con todo mi corazón a mi madre Gladys, con su trabajo diario, apoyo y dedicación ha sido y será un pilar fundamental en mi vida para seguir cumpliendo mis objetivos y metas planteadas.*

*A toda mi familia en especial a mi abuelita que depositaron su confianza en mí, con sus consejos y apoyo incondicional en mi camino hacia la culminación de mi carrera.*

*También a ti compañera de mi vida Naty R., quien estuvo en los mejores momentos de mi vida, y por tu apoyo incondicional.*

*Darío Javier Quilumba Muso*

## AGRADECIMIENTO

*A la Universidad Técnica de Ambato y  
la Facultad de Ingeniería en Sistemas,  
Electrónica e Industrial.*

*A todos los docentes que impartieron  
sus conocimientos, anécdotas y  
experiencias para en un futuro seamos  
profesionales de bien.*

*A mi tutor el ing. Franklin Salazar  
quien me guio para la culminación de  
este trabajo de titulación.*

*También a todos mis compárenos de  
aulas, con quienes hemos  
compartimos momentos divertidos e  
inolvidable en todo el trascurso de la  
universidad.*

*Darío Javier Quilumba Muso*

## ÍNDICE GENERAL DE CONTENIDOS

### CONTENIDO

APROBACIÓN DEL TUTOR.....	ii
AUTORÍA.....	iii
APROBACIÓN DEL TRIBUNAL DE GRADO .....	iv
DERECHOS DEL AUTOR .....	v
DEDICATORIA .....	vi
AGRADECIMIENTO .....	vii
ÍNDICE GENERAL DE CONTENIDOS.....	viii
ÍNDICE DE FIGURAS.....	xii
ÍNDICE DE TABLAS .....	xiv
RESUMEN EJECUTIVO .....	xv
ABSTRACT.....	xvi
CAPÍTULO I.....	1
MARCO TEÓRICO.....	1
1.1. Antecedentes Investigativos.....	1
1.1.1. Contextualización del problema .....	1
1.1.2. Estudio del arte .....	2
1.1.3. Fundamentación teórica .....	4
Interfaz hombre -máquina HMI.....	4
Guía GEDIS .....	4
PLC (Controlador Lógico programable).....	5
Tarjetas electrónicas de propósito general.....	6
Raspberry Pi.....	7
Servidor web .....	8
Arquitectura del servidor web.....	8
Esquema del servidor web .....	8
Servidor HTTP Apache .....	9
Base de datos .....	10
MySQL .....	10
1.2. Objetivos .....	11

CAPÍTULO II .....	12
METODOLOGÍA .....	12
2.2. Métodos.....	12
2.2.1. Modalidad de la Investigación.....	12
De campo .....	13
Bibliográfica .....	13
Aplicada.....	13
2.2.2. Población y Muestra (en caso de requerir) .....	13
2.2.3. Recolección de Información .....	13
2.2.4. Procesamiento y Análisis de Datos.....	13
2.2.5. Desarrollo del Proyecto.....	14
CAPÍTULO III.....	15
RESULTADOS Y DISCUSIÓN .....	15
3.1. Etapas del proceso de lavado de vehículos .....	15
3.2. Sistemas de lavado .....	16
3.2.1. Puente de lavado .....	16
3.2.2. Túnel de lavado.....	16
3.2.3. Auto-lavado de coches .....	17
3.3. Sistemas de auto-lavado automático.....	17
3.3.1. Clean-Car .....	17
3.3.2. Kondor .....	18
3.3.3. Saber S11 .....	18
3.4. Sistemas de cobro automáticos .....	19
3.4.1. CashDro .....	19
3.4.2. ERP de Telemetal .....	19
3.4.3. Burget cajero automático .....	20
3.5. Selección de equipos.....	21
3.5.1. Selección del autómeta programable .....	21
3.5.2. Selección de SBD para servidor web.....	21
3.5.3. Selección de sistema HMI .....	22
3.6. Diagrama de flujo del funcionamiento del prototipo .....	23
3.7. Prototipo de la estación de Auto-lavado .....	24
3.8. Diagrama de conexión del prototipo.....	26

3.8.1. Diagrama de conexión de entradas del PLC .....	26
3.8.2. Diagrama de conexión de las salidas del PLC .....	26
3.9. Programación del PLC .....	28
3.9.1. Software TIA Portal .....	28
3.9.2. Variables utilizadas en el PLC .....	28
3.9.3. Programación para lavado simple .....	29
3.9.4. Programación para lavado doble .....	30
3.10. Desarrollo de las pantallas HMI .....	32
3.10.1. Consideraciones en el desarrollo de las HMI .....	32
3.10.2. Conexión entre PLC y HMI .....	34
3.10.3. Pantalla principal .....	34
3.10.4. Pantalla para el funcionamiento de la bomba .....	36
3.10.5. Pantalla de tiempo real .....	38
3.10.6. Pantalla de etapas de lavado .....	39
3.11. Plataforma de arquitectura del prototipo .....	41
3.12. Diseño del mensaje de alarma por SMS a teléfono móvil .....	41
3.12.1. Plataforma Twilio .....	42
3.12.2. Creación de cuenta en plataforma Twilio .....	42
3.12.3. Programación en Node-Red para envío de mensaje .....	43
3.12.4. Programación para obtener datos del PLC .....	43
3.12.5. Programación para envío de mensaje .....	44
3.12.6. Recepción del mensaje de alerta .....	46
3.12.7. Página web con Node-Red .....	46
3.13. Diseño del pago automático .....	47
3.13.1. Plataforma PayPal .....	47
3.13.2. Creación de cuenta en PayPal .....	48
3.13.3. Creación del botón de pago en PayPal .....	48
3.13.4. Prueba del botón de pago en PayPal con PHP .....	51
3.14. Comunicación del PLC S7-1200 con la Raspberry Pi .....	52
3.15. Programación de las páginas web con PHP .....	53
3.15.1. Escritura de datos del PLC mediante página web .....	55
3.15.2. Lectura de datos del PLC mediante página web .....	56
3.16. Diseño de la página web para control y adquisición de datos .....	56

3.16.1. Página para el registro de usuarios .....	57
3.16.2. Página de bienvenida y pago automático .....	57
3.16.3. Página de control del proceso de Auto-lavado .....	58
3.16.4. Página de monitorización de variables .....	58
3.17. Análisis y discusión de resultados.....	59
CAPÍTULO IV.....	63
CONCLUSIONES Y RECOMENDACIONES.....	63
4.1. Conclusiones .....	63
4.2. Recomendaciones.....	64
BIBLIOGRAFÍA .....	65
ANEXOS .....	69

## ÍNDICE DE FIGURAS

<b>Figura 1:</b> Esquema servidor web. ....	9
<b>Figura 2:</b> Diagrama de flujo.....	23
<b>Figura 3:</b> Tablero de control del prototipo de la estación.....	24
<b>Figura 4:</b> Maqueta del prototipo de la estación.....	25
<b>Figura 5:</b> Diagrama de conexión de las entradas físicas del PLC .....	26
<b>Figura 6:</b> Diagrama de conexión de las salidas del PLC .....	27
<b>Figura 7:</b> Diagrama de conexión de las salidas del PLC .....	27
<b>Figura 8:</b> Programación segmento 1 .....	29
<b>Figura 9:</b> Programación segmento 2 .....	29
<b>Figura 10:</b> Programación segmento 2 .....	30
<b>Figura 11:</b> Programación segmento 3 .....	30
<b>Figura 12:</b> Programación segmento 4 .....	31
<b>Figura 13:</b> Programación segmento 5 .....	31
<b>Figura 14:</b> Programación segmento 6 .....	32
<b>Figura 15:</b> Conexión de la interfaz .....	34
<b>Figura 16:</b> Diseño de la pantalla principal .....	34
<b>Figura 17:</b> Pantalla principal HMI.....	35
<b>Figura 18:</b> Programación segmento pantalla principal .....	36
<b>Figura 19:</b> Diseño de la pantalla de la bomba de agua .....	36
<b>Figura 20:</b> Pantalla tiempo de funcionamiento de la bomba de agua .....	37
<b>Figura 21:</b> Programación del tiempo de apagado de la bomba.....	37
<b>Figura 22:</b> Diseño de la pantalla de tiempo real .....	38
<b>Figura 23:</b> Pantalla de tiempo real .....	38
<b>Figura 24:</b> Diseño de la pantalla etapas de lavado.....	39
<b>Figura 25:</b> Pantalla etapas de lavado.....	40
<b>Figura 26:</b> Programación para visualizar el contador .....	40
<b>Figura 27:</b> Plataforma de arquitectura .....	41
<b>Figura 28:</b> Número de teléfono liberado en la plataforma Twilio .....	42
<b>Figura 29:</b> Código base de programación de mensaje .....	43
<b>Figura 30:</b> Programación de PLC en Node-Red .....	44
<b>Figura 31:</b> Estructura final del programa para enviar de mensaje .....	44
<b>Figura 32:</b> Programación bloque de función.....	45

<b>Figura 33:</b> Programación bloque de mensaje.....	45
<b>Figura 34:</b> Recepción de mensaje .....	46
<b>Figura 35:</b> Página web con Node-Red .....	47
<b>Figura 36:</b> Página de usuario creado en PayPal .....	48
<b>Figura 37:</b> Crear botón de pago en PayPal .....	49
<b>Figura 38:</b> Crear botón de pago en PayPal .....	49
<b>Figura 39:</b> Personalización de botón de pago en PayPal .....	50
<b>Figura 40:</b> Código HTML de botón de pago en PayPal .....	51
<b>Figura 41:</b> Página web de pago automático en PayPal .....	51
<b>Figura 42:</b> Pago del servicio de lavado en PayPal.....	52
<b>Figura 43:</b> Código Python para escritura de datos.....	53
<b>Figura 44:</b> Código Python para lectura de datos.....	53
<b>Figura 45:</b> Código para guardar el registro en base de datos.....	54
<b>Figura 46:</b> Código de conexión con la base de datos.....	54
<b>Figura 47:</b> Código de acceso al sistema de Auto-lavado .....	55
<b>Figura 48:</b> Importar scripts de Python en PHP .....	55
<b>Figura 49:</b> Actualizar datos con JQuery .....	56
<b>Figura 50:</b> Tempo de actualización de datos con JQuery .....	56
<b>Figura 51:</b> Página web para acceder al servicio.....	57
<b>Figura 52:</b> Página web de registro .....	57
<b>Figura 53:</b> Página web de bienvenida y pago automático .....	58
<b>Figura 54:</b> Página web para el control del auto-lavado .....	58
<b>Figura 55:</b> Página web de monitorización .....	59
<b>Figura 56:</b> Tiempo de arranque de escritura de datos.....	60
<b>Figura 57:</b> Tiempo de arranque de lectura de datos.....	61
<b>Figura 58:</b> Consumo de la CPU en la Raspberry Pi.....	62
<b>Figura 59:</b> Grafica consumo de la CPU en la Raspberry Pi.....	62

## ÍNDICE DE TABLAS

<b>Tabla 1:</b> Metodología Guía GEDIS. ....	5
<b>Tabla 2:</b> Tarjetas electrónicas. ....	6
<b>Tabla 3:</b> Etapas de lavado .....	15
<b>Tabla 4:</b> Selección de PLC.....	21
<b>Tabla 5:</b> Selección de SBD .....	22
<b>Tabla 6:</b> Selección de HMI .....	22
<b>Tabla 7:</b> Declaración de variables.....	28
<b>Tabla 8:</b> Selección de tipo y tamaño de fuente según la Norma GEDIS. ....	32
<b>Tabla 9:</b> Selección de colores según la Norma GEDIS. ....	33

## RESUMEN EJECUTIVO

El objetivo de la investigación fue desarrollar, implementar y proporcionar una herramienta adecuada e innovadora de pago automático para un prototipo de una estación de Auto-lavado, centrados en nuevas tecnologías aplicadas a los modelos de negocio, cuenta con páginas web, para el acceso al servicio por parte de los usuarios de los vehículos, también con un sistema de aviso de emergencia por medio de un mensaje SMS de alerta, el pago se realiza mediante un sistema de pago en línea y la monitorización del proceso y estado de los equipos se utilizó la misma plataforma web y una pantalla HMI. La investigación se centra en la aplicación de tecnología vanguardista en la industria sectorial y nacional con eficiencia y calidad.

Para el desarrollo del sistema planteado se utilizó un PLC s7-1200 1214, además una pantalla HMI KPT 600, que fueron programados con el software SIMATIC STEP7 (TIA portal V13). Para la escritura y lectura de datos desde el PLC hacia las páginas web, se utiliza el lenguaje de programación Python y se comunica con la Raspberry Pi mediante el protocolo Ethernet S7, con un tiempo promedio de 2.16 segundos en la escritura de datos y en la lectura es de 860 milisegundos, desde las páginas web creadas hasta el PLC que controla el prototipo de la estación de Auto-lavado.

El usuario tiene la facilidad de seleccionar entre dos tipos de lavado el simple y doble y posteriormente liquida el servicio, con su tarjeta de crédito en la plataforma de pagos en línea PayPal, y por último para él envío del mensaje de alerta se utiliza la plataforma de comunicaciones y mensajería Twilio.

**Palabras clave:** Pago Automático, SMS de Alerta, PLC, HMI, Python, Raspberry Pi

## **ABSTRACT**

The objective of the research was to develop, implement and provide an adequate and innovative automatic payment tool for a prototype of a self-cleaning station, focused on new technologies applied to business models, has web pages, for access to service by the users of the vehicles, also with an emergency warning system by means of an SMS alert message, the payment is made through an online payment system and the monitoring of the process and condition of the equipment was used the same web platform and an HMI screen. The research focuses on the application of cutting-edge technology in the sector and national industry with efficiency and quality.

For the development of the proposed system, an s7-1200 1214 PLC was used, in addition a KMI 600 HMI screen was programmed with the SIMATIC STEP7 software (TIA portal V13) .For writing and reading data from the PLC to the web pages, The Python programming language is used and communicates with the Raspberry Pi via the Ethernet S7 protocol, with an average time of 2.16 seconds in writing data and reading 860 milliseconds, from the web pages created to the PLC that controls the prototype of the Auto-wash station.

The user has the facility to select between two types of washing the single and double and subsequently liquidates the service, with his credit card on the PayPal online payment platform, and finally the platform is used to send the alert message of communications and messaging Twilio.

**Keywords:** Automatic Payment, Alert SMS, PLC, HMI, Python, Raspberry Pi

# **CAPÍTULO I**

## **MARCO TEÓRICO**

### **1.1. Antecedentes Investigativos**

#### **1.1.1. Contextualización del problema**

En el entorno actual de competencia mundial, desarrollo científico, tecnológico e innovación, las empresas en especial de manufactura, se ven forzadas a reconfigurar sus procesos. La Industria 4.0, la automatización de procesos industriales, los instrumentos y la manufactura inteligente son parte de esta transformación, en la que tecnologías de producción e información se han integrado para mejorar los procesos creando innovadores sistemas de manufactura, gestión que permiten el aumento de la productividad y mejora de la calidad, alcanzando una mayor flexibilidad, eficiencia y así generar una propuesta de valor agregado para sus clientes [1].

Los sistemas SCADA consisten en gobernar, controlar y supervisar los procesos sin la intervención continua de un operador humano. Para ello se utilizan un conjunto de aplicaciones de software especialmente diseñadas para funcionar sobre ordenadores de control de producción, con acceso a la planta mediante la comunicación por una red ethernet con los demás equipos, tales como plc's, variadores, instrumentos de medición y actuadores [2]. A nivel mundial las empresas cuentan con Sistemas SCADA y HMI para la automatización de maquinarias, plantas industriales, los cuales se encargan de controlar, supervisar y monitorear los procesos de manera visual, desde los procesos simples hasta los más complejos, obteniendo datos estadísticos para tomar decisiones y perfeccionar la producción. Los HMI's cada vez se han ido masificando, por la necesidad de tener el control más preciso de la producción, además de contar con indicadores de rendimiento de procesos en tiempo real [3] [4].

Las empresas nacionales de Ecuador en la actualidad cuentan con procesos de automatización los cuales se han ido desarrollando de acuerdo al avance tecnológico, consiguiendo un cambio en el modelo productivo, basado en el internet de las cosas [5]. Permitiendo que los sistemas de producción tengan un mayor aprovechamiento de los recursos naturales, para fábricas como el acero, el cable, el papel, evitando el

desperdicio de materia prima, los costes de fabricación, y mejorando la calidad y continuidad de los procesos para la elaboración de los productos, liberando al ser humano de tareas monótonas y peligrosas [6].

Los nuevos mercados se establecen en la personalización y la creación de nuevos productos y servicios innovadores. Exigiendo calidad en los productos, pagando por la experiencia o el servicio más que por el producto. Cualidades que ésta cuarta revolución industrial provoca debido al impacto de las tecnologías digitales aplicadas a los modelos de negocio de la industria.

De acuerdo a lo mencionado anteriormente, y con el afán de proporcionar una herramienta adecuada e innovadora con la cual los usuarios en este caso propietarios de los vehículos podrán acceder de manera automática al servicio de lavado de su auto, por medio de una página web amigable y fácil de utilizar, también contara con un servicio de aviso de emergencia mediante un mensaje SMS de alerta, que será enviado hasta el teléfono celular además, para el pago por el servicio brindado se utiliza un sistema de pago en línea y la monitorización del proceso y estado de los equipos se lo realizará en tiempo real, se utiliza la misma plataforma web y una pantalla HMI, con lo cual se podrá tomar decisiones en caso de existir alguna falla dentro del sistema. La investigación se centra en la aplicación de tecnología vanguardista en la industria sectorial y nacional con eficiencia y calidad.

### **1.1.2. Estudio del arte**

Dado que el propósito del presente trabajo de investigación se centra en la automatización de un prototipo automático de auto-lavado, es muy importante tener un sistema de control con alto nivel de calidad y seguridad y hacerlos encajar en el proceso productivo existente dentro del mercado de lavado de autos.

Basado en estas premisas se analizaron trabajos que involucran la automatización, control y monitorización.

En el trabajo de Posgrado y Educación continua de la Escuela Politécnica Superior de Chimborazo titulada como: “Desarrollo e implementación de internet industrial de las cosas aplicada al laboratorio de PLC’S de la Facultad de Ingeniería Industrial en Procesos de Automatización de la Universidad Técnica de Ambato”, publicada en el

año 2018 por la Mg, Tanny Gallo menciona que la programación que se realiza para la conexión entre el PLC Siemens 1200 y la Raspberry Pi fue mediante el uso del software PYTHON, donde se utiliza la librería SNAP7 que permite adquirir los datos del PLC los cuales son enviados utilizando el protocolo del IIoT denominado MQTT, luego se utiliza el software PhpMyAdmin para crear una base de datos donde se almacena los estados y datos, por último el software PHP que mediante páginas web permite presentar los datos y estados del PLC, para monitorear de manera remota desde cualquier dispositivo inteligente que se conecte a la red ethernet o wifi [7].

En el trabajo de investigación titulada como: “Sistema SCADA para la supervisión en tiempo real de medidores industriales de energía en la Empresa Novacero S.A.”, publicada en el año 2019 por el ingeniero Manuel Molina, entre los resultados obtenidos el sistema SCADA ofrece una perspectiva integrada de la información de los datos adquiridos por los elementos de medición y todos los recursos unificados en la implementación del sistema. De tal manera, los supervisores y operadores pueden observar e interactuar con los procesos de supervisión mediante las representaciones graficas de cada proceso integrado en la HMI. Respetando las normas que rigen a este tipo de proyecto para una representación adecuada en tonalidades en alertas tempranas y un entendimiento de recursos integrados al sistema de supervisión [8].

En el trabajo de investigación, Master en Ingeniera Industrial de la Universidad Politécnica de Valencia titulada como: “Proyecto de integración de autómatas programables Simatic s7-1200 en red Iot.” publicada por el Mg, Roberto Villena. Menciona que la idea del proyecto expuesto es un modo diferente y más económico de implementar la industrial 4.0 que cada día se extiende más, se utilizan protocolos de comunicación estándar que permiten una vinculación a mayor escala entre sistemas y mantener una comunicación más general, insertando comunicaciones vía redes sociales o email. Esta solución propuesta asocia la robustez de un sistema de control industrial, basado en Siemens lenguaje de programación TIA Portal, y la flexibilidad de un entorno de programación avanzado (Linux en Raspberry-Pi) y sencillo de implementar y mantener (Node-Red). Además, para permitir unificar todos los elementos como uno solo, es necesario utilizar el software de la multiplataforma “Snap 7”, que permite mantener una conexión y comunicación entre autómatas [9].

El sistema propuesto en comparación a los trabajos de investigación ya citados con anterioridad, se añade un sistema de pago automático en línea, además de contar con un servicio de avisos de emergencia, vía mensaje SMS hacia un número de celular. Con lo que crearemos sistemas de control basados en la seguridad y excelencia para el prototipo de auto-lavado.

### **1.1.3. Fundamentación teórica**

La implementación de un sistema automatizado y de monitorización abarca la temática que se detalla a continuación.

#### **Interfaz hombre -máquina HMI**

HMI es una interfaz que permite que el usuario o el operador del sistema de control o supervisión, interactúe con los procesos. Del acrónimo “Human Machine Interface”, es la herramienta esencial para la comunicación entre el operario y la máquina, facilitando el trabajo al operador para manipular, controlar, procesar y almacenar información útil y lógica mediante una interfaz intuitiva. Las señales de los procesos son conducidas por dispositivos de entrada/salida en la computadora [10].

#### **Funciones de un Sistema HMI**

- Supervisión del proceso productivo.
- Control de calidad de la producción.
- Control de la productividad.
- Mensajes de proceso, alarmas, averías, fallos y curvas de tendencias.
- Programación de tareas y paradas.
- Integración con máquinas para avisos de errores.
- Posibilidad de actuar desde la pantalla [10].

#### **Guía GEDIS**

La guía ergonómica de diseño de interfaz de supervisión GEDIS ofrece un método de diseño especializado en sistemas de control, supervisión industrial basado en niveles donde se van concretando los diseños de los distintos tipos de pantalla y contenidos [10] [11].

Es un conjunto de metodologías que tiene como fin el desarrollo integral de una interfaz hombre-máquina HMI.

**Tabla 1:** Metodología Guía GEDIS [10].

<b>Nº-</b>	<b>Elementos de la interfaz</b>	<b>Definición</b>
<b>1</b>	Arquitectura	Organización jerárquica de pantalla
<b>2</b>	Distribución de pantallas	Plantillas de los diferentes tipos de pantallas
<b>3</b>	Navegación	Modos de navegación entre pantallas
<b>4</b>	Uso de color	Asociación del color según la funcionalidad del proceso
<b>5</b>	Uso de fuentes e información textual	Tipos de tamaños y fuentes de texto
<b>6</b>	Estatus de los equipos y eventos del proceso	Uso de símbolos e iconos para representar el estado de la planta así como los cambios en la misma
<b>7</b>	Información y valores de proceso	Presentación de datos analógicos y digitales en la interfaz
<b>8</b>	Gráfico de tendencia y tablas	Presentación de valores históricos en gráficos y/o tablas
<b>9</b>	Comandos y entradas de datos	Modo de ingreso de datos en la interfaz
<b>10</b>	Alarmas	Características principales del sistema de alarmas

### **PLC (Controlador Lógico programable)**

Los PLC son equipos electrónicos y programables de acuerdo a algún lenguaje de programación, muy utilizados en muchas industrias y máquinas, que pueden ser programados por los usuarios y están diseñados para automatizar procesos industriales electromecánicos que pueden contar con múltiples señales: entrada/salida, analógicas y digitales con amplios rangos de temperatura, presentan inmunidad al ruido eléctrico y buena resistencia a la vibración y al impacto (robustez) [12].

### **Siemens s7 1200**

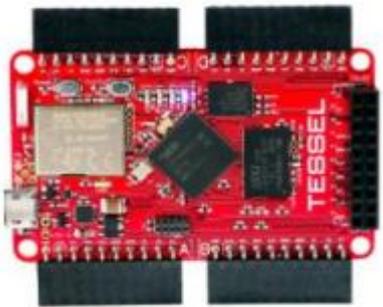
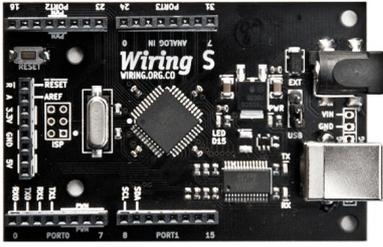
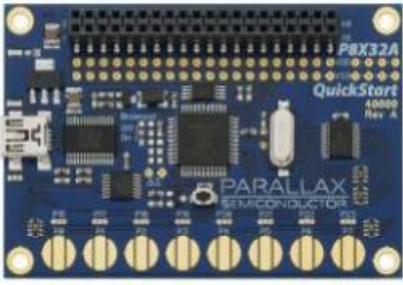
Es un Controlador Lógico Programable (PLC), modular compacto de la compañía SIEMENS, ofrece la flexibilidad y capacidad de controlar una gran variedad de dispositivos para las distintas necesidades de automatización, además se puede conectar con un HMI para el control de procesos. Gracias a su diseño compacto, configuración flexible y amplio juego de instrucciones, el s7- 1200 es idóneo para controlar una gran variedad de aplicaciones [1].

El PLC a utilizar en el proyecto es un SIMATIC S7 1200 CPU 1214C AC/DC/Rly, con alimentación de 85-264 VAC, 47-63 Hz, 300 mA. Posee 7 entradas digitales de 24 VDC a 6mA/pt, 7 salidas digitales tipo Rly de 30VDC/250VAC a 2A/pt.

### Tarjetas electrónicas de propósito general

Son aquellos dispositivos con la capacidad de recibir información, realizar cálculos y dar una respuesta como salida con el fin de resolver un sistema. A continuación, describiremos algunos de ellos:

**Tabla 2:** Tarjetas electrónicas.

Nombre	Imagen	Descripción
Tessel		<p>Esta tarjeta de desarrollo maneja el lenguaje de programación JAVASCRIPT compatible con el API de NODE.JS. Es una tarjeta que se programa de manera familiar para desarrolladores de software [13].</p>
Wiring		<p>El entorno de programación de es compatible con cualquier hardware basado en la serie de procesadores AVR de Atmel, no sólo las propias tarjetas Wiring. Y esto incluye la línea de Arduino [13].</p>
Parallax Propeller		<p>Esta placa tiene un procesador multi-núcleo paralelo arquitectura de ordenador microcontrolador de chip con ocho de 32 bits denominado Parallax Propeller, esta placa puede ejecutar aplicaciones en paralelo en lugar de utilizar interrupciones tiene la forma similar a un Arduino [13].</p>

Jaguar One		El JaguarBoard nació recientemente de un proyecto de Kickstarter y ya tiene su primer modelo, Jaguar One. Su mayor atractivo es la compatibilidad con X86 y la posibilidad de instalar Windows 8 o 10 en su versión completa [13].
------------	---	--

## Raspberry Pi

Muy conocido como un ordenador de placa única o reducida (SBC) de bajo costo o como un mini ordenador, diseñada originalmente para la educación y hasta el momento es utilizada en ámbitos como la robótica y domótica, inspirada en la BBC Micro de 1981. El objetivo del creador era crear un dispositivo que mejorara las habilidades de programación en especial del lenguaje Python y la comprensión del hardware, expresamente no se define como hardware libre sin embargo al ser de código abierto (Open Source) cualquiera puede aportar haciendo modificaciones en la programación del software original, la tarjeta permite el uso de varios sistemas operativos GNU/Linux como Ubuntu, Raspbian e incluso Windows [13] [14] [15].

La Raspberry Pi 3 Model B forma parte de la tercera generación de estos modelos y mantiene el mismo formato de placa popular que el Raspberry Pi , pero cuenta con un SoC de 64 Bits, con una velocidad de 1,2 GHz, además de Wi-Fi de doble banda y Bluetooth incorporado [15].

## Características

- Consta de una ranura SD donde debe ser instalado el sistema operativo.
- Memoria RAM fija.
- Posee varios puertos USB expandibles.
- Es compatible con módulos WIFI para su conexión inalámbrica con la red del router.
- No cuenta con un reloj de tiempo real.
- Consta con un puerto Ethernet para su conexión.
- Posee una salida HDMI para conectarla a un monitor.
- Es considerado como un dispositivo de bajo costo [15].

## **Servidor web**

Son programas de uso cotidiano en internet que implementa el protocolo HTTP (hypertext transfer protocol), realizando conexiones bidireccionales o unidireccionales. Este protocolo está diseñado para transferir lo que llamamos hipertextos, páginas web o páginas HTML (hypertext markup language): textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de sonidos [16].

## **Arquitectura del servidor web**

La arquitectura utilizada es la de cliente/servidor, es decir, el equipo (el cliente) hace una solicitud o petición al equipo servidor, y éste atiende la solicitud.

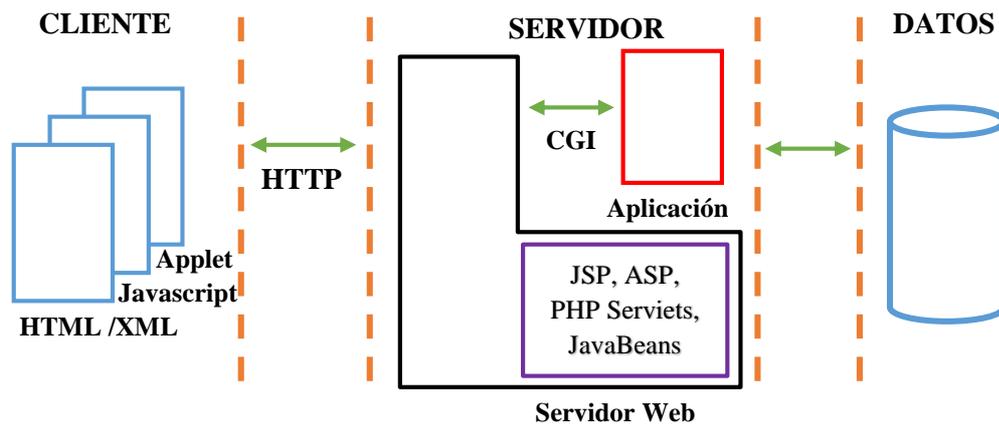
En el equipo cliente se ejecuta una aplicación llamada navegador o cliente web que:

- Sirve de interfaz con el usuario: atiende sus peticiones, muestra los resultados de las consultas y proporciona al usuario un conjunto de herramientas que facilitan su comunicación con el servidor.
- Se comunica con el servidor web: transmite las peticiones de los usuarios [16].

## **Esquema del servidor web**

La figura 1, se indica el esquema de funcionamiento de un servidor web, en esta se observa los tres elementos básicos que componen las peticiones web dinámicas:

- El cliente que hace la petición.
- El servidor que atiende la petición.
- Y los datos solicitados mediante la petición.



**Figura 1:** Esquema servidor web [16].

Para que cualquier navegador pueda llegar al servidor web y atender la petición (ya sea en una red local o en Internet) el servidor debe tener asignada una dirección IP estática (Internet Protocol) única que identifica el equipo en la red [16].

Cuando el navegador realiza una petición de página al servidor web utiliza un nombre único que identifica al servidor y esta solicitud se transmite a un servidor DNS. Éste resuelve el nombre y devuelve la dirección IP que corresponde al nombre [16].

### **Servidor HTTP Apache**

Es uno de los servidores más usados en el planeta ya que cuenta con una arquitectura modular (open source), en donde el servidor cuenta con una parte de core y diferentes módulos los cuales son altamente configurables [17].

El servidor va a permitir que las aplicaciones tanto de la estación como las móviles se comuniquen con la base de datos, este va a estar a la espera de peticiones de los usuarios o administradores/operadores para acceder a la base de datos. Además, en su configuración se encuentran los scripts escritos en PHP, los cuales van a ser necesarios para que la aplicación móvil pueda realizar las consultas a la base de datos [17].

### **Características**

- **Estable:** consecuencia de su probada robustez que impide caídas o cambios en el servidor inesperados.
- **Flexible:** es capaz de trabajar con el estándar HTTP/1.1 (RFC2616) y con la mayor parte de las extensiones web existentes en la actualidad, como son los módulos PHP, SSL, CGI.

- Extensible: dispone de gran cantidad de librerías que amplían su funcionalidad.
- Multiplataforma ya que está disponible para diferentes plataformas como GNU/Linux, Windows, MacOS [16] [17].

### **Base de datos**

Es un conjunto de elementos interrelacionados, organizados en cuyo interior guardan información de documentos, textos indexados o información de algún proceso. Las bases de datos pueden ser físicos o electrónicos, siendo la primera administrada manualmente por una persona. Las bases de datos electrónicas están gobernadas por un software como por ejemplo MySQL, Microsoft Access. Son creadas mediante tablas que identifican los elementos de una base de datos como usuarios y administradores, y estos permiten luego acceder a la información guardada [18].

### **MySQL**

Es un software de código abierto que sirve para administrar bases de datos de manera sencilla y rápida. Permite acceder a las bases de datos desde navegadores web, lo que lo hace ideal para transacciones de información online y demás soluciones que impliquen el almacenamiento de datos. Con MySQL se puede realizar múltiples tareas como consultas rápidas, creación de nuevas bases de datos, modificación de las mismas, entre otros [18].

### **Lenguaje de programación PHP, HTML**

- HTML es un lenguaje de programación que permite crear y diseñar páginas web utilizando comandos, marcas o etiquetas que permiten definir los atributos del mismo como: color del texto, hipervínculos, contenidos multimedia, etc.
- PHP fue inicialmente creado para entornos UNIX y es sobre este sistema operativo que se puede aprovechar de mejor manera sus funcionalidades y características. A diferencia de HTML, PHP es un lenguaje que resulta más sencillo de aprender e interpretar las líneas de código programado.

Para programar en PHP es necesario tener instalado un servidor web, por ejemplo, Apache en la Raspberry Pi o cualquier otro, trabaja bajo cualquier plataforma, Linux, Unix, Microsoft, etc. También es tener instalado un intérprete de PHP ya que el documento PHP pasa por el intérprete antes de ser mostrado al usuario [18].

## 1.2. Objetivos

- Desarrollar un sistema de control inteligente mediante un servidor web y pantallas HMI para un prototipo de una estación automática de auto-lavado.

El objetivo principal del presente proyecto es crear un prototipo de auto-lavado novedoso y tecnológico, que cuenta con un sistema de pago automático por medio de un código que es enviado hacia un teléfono móvil, además de contar con un sistema de monitorización en tiempo real por medio de una plataforma web y de una HMI con lo cual se podrá tomar acciones en caso de existir algún fallo en el sistema, también cuenta con un sistema de alertas de emergencia por medio de un mensaje SMS, para lo cual se han realizado las siguientes actividades:

1. Buscar información sobre el funcionamiento de sistemas de auto-lavado y cobros automáticos.
2. Seleccionar equipos y dispositivos para el funcionamiento del prototipo.
3. Programar la estación de auto-lavado.
4. Diseñar y programar las pantallas HMI.
5. Crear un servidor web para enviar código de alerta a teléfono móvil.
6. Crear un servidor web para el pago automático.
7. Diseñar la estación de auto-lavado con cobro automático.
8. Crear página para el control y monitoreo de datos.
9. Pruebas de funcionamiento.

## **CAPÍTULO II**

### **METODOLOGÍA**

#### **2.1. Materiales**

Los materiales y softwares a utilizar para la programación, control y monitorización de la estación de auto-lavado son:

- PLC S7-1200
- Pantallas KTP 600
- Software TIA Portal V13
- Prototipo de la estación de auto lavado

Para crear la página web, el pago automático y el aviso de emergencia se utilizan los siguientes materiales:

- Raspberry Pi 3 B+
- Software Rasbian
- Software Python
- Node-Red
- Lenguaje de programación HTML
- Leguaje de programación PHP
- Teléfono celular

Para la comunicación entre el PLC y la Raspberry Pi se utilizada la librería:

- Snap 7
- Protocolo Ethernet S7

#### **2.2. Métodos**

##### **2.2.1. Modalidad de la Investigación**

En el desarrollo del presente proyecto se utilizan las siguientes modalidades de investigación que se describen a continuación.

## **De campo**

El proyecto se realizó en los laboratorios de la facultad de Ingeniería en Sistemas, Electrónica e Industrial de la Universidad Técnica de Ambato, utilizando un prototipo de auto-lavado existente en el club de robótica.

## **Bibliográfica**

La sustentación de parte teórica de la investigación se realiza mediante la búsqueda de información en: libros, revistas tecnológicas, paper, trabajos de titulación, manuales, catálogos, así también de fuentes virtuales válidas. Que ayudaran con el cumplimiento de los objetivos del presente trabajo de titulación.

## **Aplicada**

La presente investigación es de característica aplicada ya que se implementa el sistema de control para la estación de auto-lavado y se realiza pruebas de funcionamiento del pago automático, mensaje de alerta y de los sistemas de monitorización tanto en plataforma web y en la interfaz hombre máquina HMI.

### **2.2.2. Población y Muestra (en caso de requerir)**

La presente investigación al tener una modalidad de investigación aplicada y el desarrollo se basa en crear un sistema inteligente, no requiere datos de población y muestra.

### **2.2.3. Recolección de Información**

Para el presente proyecto la información recopilada será en base a fuentes confiables de investigación: tesis, revistas, artículos científicos, manuales, libros de automatización, páginas de internet, etc. Los mismo que aporten de conocimientos para el desarrollo de la investigación.

### **2.2.4. Procesamiento y Análisis de Datos**

Una vez recopilada toda la información, el procesamiento será efectuado siguiendo los siguientes pasos:

- a) Revisión de la información
- b) Transcripción de información relevante
- c) Selección de equipos

- d) Generar bloques de programación
- e) Pruebas de funcionamiento.
- f) Corrección de errores
- g) Interpretación
- h) Presentación de resultados

### **2.2.5. Desarrollo del Proyecto**

La presente investigación tuvo la finalidad de implementar un sistema de pago automático en línea, mediante una plataforma web, para dar funcionamiento al prototipo de auto-lavado, además de contar un SMS de alarma que será enviado al teléfono celular del cliente, también cuenta con un sistema HMI y pagina web para su monitorización, esto se realizaría con los equipos disponibles en los laboratorios de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial de la Universidad Técnica de Ambato.

Las actividades que se llevaron a cabo para el desarrollo del proyecto son las siguientes:

- Buscar información sobre el funcionamiento de sistemas de auto-lavado y cobros automáticos.
- Seleccionar equipos y dispositivos para el funcionamiento del prototipo.
- Programar la estación de auto-lavado.
- Pruebas de funcionamiento.
- Diseñar la interfaz HMI.
- Programar las pantallas HMI.
- Crear un servidor web para enviar mensaje de alerta a teléfono móvil.
- Verificar la recepción de mensaje al teléfono móvil.
- Crear un servidor web para el pago automático.
- Implementar la estación de auto-lavado con cobro automático.
- Comunicar PLC con la Raspberry Pi.
- Crear página web para control y monitoreo de datos.
- Pruebas de funcionamiento.
- Elaborar el informe final.

## CAPÍTULO III

### RESULTADOS Y DISCUSIÓN

En el presente capítulo se detalla el proceso con el que se diseñó el prototipo de la estación de auto-lavado utilizando sistemas de control inteligente.

#### 3.1. Etapas del proceso de lavado de vehículos

**Tabla 3:** Etapas de lavado

N-º	Etapa	Definición
1	Prelavado	Durante el prelavado, el vehículo es rociado con productos desincrustantes. Son destinados a deshacer la suciedad más incrustada de la superficie del auto, como pueden ser mosquitos, defecaciones de aves, etc., en las siguientes etapas serán eliminados con la ayuda mecánica de los cepillos o del lavado de alta presión [19].
2	Lavado	El vehículo es rociado con agua y una solución de champú a través de los pulverizadores o rociadores. Su función, además de limpiadora, es suavizar la superficie de la pintura, maximizando la acción mecánica de los cepillos y previniendo las raspaduras. A menudo, también se aplica otra solución de producto detergente (conocido como espuma activa) que tiene por finalidad crear una espuma densa que cubra completamente la pintura. La espuma asegura la acción penetrante y limpiadora extrayendo poco a poco la suciedad que queda en la superficie [19].

3	Acabado	El acabado incluye un aclarado con agua limpia después del lavado. La aplicación de ceras es opcional de acuerdo a la necesidad, pero es una fase ampliamente extendida en todos los lavados, y se aplica, habitualmente, a través de la maquinaria del lavado y en frío. Se pueden utilizar varias ceras, según la finalidad. Si es así, primero, se suele aplicar una cera abrillantadora que tiene la función de cubrir las raspaduras que pueda tener la pintura, proporcionando una superficie lisa [19].
4	Secado	Como última fase del proceso de lavado tenemos el secado del vehículo. Esta es una fase opcional que se ofrece sobre todo los lavados tipo túnel. Generalmente se hace a través de sopletes o turbo ventiladores que arrastran las gotas de agua que han quedado en la superficie del vehículo [19].

### 3.2. Sistemas de lavado

#### 3.2.1. Puente de lavado

Es un sistema de lavado controlado por ordenador, durante todo el proceso, el vehículo se mantiene estacionado en un solo lugar, mientras que el equipo de lavado se mueve hacia adelante y hacia atrás sobre los raíles, controlado por ordenador y según el programa elegido [20].

El lavado se realiza mediante cepillo, cortinas textiles o lavado sin contacto con rociadores de alta presión. Se aplican en las etapas del proceso de lavado detergentes y productos de acabado, según el programa. Generalmente, es el sistema más utilizado a nivel de limpieza para el lavado de camiones, buses, etc. En estos casos se suele complementar con un prelavado manual con agua a presión y temperatura [20].

La velocidad de lavado de un puente, en función de los programas y modelos, suele ser en promedio de unos 8-12 vehículos por hora y 4-5 camiones por hora [19] [20].

#### 3.2.2. Túnel de lavado

En este sistema de lavado controlado por ordenador, el vehículo avanza linealmente gracias a una cinta transportadora, mientras que los elementos del túnel se encuentran fijos. Si se dispone de un servicio de personal presente (operador), a menudo se realiza

un pre tratamiento (desincrustante), con agua a presión. Existen también a nivel general tres tipos de túnel de lavado: con cortinas textiles (lavado suave) o con cepillo de nailon, y lavado sin contacto. Habitualmente, todo el proceso de lavado se lleva a cabo con agua fría [21].

La capacidad de los túneles de lavado de coches suele ser en promedio de unos 50- 60 coches por hora [21].

### **3.2.3. Auto-lavado de coches**

Es un servicio de auto-lavado (self-service) formado por un equipo (box de lavado), conformado por un compresor que impulsa agua y que va equipado con dos o tres pistolas: una, con el cepillo y la otra, impulsa agua a presión a la superficie. En este servicio el propietario del vehículo es el encargado de lavar él mismo el vehículo. Funciona con gajo automático por medio de monedas que introduce en la máquina, el usuario podrá elegir el ciclo de lavado, de acuerdo a las etapas: prelavado, lavado, aclarado y encerado. A diferencia de los túneles, la fase de lavado se hace con agua tibia para asegurar un buen resultado. No incluye la fase de secado, por el cual esta fase del proceso se realiza al aire libre. Para evitar la mancha de cal de la gota, la mayoría de boxes de auto-lavado tienen equipo de agua osmotizada para la fase de aclarado [21].

La capacidad de lavado en este sistema puede ser muy variable y estará en función del tiempo de duración que hayan escogido los usuarios. Se estima un promedio máximo de 7 coches por hora [21].

## **3.3. Sistemas de auto-lavado automático**

### **3.3.1. Clean-Car**

El Sistema de Lavado Clean-Car es un servicio de auto-lavado vehicular por fichas o "self service", Clean-Car está diseñada y construida de manera robusta, para un correcto desempeño diario, su estructura está conformada por tuberías de acero al carbono de probada resistencia, su exterior se compone de acero inoxidable AISI-304, lo que le confiere cualidades óptimas para un servicio a intemperie. Su operación es simple, intuitiva y amigable para el usuario al momento de lavar su vehículo [22].

Al colocar una ficha, se habilita el sistema de control, seguidamente el propietario del vehículo toma la lanza o pistola y apuntando hacia el vehículo se elige una de las 4 opciones que ofrece este modelo, 1) prelavado, 2) champo, 3) enjuague y 4) teflón en el panel de control, presionando el botón correspondiente, comienza el proceso hasta finalizar el tiempo correspondiente al crédito programado [22].

### **3.3.2. Kondor**

Otro sistema de lavado sin contacto automático es Kondor ofrece la posibilidad de actualizar cualquier tipo de lavadero ya que se puede adaptar a pistas nuevas o existentes. Kondor incorpora de serie 2 prelavados para ofrecer una limpieza superior y 1 cera de secado [23].

Incorpora un Sistema de posicionamiento virtual (VPS) y también una pantalla táctil permitiendo procesar los datos un 30% más rápido respecto a otros sistemas y poder flexibilizar los programas de lavado gracias al software que incluye. También incorpora un modem para envíos de mensajes, así los clientes controlan el trabajo remotamente. Se programa automáticamente con 2 números de teléfono donde se envían alertas de posibles fallos [23].

### **3.3.3. Saber S11**

La máquina de lavado sin contando denomina como Saber S11 lava una vez hecha la lectura de la máquina para saber las dimensiones del vehículo, hay 2 brazos articulados que van girando alrededor de la superficie del vehículo y haciendo las diferentes funciones como son el prelavado, el lavado, el aclarado y el encerado con osmosis, su diseño es de tipo abierto y libre no creando nunca claustrofobia y son de tracción eléctrica y sin necesidad de engrase, para reducir complejidad al momento de realizar el mantenimiento [24].

Además, la flexibilidad para la programación de la máquina mediante el teclado es versátil, también permite cambiar fácilmente entre sus ocho programas de lavado diferentes en cuestión de minutos adaptándose rápidamente a los cambios en las colas en tiempo real. Existe la posibilidad como una opción de incorporar una nueva pantalla táctil, con software de programación para un control remoto desde una página web propio de Saber 11, permitiendo el acceso a los datos e informes del día desde cualquier lugar del mundo [24].

### **3.4. Sistemas de cobro automáticos**

#### **3.4.1. CashDro**

El cajón de cobro inteligente Cashdro 5 es una máquina de gestión automatizada de almacenamiento de dinero en efectivo, custodia tu efectivo y permite que la caja siempre cuadre, CashDro es un sistema de cobro automático en el punto de venta de los establecimientos. Acepta, valida y dispensa billetes y monedas con devolución de cambio de forma automática [25].

Existe un CashDro para cada tipo de negocio o servicio que permite una gestión automatizada del efectivo, mejorando la seguridad del establecimiento, la productividad del personal y el servicio al cliente [25].

Ventajas principales:

- Cierre de caja instantáneo y sin descuadres.
- Rechaza billetes falsos.
- El efectivo siempre está seguro.
- Se eliminan errores de cambio.
- Se evitan pérdidas desconocidas de efectivo.
- Enlaza con cualquier software del mercado.
- Velocidad validación de monedas 4 monedas/s.
- Velocidad validación de billetes 3 s/billete.
- Acepta, valida y dispensa billetes EURO: de 5€ a 500€.
- Conexión Ethernet [25].

#### **3.4.2. ERP de Telemetal**

Es un sistema de cobro automático que permite realizar cobros directamente del usuario y en efectivo sin manipulación directa del personal, la solución definitiva para reducir errores humanos en los arqueos de caja utilizados especialmente en las empresas distribuidoras de material para la construcción [26].

A menudo, las empresas distribuidoras de material se encuentran con grandes problemas a la hora de cuadrar los cierres de caja. La principal causa es que la manipulación del dinero en efectivo es realizada por parte del personal, y este no es un sistema 100% fiable. Errores en los cambios, transacciones mal ingresadas o pequeños

hurtos. Pueden ser los motivos que ocasionan errores en los arqueos. Un sistema de cobro automático es la solución definitiva no solo para cuadrar caja, sino también para agilizar el tiempo de gestión y mejorar la atención al cliente [26].

### **3.4.3. Burget cajero automático**

En el caso de los cajeros automáticos, el cliente se dirige directamente con su ticket al cajero y pagar la estancia sin intervención del operario [27].

Disponen de diferentes tipos de cajeros automáticos para adaptarnos a sus necesidades:

- Sólo efectivo (monedas y billetes).
- Sólo tarjeta de crédito y Contacless.
- Efectivo, tarjeta de crédito, contactless y pago móvil (NFC), para el pago móvil es necesario que el cliente disponga de un Smartphone con tecnología NFC) [27].

Características:

- Electrónica basada en PC industria con microprocesador de última generación Intel Atom.
- Sistema operativo LINUX.
- Funcionamiento autónomo u On-Line. Comunicación TCP/IP con el servidor.
- Pulsadores táctiles anti vandálicos para selección de opciones de menú en pantalla.
- Lector de Proximidad (MIFARE) para el cobro de exceso o renovación de abonados.
- Sistema de interfonía IP para establecer conexión verbal con el centro de control [27].

En conclusión, los sistemas de cobro automáticos convierten los problemas de cierre y cuadre en una solución. Que tarda en promedio solo 3 minutos para la apertura y cierre de caja, el ahorro de tiempo equivale a 8 días de trabajo/año y 16 días/dos turnos de trabajo, también ahorrarás el tiempo de preparación del fondo de caja y la reposición del cambio al usuario. Los comercios que gestionan el efectivo mediante un cajón automático se ahorran en promedio unos 43 minutos/día, lo que equivale a 21

horas/mes para los comercios que abren de lunes a domingo. El ahorro total anual es de 258 horas/año el equivalente a 21 días de trabajo [28].

### 3.5. Selección de equipos

#### 3.5.1. Selección del autómata programable

En la siguiente tabla solo se describen los equipos que poseen los laboratorios de la facultad de Ingeniería en Sistemas, Electrónica e Industrial. Para la selección de equipos se define una serie de características, especificaciones, ventajas y cualidades que poseen, de manera siguiente se les pondera con una escala del 1 al 10: Siendo 1 bajo y 10 alto.

**Tabla 4:** Selección de PLC

Características	PLC				LOGO	
	S7-1200				V7	V8
	1212		1214			
	Dc/dc/dc	Ac/dc/dry	Dc/dc/dc	Ac/dc/dry		
<b>Nº entradas</b>	7	7	9	9	6	6
<b>Nº Salidas</b>	5	5	10	10	5	5
<b>Precio</b>	7	8	7	8	10	10
<b>Robustez</b>	9	10	9	10	7	7
<b>Disponibilidad en el mercado</b>	7	8	7	8	6	6
<b>Ámbito de Trabajo</b>	7	9	7	9	8	8
<b>Comunicaciones</b>	8	8	8	8	6	6
<b>Promedio</b>	7,14	7,85	8,14	8,85	6,85	6,85

El dispositivo a utilizarse en el presente proyecto es un PLC S7-1200 1214 Siemens Ac/dc/dry, con una relación de entradas y salidas mayor en comparación a los demás dispositivos, además trabaja en cualquier ámbito laboral y este tipo de controlador es utilizado en la mayor parte de aplicaciones industriales.

#### 3.5.2. Selección de SBD para servidor web

Para este caso se realizará un análisis de manera cualitativa entre las dos computadoras de placa única, que son las más conocidas por sus aplicaciones a nivel educativo.

**Tabla 5:** Selección de SBD

<b>Características</b>	<b>Arduino</b>	<b>Raspberry Pi</b>
<b>Conexión ethernet</b>	Adicional	Integrado
<b>Aplicación general</b>	Controlador	Varias funciones
<b>Código de programación</b>	Muy fácil	Complejo
<b>Precio</b>	Muy accesible	Accesible
<b>Sistema operativo</b>	Ninguno	Propio
<b>Velocidad</b>	16Mhz	700Mhz
<b>Multitarea</b>	No	Si

De acuerdo a los detalles referentes a la tabla 5, el ordenador seleccionado es la Raspberry Pi 3 que posee más características y ventajas superiores a su principal competidor, posee su sistema operativo propio lo cual facilita la realización de un servidor web, además posee dos salidas USB, una HDM y 40 pines GPIO.

### 3.5.3. Selección de sistema HMI

La supervisión y control de las diferentes variables del proceso en estudio se logra mediante un sistema SCADA que consta de una interfaz HMI, que se conecta con el dispositivo de control un PLC S7-1200.

**Tabla 6:** Selección de HMI

<b>Pantalla</b>	<b>Características</b>
<b>Simatic HMI KTP400</b>	<ul style="list-style-type: none"> <li>• Resolución 480x272</li> <li>• Acoplamiento al proceso S7-1200</li> <li>• Ethernet</li> <li>• Memoria de datos 256 MB</li> <li>• Teclas de función 8</li> <li>• PROFINET (LAN) X1</li> </ul>
<b>Simatic HMI KTP600</b>	<ul style="list-style-type: none"> <li>• Resolución 320x240</li> <li>• Acoplamiento al proceso S7-1200</li> <li>• Soporta protocolo para EtherNet/IP</li> <li>• Modicon (MODBUS TCP/IP)</li> <li>• PROFINET</li> </ul>
<b>Simatic HMI KTP700</b>	<ul style="list-style-type: none"> <li>• Resolución 800x480</li> <li>• Acoplamiento al proceso S7-1200</li> <li>• Ethernet</li> </ul>

- Teclas de función 10
- PROFINET (LAN) X1

Considerando aspectos relevantes y de monitorización de acuerdo la tabla 6, la pantalla táctil seleccionada para el diseño y programación en el presente proyecto es la Simatic HMI KTP600.

### 3.6. Diagrama de flujo del funcionamiento del prototipo

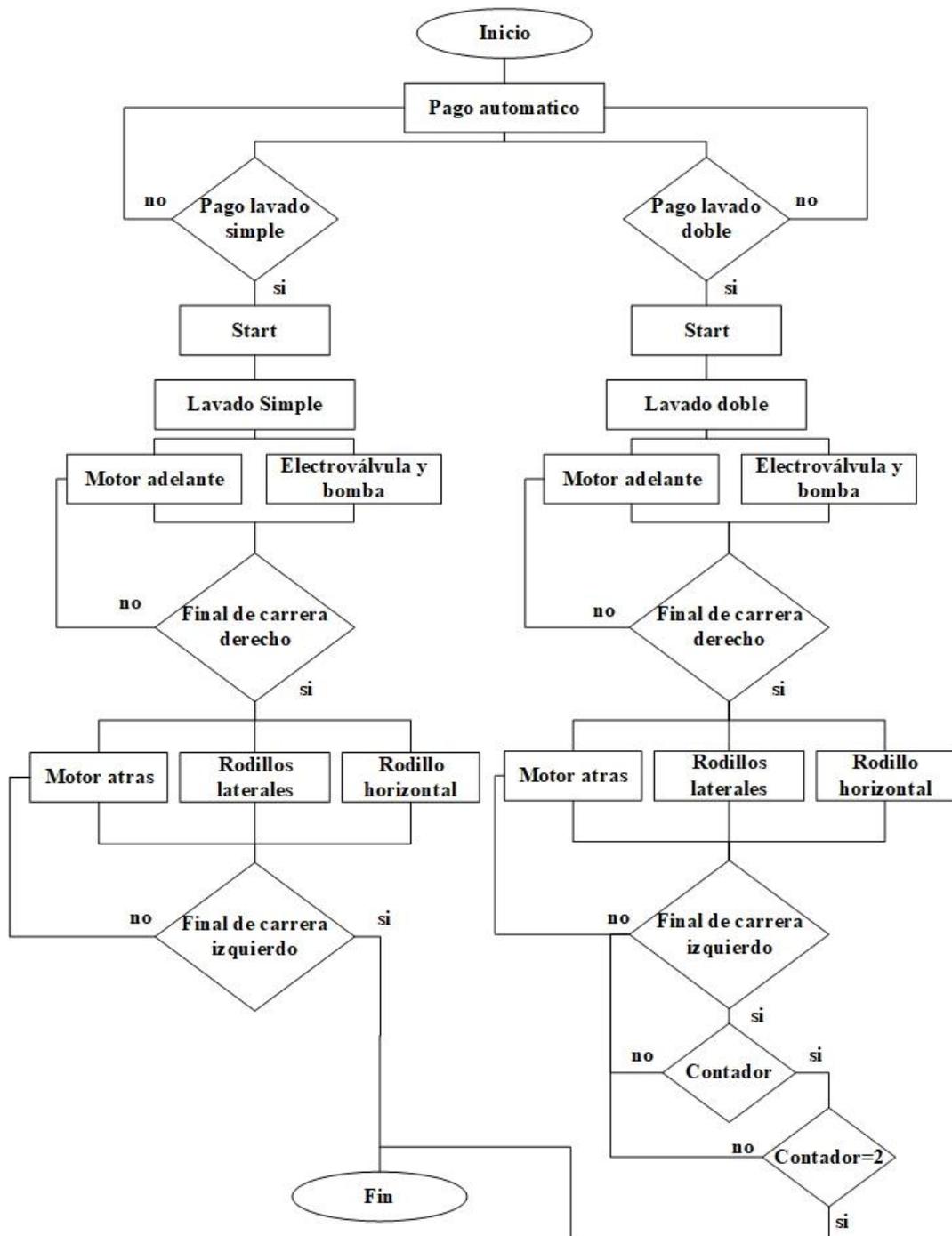
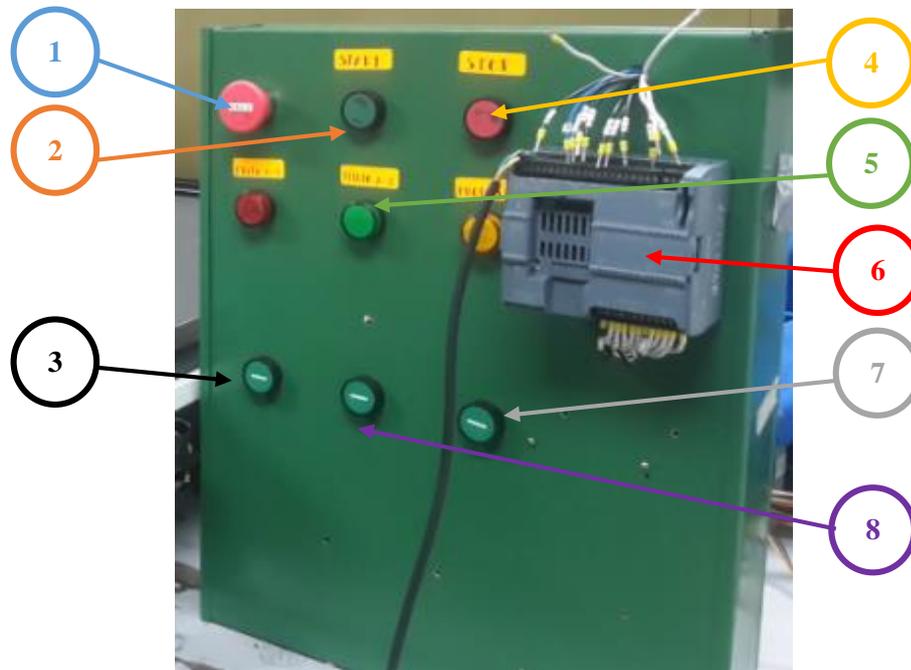


Figura 2: Diagrama de flujo

### 3.7. Prototipo de la estación de Auto-lavado

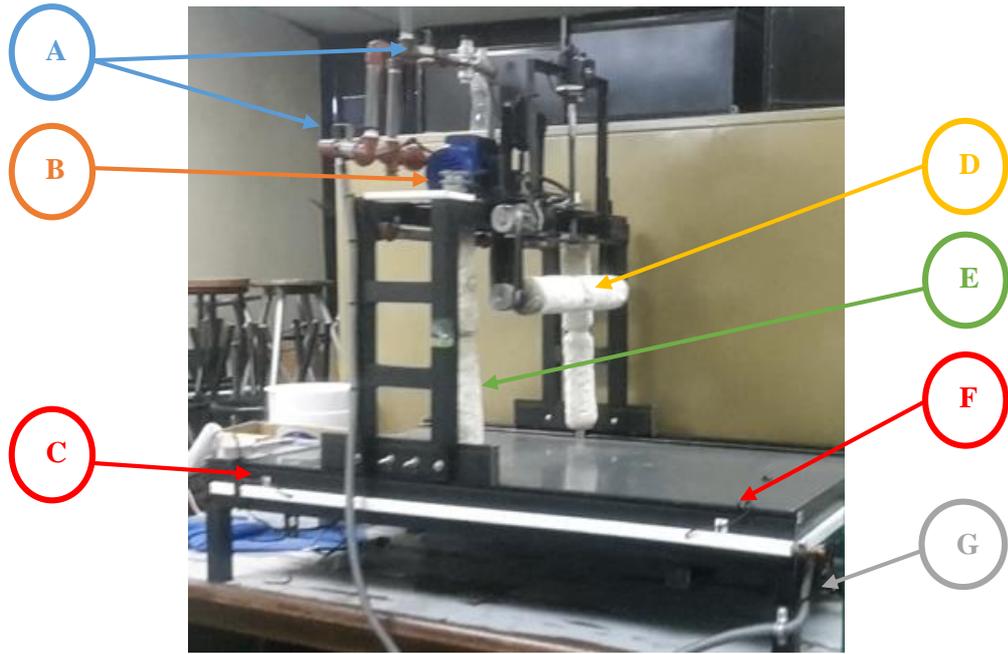
El prototipo está compuesto de dos partes: el primero es el tablero de control que contiene los pulsadores, luces piloto y el PLC, y el segundo que es la maqueta compuesto de la estructura y los actuadores (motores, electroválvulas, bomba de agua).



**Figura 3:** Tablero de control del prototipo de la estación

En la Figura 3, se muestra el tablero de control del prototipo de la estación de Auto-lavado cada elemento se describe a continuación:

- 1: Pulsador de emergencia
- 2: Pulsador de start (encendido del sistema)
- 3: Pulsador (lavado simple)
- 4: Pulsador de stop
- 5: Luces pilotos
- 6: PLC s7-1200
- 7: Pulsador (lavado doble)
- 8: Pulsador (reset)



**Figura 4:** Maqueta del prototipo de la estación

La maqueta del prototipo de la estación de Auto-lavado está formada por los siguientes elementos:

- A: Electroválvulas
- B: Bomba de agua
- D: Rodillo horizontal
- E: Rodillo laterales
- C: Final de carrera izquierdo
- F: Final de carrear derecho
- G: Motor hacia adelante y atrás

Las electroválvulas, los motores que permiten el giro de los rodillos laterales, horizontales y también el motor que permiten el desplazamiento de la plataforma hacia delante y atrás funcionan a 12 Vdc (corriente continua), la bomba de agua es alimentada a 110 Vac (corriente alterna) y los finales de carrera van a las entradas físicas del PLC.

### 3.8. Diagrama de conexión del prototipo

#### 3.8.1. Diagrama de conexión de entradas del PLC

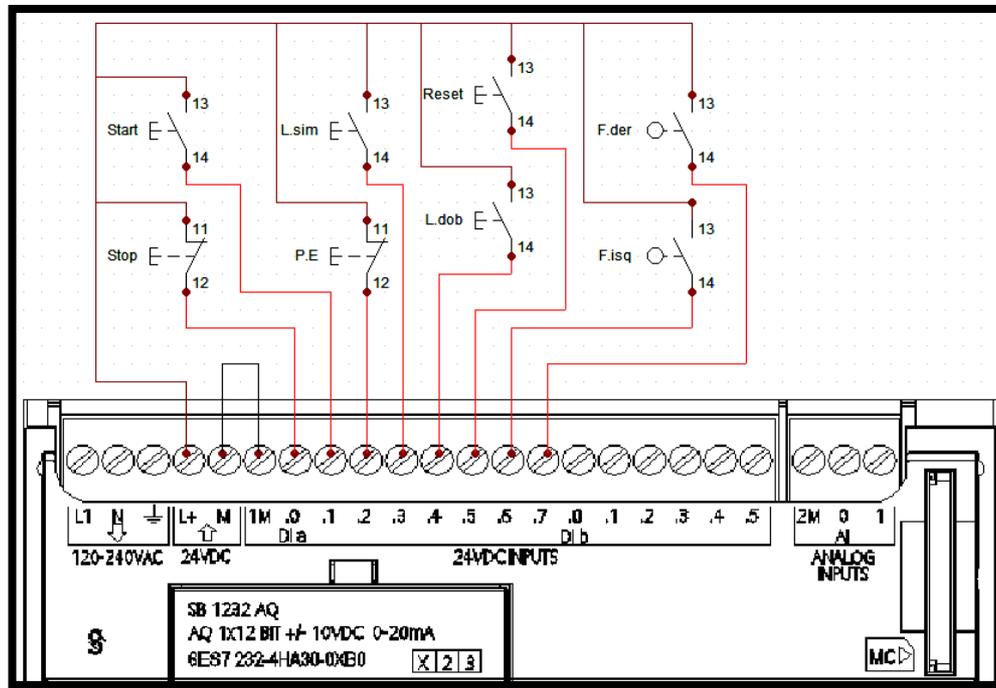


Figura 5: Diagrama de conexión de las entradas físicas del PLC

En el diagrama se puede observar la conexión de los diferentes pulsadores y finales de carrera que controlan el prototipo de la estación de Auto-lavado y son alimentadas de la fuente de 24 Vdc del PLC.

- P.E: Paro de emergencia
- Stop: Pulsador N.C
- Start: Pulsador N.O
- L.sim: Pulsador de lavado simple
- L.dob: Pulsador de lavado doble
- Reset: Pulsador de reset del contador
- F.isq: Final de carrera izquierdo
- F.der: Final de carrera derecha

#### 3.8.2. Diagrama de conexión de las salidas del PLC

A las salidas del PLC son conectados redes para activar cada actuador, como medio de protección al momento de ocurrir alguna falla en el sistema. Todos los motores son alimentados con una fuente externa de 12 Vdc (corriente continua), y los relés son alimentados con la misma fuente del PLC.

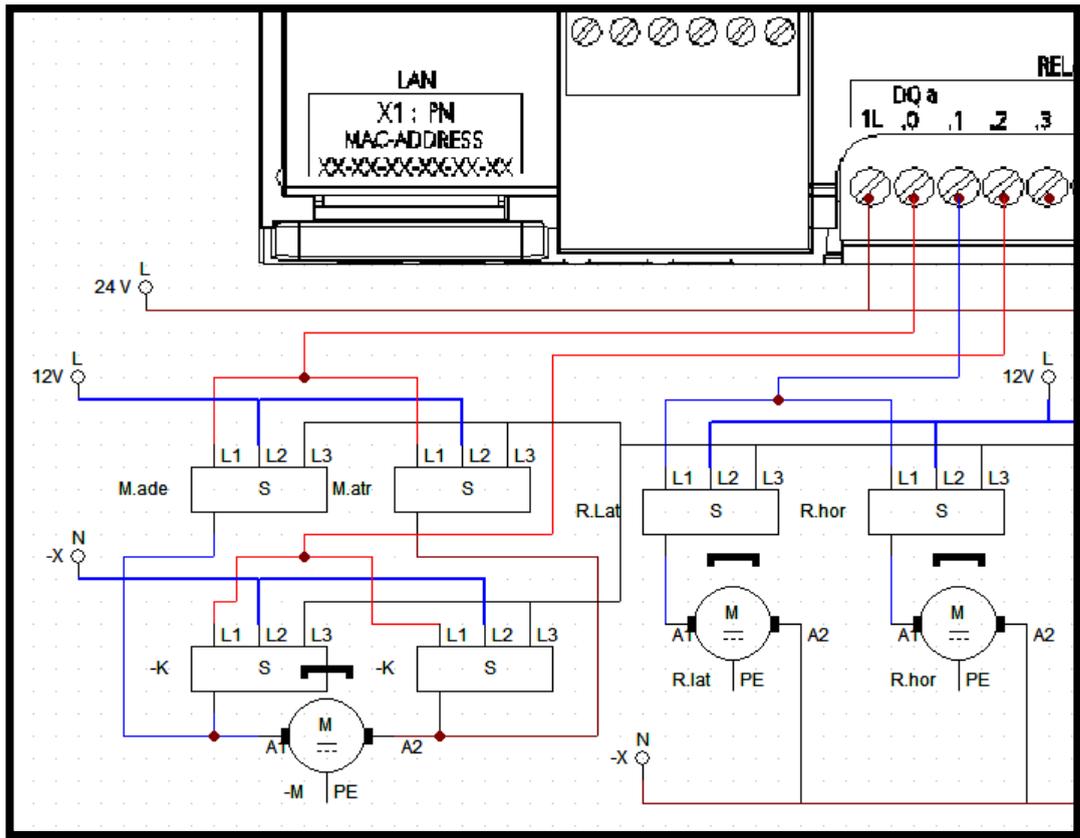


Figura 6: Diagrama de conexión de las salidas del PLC

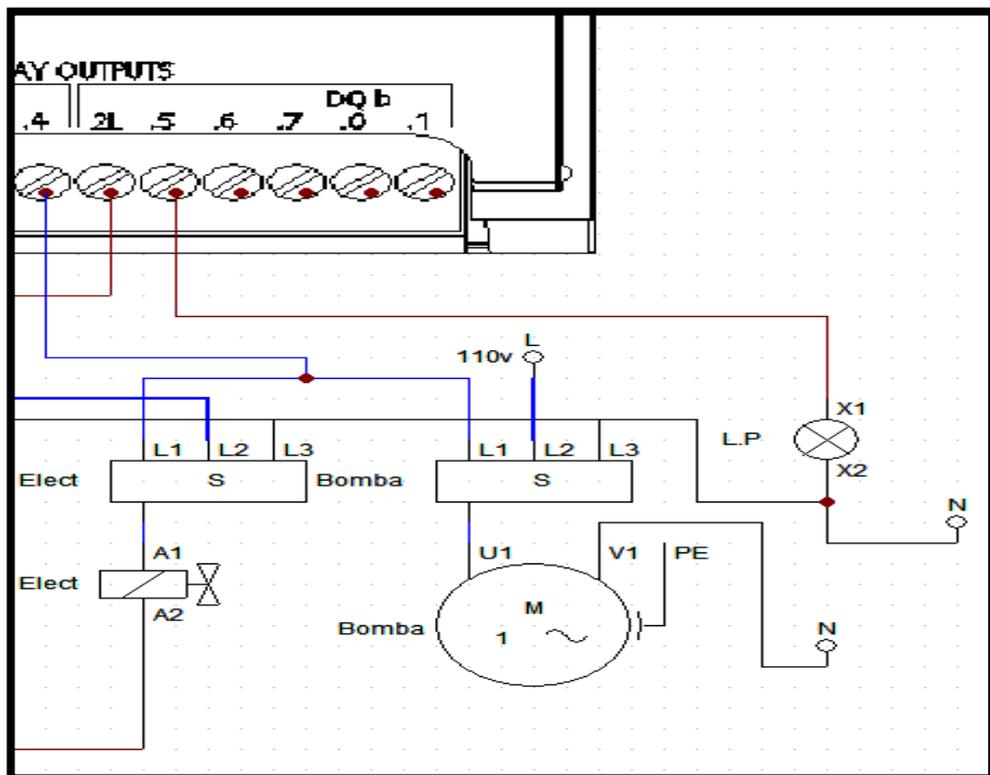


Figura 7: Diagrama de conexión de las salidas del PLC

La electroválvula es alimentada con la misma fuente de los motores, y la bomba que succiona el agua para el lavado, es activada por una red bifásica de 110Vac (corriente alterna).

### 3.9. Programación del PLC

En la actualidad el PLC s7-1200 es de mayor utilización dentro de las industrias pertenecientes a la marca Siemens. Para iniciar con la programación es necesario conocer los parámetros de funcionamiento del prototipo de auto-lavado, para este caso va a funcionar de 2 maneras: lavado simple y lavado doble, al momento de pulsar una de las dos opciones de lavado el prototipo empezara a funcionar.

#### 3.9.1. Software TIA Portal

Para la programación se utiliza el software TIA portal V13, que es una plataforma amigable y permite desarrollar varios proyectos con PLC's y HMI's.

#### 3.9.2. Variables utilizadas en el PLC

Antes de programar es conveniente declarar todas las variables utilizadas en la presente investigación.

**Tabla 7:** Declaración de variables

Tabla de variables estándar						
	Nombre	Tipo de datos	Dirección	Rema...	Visibl...	Acces...
1	Stop	Bool	%I0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2	Paro de Emergencia	Bool	%I0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	START	Bool	%I0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
4	PILOTO AMARILLO	Bool	%Q0.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
5	Lavado Simple(1)	Bool	%I0.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
6	FINAL_CARRERA_DER(1)	Bool	%I0.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
7	MOTOR ADELANTE	Bool	%Q0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
8	FINAL_CARRERA_IZQ(1)	Bool	%I0.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
9	MOTOR_REGRESO(1)	Bool	%Q0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
10	Bomba y Electro	Bool	%Q0.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
11	R. Lateral y Vertical	Bool	%Q0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
12	Lavado doble	Bool	%I0.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
13	Contador	Bool	%M0.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
14	Reset	Bool	%I0.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
15	Memoria Lavado Simple	Bool	%M8.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
16	Memoria Lavado Doble	Bool	%M9.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
17	<Agregar>			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

### 3.9.3. Programación para lavado simple

Toda aplicación debe contar con su debida protección de seguridad, en este caso un pulsador de paro de emergencia que dejara sin funcionamiento al sistema, ya que corta la energía que se dirige a los actuadores apagándolos de manera inmediata.

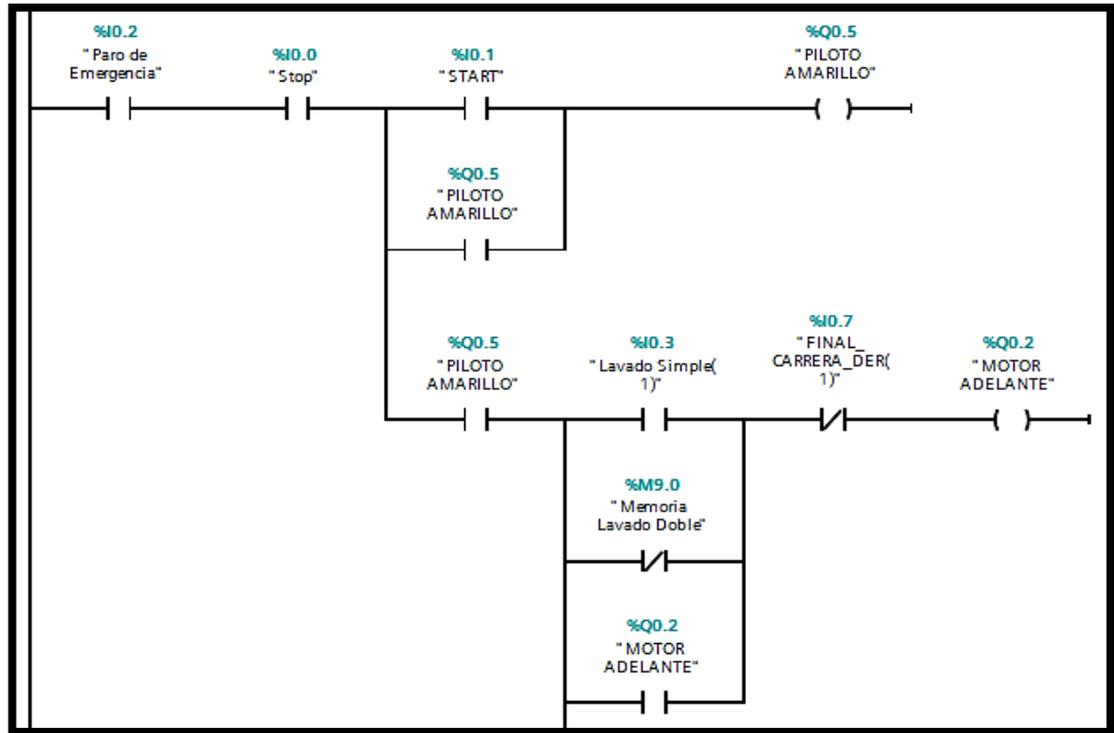


Figura 8: Programación segmento 1

El funcionamiento comienza con la entrada i0.1, que pertenece a un pulsador seguidamente se enciende una luz piloto que indica, que el sistema ya está encendido a la espera de la selección de uno de los dos tipos de lavado, en este caso lavado simple.

Al seleccionar el primer tipo de lavado se activa la salida que pone en marcha el motor con un tiempo de retraso a conexión mínimo ya que el mismo tiene que invertir de giro.

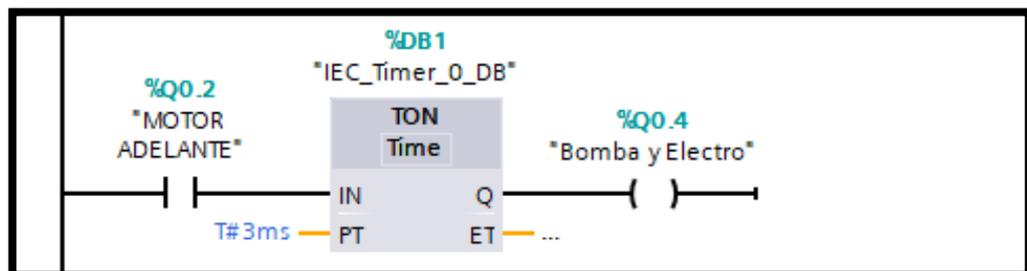


Figura 9: Programación segmento 2

Además, en el siguiente doble del programa se pone el funcionamiento la bomba de agua, conjuntamente con las electroválvulas que permiten el paso del fluido para el lavado.

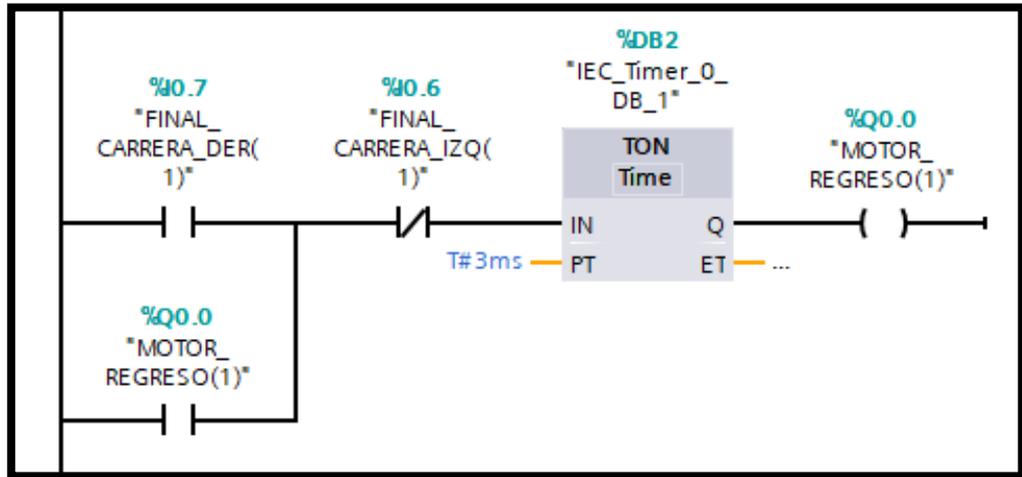


Figura 10: Programación segmento 2

Posteriormente un final de carrera que se encuentra al tope del prototipo se activa y el motor principal invierte de giro.

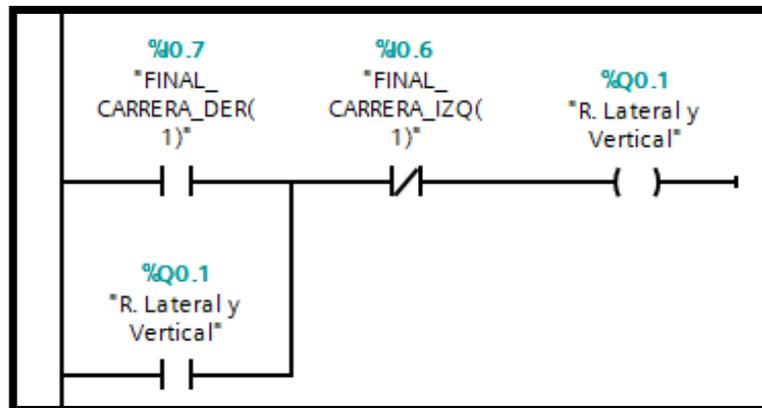


Figura 11: Programación segmento 3

Conjuntamente con el retroceso del motor, se activan los rodillos laterales e horizontales para la limpieza del auto, llegando a otro final de carrera que termina el proceso de lavado simple.

### 3.9.4. Programación para lavado doble

El funcionamiento es el mismo que el lavado simple, el único cambio es que se cumplen dos secuencias y se para automáticamente el sistema, pero se utilizó otra entrada la i0.4 perteneciente a otro pulsador.

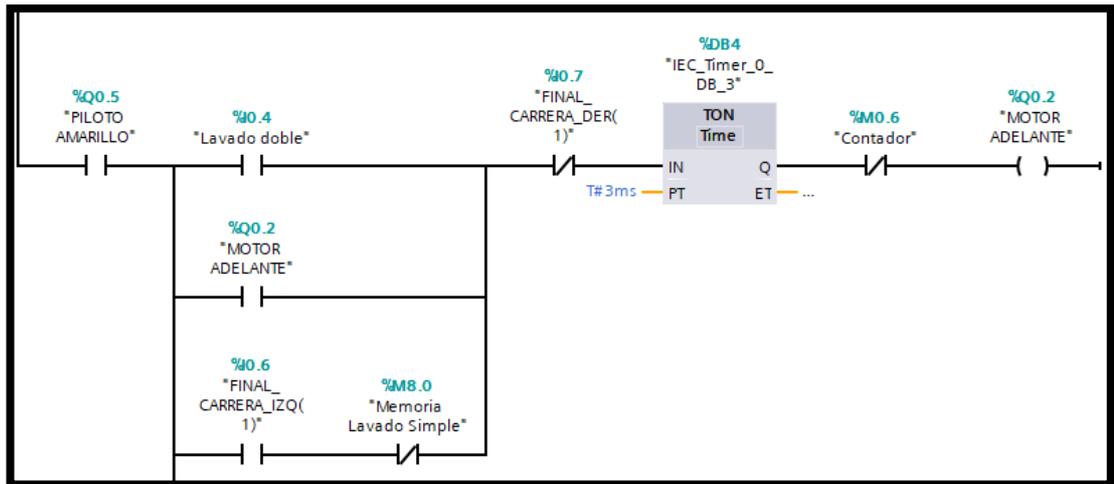


Figura 12: Programación segmento 4

Para activar el nuevo ciclo de repetición se logra, con la activación del final de carrera que termina el lavado simple, encendiendo de nuevo el motor delantero dando inicio el funcionamiento del prototipo.

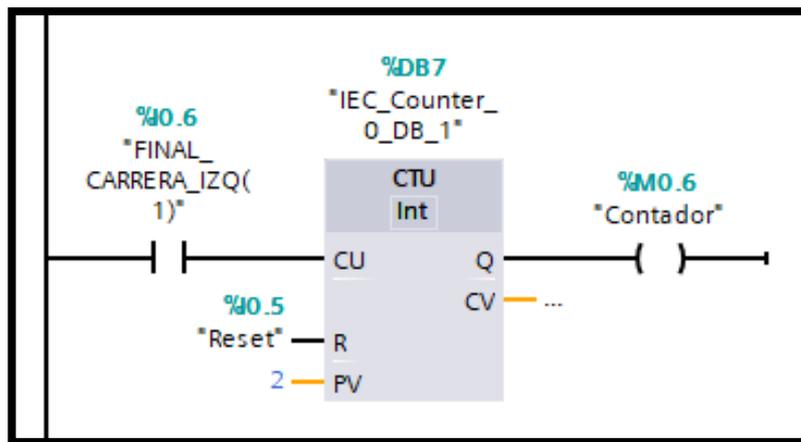
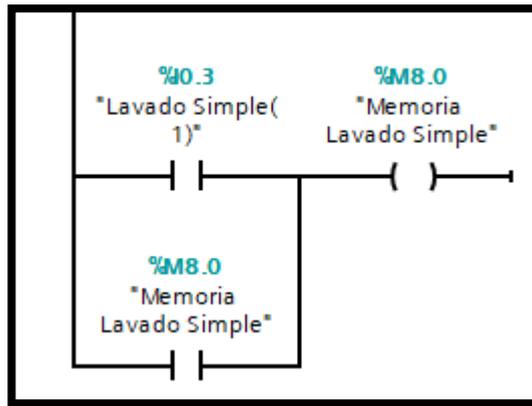


Figura 13: Programación segmento 5

El contador es el encargado de parar el funcionamiento del prototipo, al momento de contar dos pulsos que son dados con la activación del final de carrera, el cual activa un espacio de memoria y abre un contacto puesto antes de la salida del motor delantero, además con otro pulsador reseteamos el contador.



**Figura 14:** Programación segmento 6

Para que funcione únicamente uno de los dos tipos de lavados programados en el prototipo, se logra con la implantación de un circuito memoria.

### 3.10. Desarrollo de las pantallas HMI

Para monitorizar y controlar el proceso automatizado del prototipo de auto-lavado, se programó con la ayuda del software TIA portal V13, las diferentes pantallas que contienen valores, avisos e indicadores de lo que pasa en el proceso, además de poder controlar de manera manual y automática el sistema.

#### 3.10.1. Consideraciones en el desarrollo de las HMI

Para el desarrollo de las interfaces gráficas se escoge el tipo, tamaño de letra más adecuado, ya sea para título, subtítulos, botones de comando y datos numéricos del proceso, los cuales se representan en la siguiente tabla.

**Tabla 8:** Selección de tipo y tamaño de fuente según la Norma GEDIS.

	<b>Tipo de letra</b>	<b>Tamaño</b>	<b>Referencia</b>	<b>Color</b>	<b>Alineación</b>
<b>Títulos</b>	Fuente MS Sans Serif	19 pts.	<b>Títulos</b>	Negro	Central
<b>Subtítulos</b>	Fuente MS Sans Serif	17 pts.	<b>Subtítulos</b>	Negro	Central
<b>Botones de comando</b>	Fuente MS Sans Serif	13 pts.	<b>Botones de comando</b>	Blanco	Central

<b>Datos numéricos de proceso</b>	Fuente MS Sans Serif	13 pts.	<b>Datos numéricos del proceso</b>	Negro	Central
<b>Alarmas</b>	Fuente MS Sans Serif	13 pts.	<b>Estado de alarmas</b>	Negro	Central

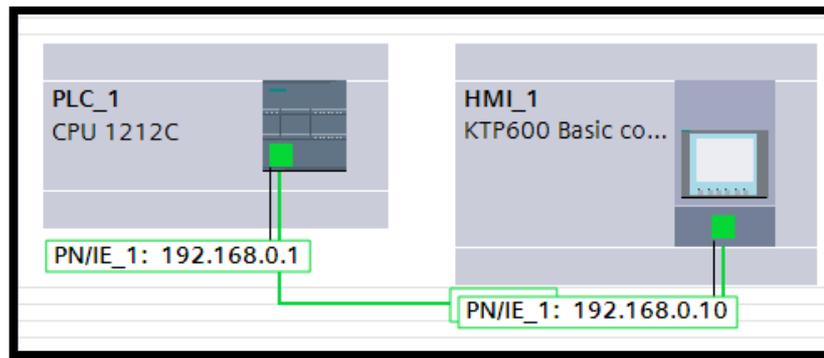
El HMI del prototipo de estación de auto-lavado usa el siguiente uso de color para garantizar la armonía entre el usuario y las pantallas, en la tabla 9 se visualizan los diferentes colores utilizados en los aspectos considerados.

**Tabla 9:** Selección de colores según la Norma GEDIS.

<b>ASPECTO</b>	<b>COLOR A USAR</b>	<b>Color</b>
<b>EQUIPOS DE LA PLANTA</b>		
Botón de RESET	Plomo	
Botón de STOP	Rojo	
Botón de START	Verde	
Botón de Lavado simple	Celeste	
Comandos de Lavado doble	Violeta	
Estatus del proceso encendido	Verde	
<b>FONDO DE PANTALLA</b>		
Fondo de pantalla principal	Gris	
Fondo de pantalla de para la bomba	Gris	
Fondo de pantalla de alarmas y etapas de lavado	Gris	
<b>TEXTO EN GENERAL</b>		
Valores de proceso	Negro	
Texto en general	Negro	

### 3.10.2. Conexión entre PLC y HMI

Una vez escogida la pantalla de acuerdo a los requerimientos de la automatización, se tiene que seleccionar el tipo de conexión para comunicar con el autómata y automáticamente se crea el driver de conexión. Para observar si la conexión fue exitosa debe existir una línea verde entre los dos dispositivos como se observa en la imagen.



**Figura 15:** Conexión de la interfaz

### 3.10.3. Pantalla principal

Con la ayuda de las funciones que ofrece la interfaz del TIA portal se programó el proceso de auto-lavado y se diseñó la interfaz principal, además permite monitorear las variables en tiempo real, de todo lo que está pasando dentro del prototipo.



**Figura 16:** Diseño de la pantalla principal

La estructura de la pantalla principal se observa en la figura 16, cuenta con el título, logotipo, hora y fecha, comandos operativos. La figura 17 presenta el aspecto final de la pantalla principal.

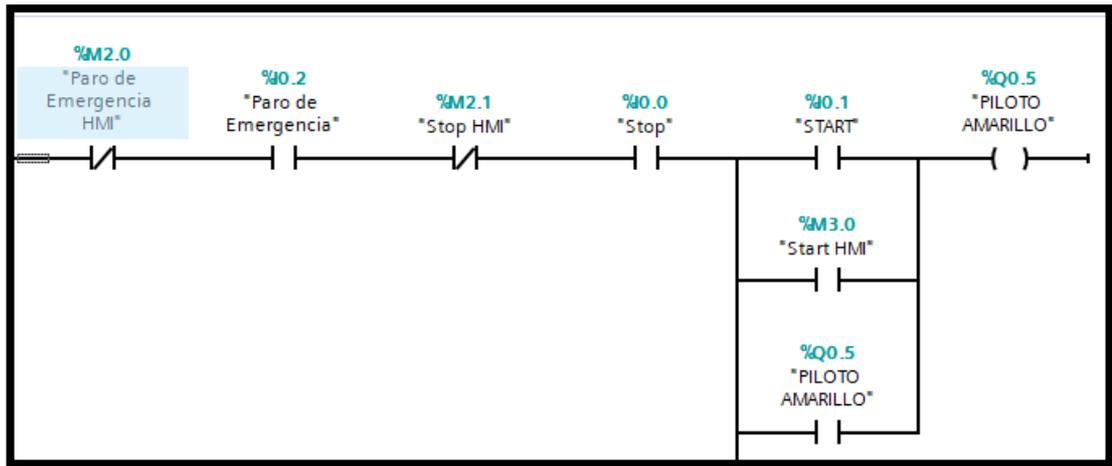


**Figura 17:** Pantalla principal HMI

En la figura 17, se puede observar el diseño de la pantalla principal el cual consta de los siguientes elementos:

- Paro de emergencia: en caso de existir alguna falla para todo el sistema.
- Start: Encendido del sistema.
- Stop: Apagado del sistema.
- Luz piloto: Indicador que el sistema esta encendido.
- Lavado simple: primer ciclo de lavado.
- Lavado doble: Segundo ciclo de lavado.

Todas estas variables presentes en la pantalla principal deben ser añadidos en el bloque de programación de los diferentes ciclos de lavado.



**Figura 18:** Programación segmento pantalla principal

Se puede observar las variables o espacios de memoria que son añadidas dentro del segmento de programación, que pertenecen a los diferentes botones que están en la pantalla.

### 3.10.4. Pantalla para el funcionamiento de la bomba



**Figura 19:** Diseño de la pantalla de la bomba de agua

La pantalla cuenta con los tres parámetros básicos de la pantalla principal, mismos que se repiten en todas las pantallas, además se encuentra el estado de la bomba, y los tiempos de funcionamiento.



**Figura 20:** Pantalla tiempo de funcionamiento de la bomba de agua

Se debe ingresar el tiempo para que la bomba de agua funcione en milisegundos, se puede considerar este tiempo como la utilización responsable del agua que se utiliza para el lavado de autos, para que no exista un desperdicio excesivo. También se muestra el paso del tiempo hasta que la bomba se apague.



**Figura 21:** Programación del tiempo de apagado de la bomba

Para ingresar y observar el tiempo programado para el funcionamiento de la bomba de agua se añade la variable %MD10 Y %MD20 de tipo DWord, en un temporizador de tipo Tp impulso.

### 3.10.5. Pantalla de tiempo real



**Figura 22:** Diseño de la pantalla de tiempo real

La estructura de esta pantalla es simple puesto que solo indican los diferentes actuadores que conforman la estación de auto-lavado.



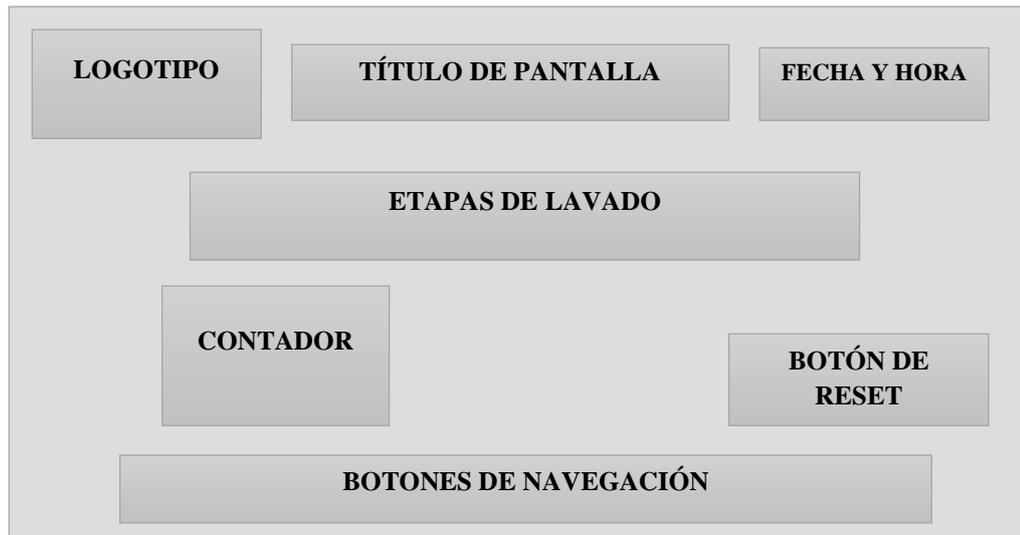
**Figura 23:** Pantalla de tiempo real

En esta pantalla se observa el estado de los actuadores, ya sea que estén prendidos o apagados, además se muestra de manera visual y textual el funcionamiento del motor

que se dirige hacia adelante y hacia atrás, también del encendido de los rodillos laterales y horizontales.

Las variables utilizadas dentro de los indicadores o luces piloto de cada motor, son las mismas salidas físicas del PLC.

### 3.10.6. Pantalla de etapas de lavado



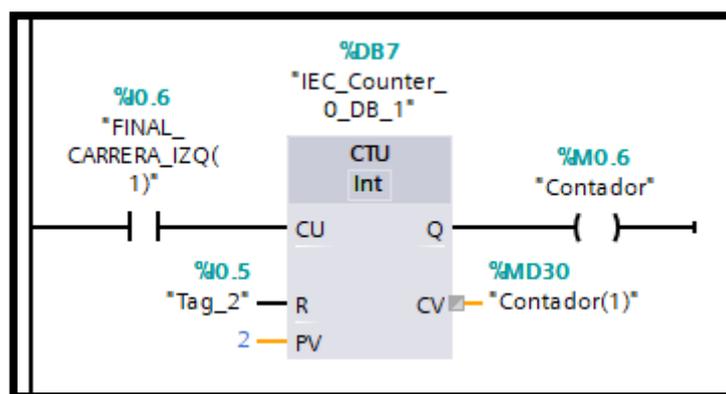
**Figura 24:** Diseño de la pantalla etapas de lavado

La pantalla cuenta con los tres parámetros básicos de la pantalla principal, además se encuentra el estado de las etapas de lavado, un contador y un botón de reset, los botones de navegación permanecen fijos, los cuales permiten desplazarse con libertad a cualquier pantalla dentro del sistema de lavado sin ningún tipo de restricción.



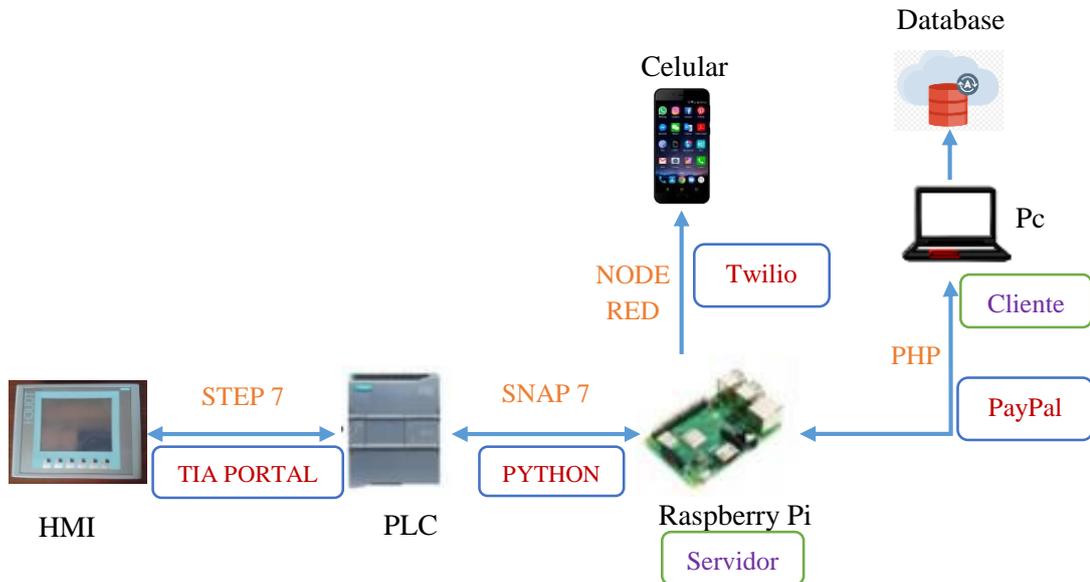
**Figura 25:** Pantalla etapas de lavado

En esta pantalla se observa la etapa en la que se encuentra el prototipo de estación de auto-lavado, ya sea lavado simple, lavado doble y el final del lavado doble, esto se logra con la visualización de un contador enlazado al final de carrera que mediante un punto procede a contar. Además, la visualización se realiza a través de la variable %MD30, y para resetear se realiza con una entrada física del PLC.



**Figura 26:** Programación para visualizar el contador

### 3.11. Plataforma de arquitectura del prototipo



**Figura 27:** Plataforma de arquitectura

El desarrollo del sistema planteado abarca una estructura simple, para programar tanto el PLC como el HMI se utilizó, el software de programación SIMATIC STEP7 (TIA portal V13), para la comunicación entre el PLC y la Raspberry Pi se utiliza la Librería SNAP7, que termine la escritura y lectura de datos de todos los tipos de variables (bits, Word, Dword, Real). Para que esos datos sean enviados a una página web se utiliza el lenguaje PHP, que con la importación de los scripts de PYTHON en la programación lean o escriban las variables del PLC en tiempo real. Para el mensaje SMS de alarma se emplea la programación de flujo visual NODE-RED, y para el pago automático se utiliza el sistema de pago en línea PayPal, todos los usuarios que requieran el sistema primeramente serán guardados en una base de datos.

### 3.12. Diseño del mensaje de alarma por SMS a teléfono móvil

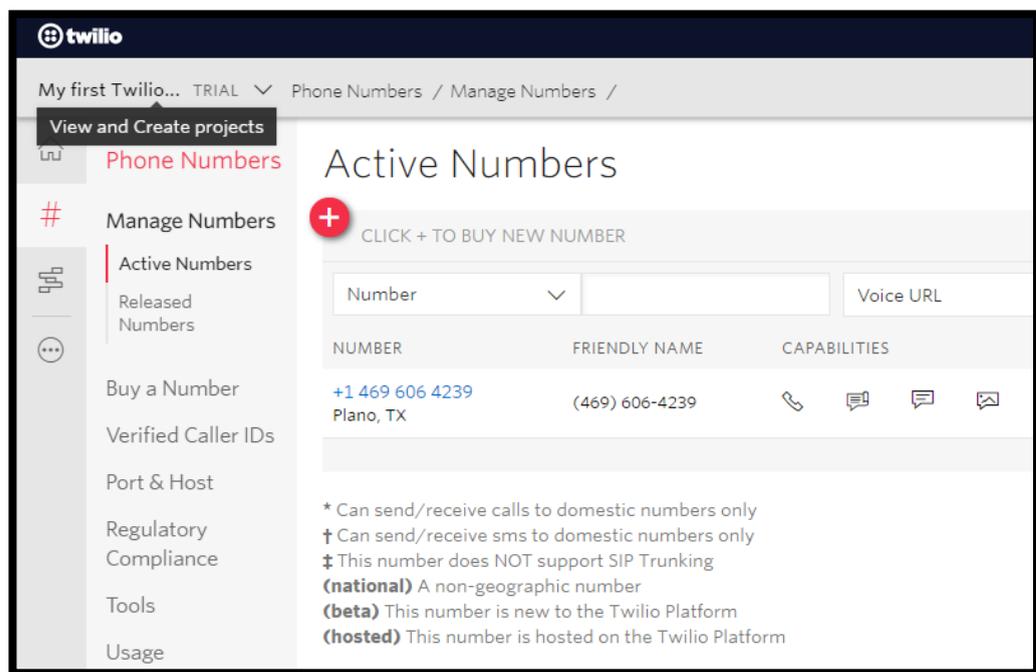
El mensaje de alarma es enviado una vez que haya ocurrido algún problema en la estación de auto lavado, cuando el botón de emergencia es pulsado inmediatamente un mensaje llega al número del teléfono móvil del cliente, para motivos de prueba solo se realiza con un numero de celular.

### 3.12.1. Plataforma Twilio

Twilio es una plataforma que permite a los desarrolladores construir aplicaciones de comunicación con la nube y sistemas web avanzados. Las API de comunicaciones de Twilio permiten a las empresas proporcionar una experiencia de comunicación adecuada para sus clientes dentro de la web y en aplicaciones móviles. Al usar las API de Twilio, los desarrolladores pueden agregar estas funcionalidades a una aplicación, como mensajes de voz, video llamadas, mensajes de texto y más [29].

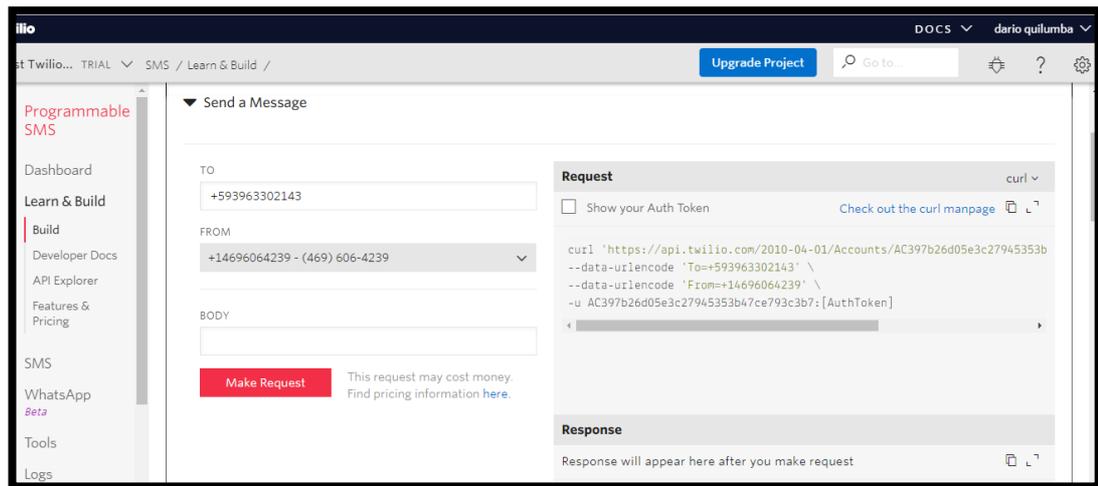
### 3.12.2. Creación de cuenta en plataforma Twilio

Para crear una cuenta se debe ingresar un correo electrónico y posteriormente ingresar el número de celular, al cual está vinculada la cuenta. Posteriormente se procede a liberar un número de teléfono propio de la plataforma con la opción de compra que es gratuita.



**Figura 28:** Número de teléfono liberado en la plataforma Twilio

En la figura 28, se muestra el número que fue liberado el cual es el utilizado en la programación de Node-Red para él envío del mensaje de alerta hacia el teléfono móvil vinculado en la plataforma.



**Figura 29:** Código base de programación de mensaje

En la opción de programar mensajes se ingresa, el número vinculado y el liberado, e inmediatamente la plataforma muestra los parámetros a utilizar, en la programación para el envío del mensaje.

### 3.12.3. Programación en Node-Red para envío de mensaje

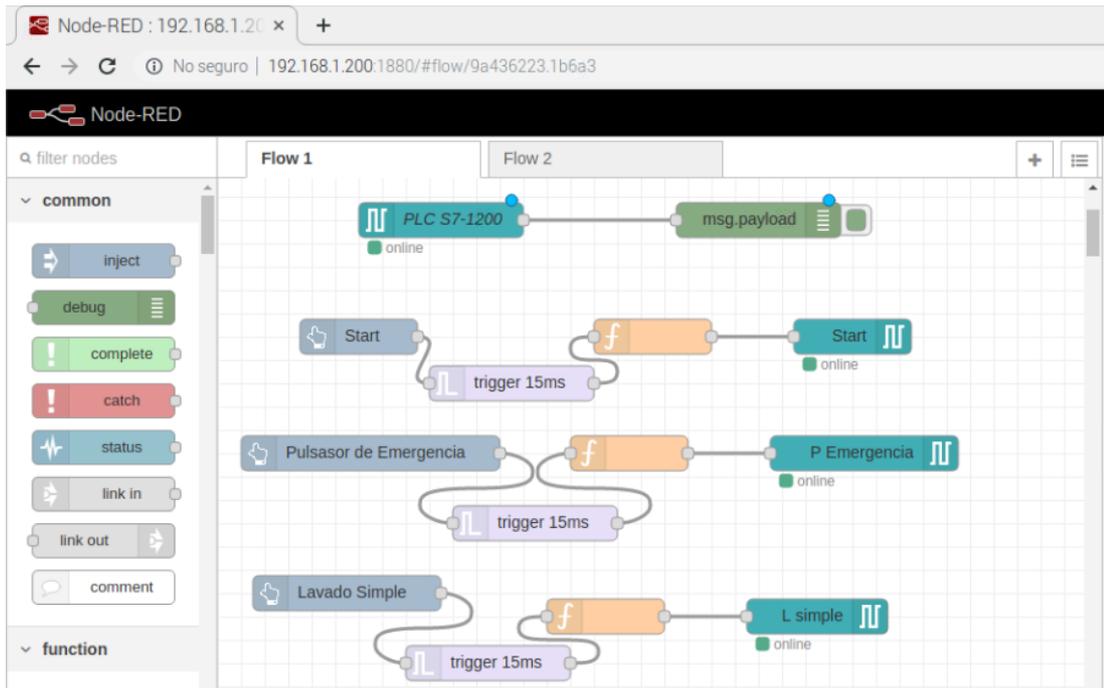
Node-Red es una herramienta de programación para conectar dispositivos de hardware, API y servicios en línea de formas nuevas e interesantes.

Proporciona un editor basado en navegador que facilita la conexión de flujos utilizando la amplia gama de nodos en la paleta que se pueden implementar en su tiempo de ejecución con un solo clic.

El bloque de programación se divide en dos, el primero es la conexión entre el plc s7-1200 y la raspberry pi, y el segundo es el que envía el mensaje al teléfono móvil.

### 3.12.4. Programación para obtener datos del PLC

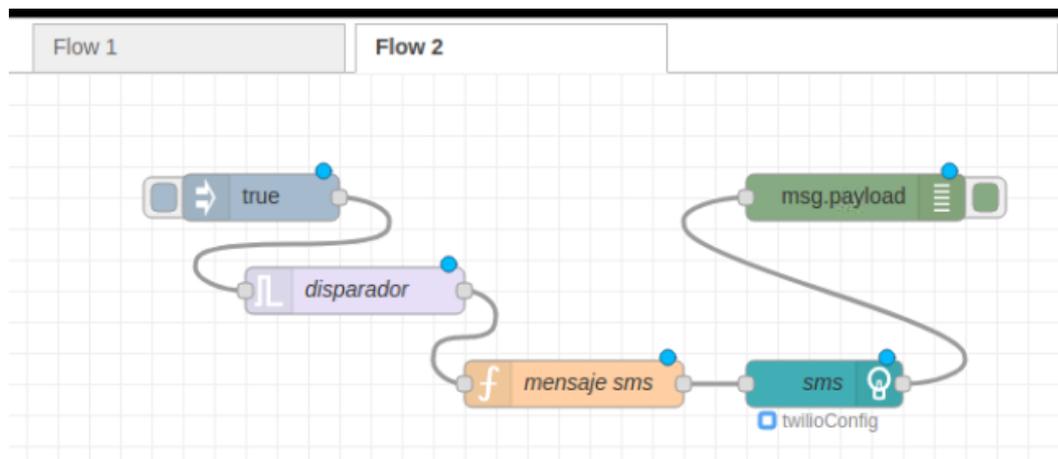
En el bloque de entrada denominado PLC S7-1200 se declaran las variables, que son utilizadas en los bloques de salida, al momento de pulsar el botón de parada de emergencia, crea un pulso y envía una señal al siguiente programa que se encuentra en Flow 2.



**Figura 30:** Programación de PLC en Node-Red

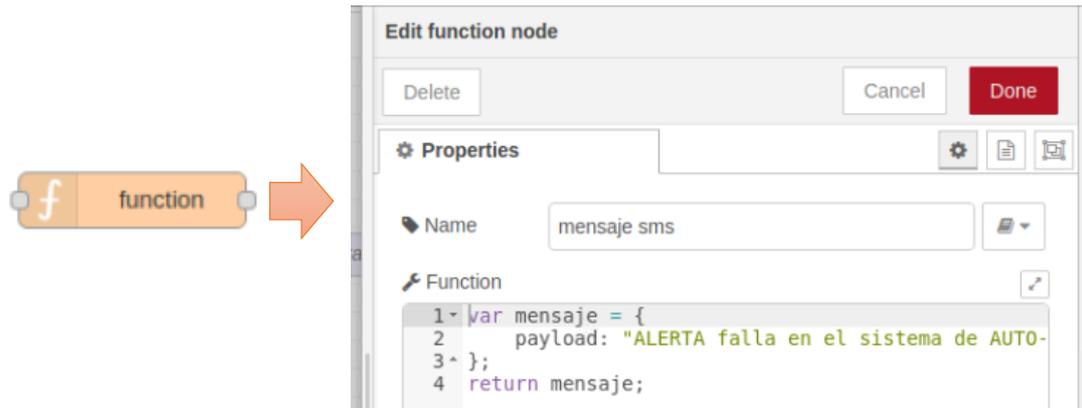
### 3.12.5. Programación para envío de mensaje

Con el siguiente programa Flow 2, se envía el mensaje de texto al número configurado en la plataforma de Twilio, y se utilizó la misma estructura que muestra, como ayuda la plataforma.



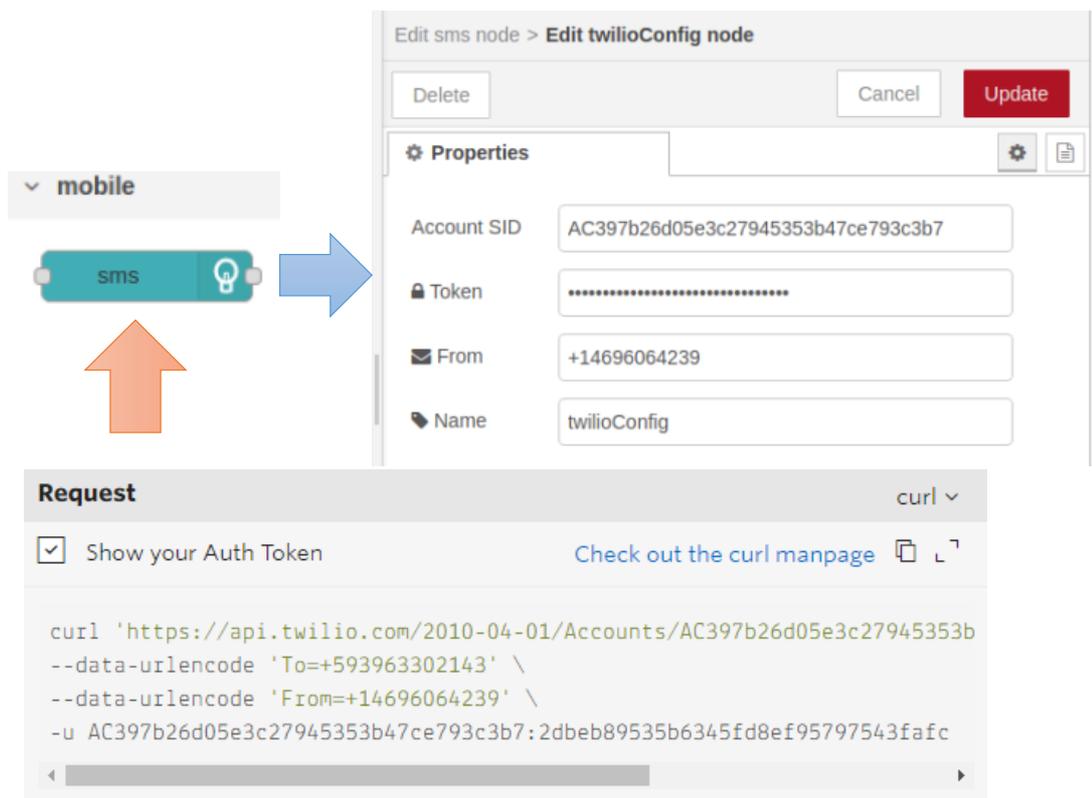
**Figura 31:** Estructura final del programa para enviar de mensaje

Comienza con la detección de un impulsó, seguido de un trigger con el tiempo de envío, también los bloques de programación esenciales son explicados a continuación.



**Figura 32:** Programación bloque de función

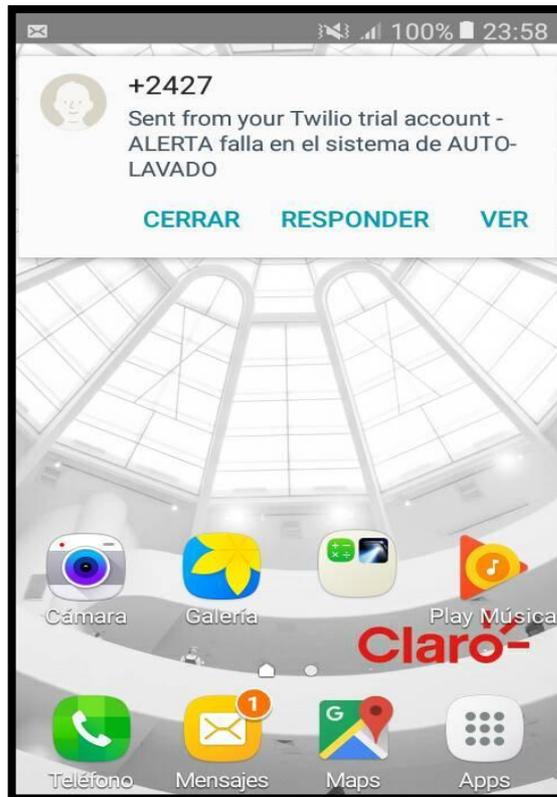
En este bloque se programó el mensaje que dice lo siguiente **“ALERTA falla en el sistema de AUTO-LAVADO”**, que se envía una vez detectado el cambio de impulso en el pulsado de paro de emergencia.



**Figura 33:** Programación bloque de mensaje

En el bloque de mensaje se añade los datos que provienen de la página de Twilio una vez que se libera el primer número de teléfono. Se ingresa SID de la cuenta, el token que viene a ser la contraseña, el número que envía y recibe el mensaje de aviso.

### 3.12.6. Recepción del mensaje de alerta

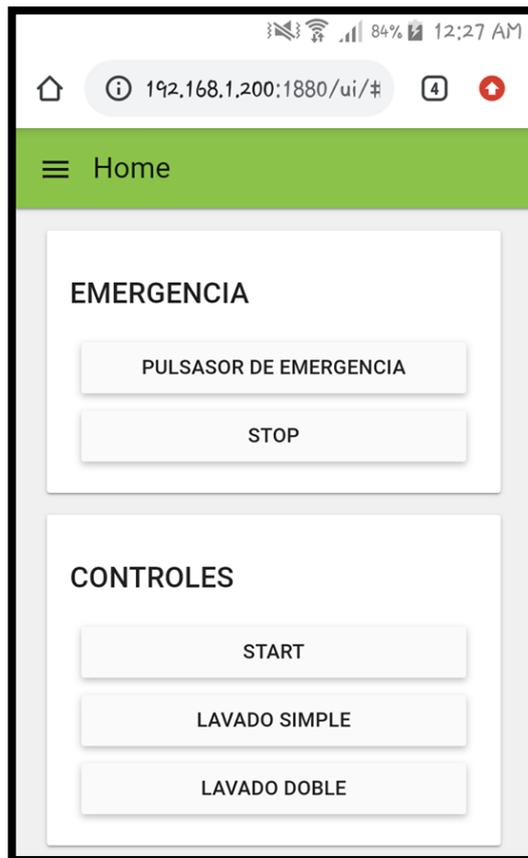


**Figura 34:** Recepción de mensaje

El mensaje de alerta enviado mediante la programación se puede observar en la pantalla principal del teléfono, esto se logró con la vinculación entre el PLC S7-1200, raspberry pi, Node-Red, Twilio y las plataformas web instalas anteriormente en el sistema operativo de raspbian.

### 3.12.7. Página web con Node-Red

Por motivos de prueba entre la conexión del PLC s7-1200 y la raspberry pi se creó, también una pequeña página web con los botones de mando y paro, se demuestra que se puede controlar de manera remota la estación de auto-lavado. Además, la prueba también se lo realizo por medio de un teléfono ingresando la dirección a la que está vinculada la página. Todo el diagrama de flujo de la programación del envío del mensaje de alarma se encuentra en el **Anexo 3**.



**Figura 35:** Página web con Node-Red

### **3.13. Diseño del pago automático**

Para que el pago se realice de manera automática por el servicio prestado, en este caso el lavado de vehículos, se utiliza la plataforma PayPal que es un sistema de pagos en línea para vendedores o personas que ofertan servicios además soporta transferencias de dinero entre usuarios.

#### **3.13.1. Plataforma PayPal**

PayPal Holdings es una empresa estadounidense que opera en casi en todos los lugares del mundo, es un sistema de pagos en línea, además soporta transferencias de dinero entre usuarios y sirve como una alternativa electrónica a las técnicas de pago tradicionales. PayPal es una de las mayores empresas de pago por Internet del mundo. La compañía opera como un procesador de pagos para vendedores y compradores en línea, sitios de subastas y otros usuarios comerciales incluido comercios físicos, por lo que cobra un porcentaje de comisión y tarifa [30].

### 3.13.2. Creación de cuenta en PayPal

Para crear una cuenta se debe elegir una de las dos opciones: Personal o Negocios, como paso siguiente, el registro es con el correo electrónico, posteriormente se valida la cuenta mediante un correo que es enviado al email de registro. También puedes añadir el número de tu tarjeta de crédito de ser el caso para realizar las compras en línea, pero no es obligatorio.

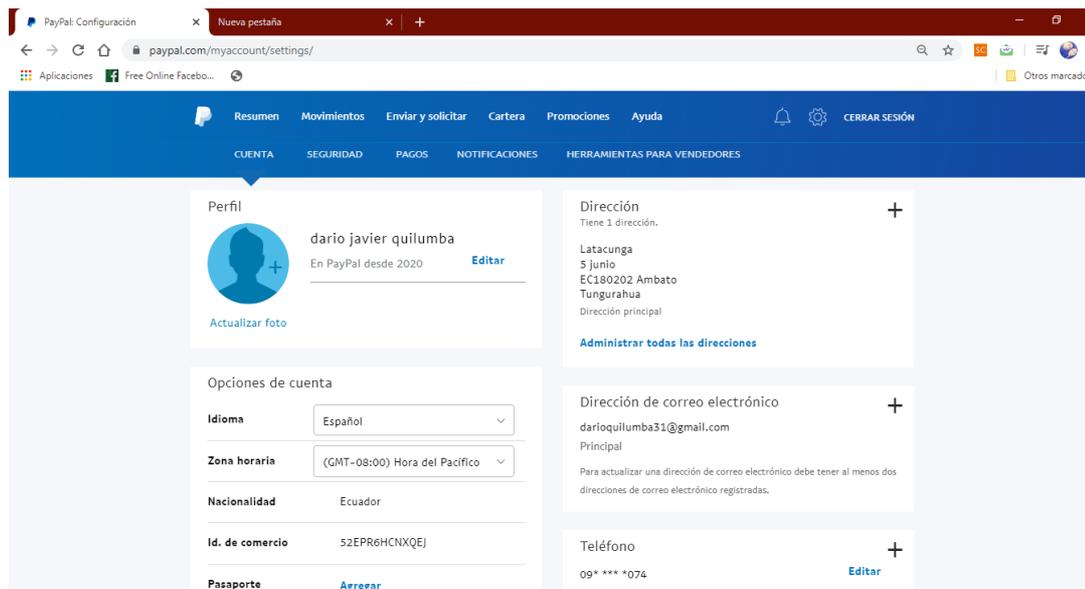
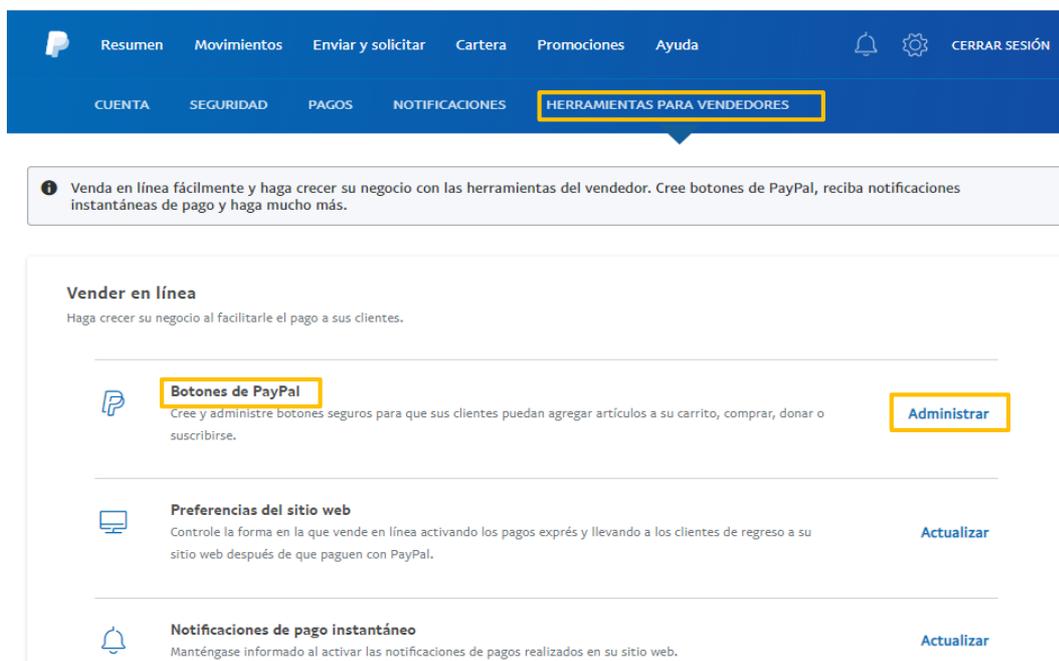


Figura 36: Página de usuario creado en PayPal

En la figura 36, se observa la página del perfil de usuario, si la cuenta fue creada correctamente, además indica todos los datos personales con las que se realizó el registro.

### 3.13.3. Creación del botón de pago en PayPal

Una vez iniciado la sesión con el usuario y la contraseña en PayPal se da clic en el icono de la pantalla principal que dice HERRAMIENTA PARA VENEDORES, seguido aparecen más opciones y se elige Botones de PayPal.



**Figura 37:** Crear botón de pago en PayPal

Escogemos la opción de Administrar y seguido nos dirige a la página siguiente mostrada en la figura 38.



**Figura 38:** Crear botón de pago en PayPal

En esta página podemos observar todos los servicios de los cuales vamos a recibir un pago en línea, también ejemplos prácticos de cómo crear un botón de pago de diferentes tipos, procedemos a dar clic en el icono de Crear nuevo botón y vamos a personalizarlo para el servicio del prototipo de la estación de Auto-lavado.

▼ Paso 1: Elegir un tipo de botón e introducir detalles del pago

Escoja un tipo de botón  
 [¿Qué botón debo escoger?](#)

Nota: [Vaya a Mis botones guardados](#) para crear un nuevo botón parecido al existente.

Nombre del artículo  Id. de artículo (opcional)  [¿Qué es esto?](#)

Precio  Divisa  [¿Necesita varios precios?](#)

**Personalizar botón**

Añadir menú desplegable con el precio o la opción  
[Ejemplo](#)

Nombre del menú desplegable (ejemplo: "Colores", "Tallas")

Nombre de opción de menú	Precio	Divisa
<input type="text" value="Lavado Simple"/>	<input type="text" value="1.50"/>	<input type="text" value="USD"/>
<input type="text" value="Lavado Doble"/>	<input type="text" value="2.00"/>	<input type="text" value="USD"/>

[Añadir otra opción](#) [Eliminar opción](#)

**Vista que tendrá el cliente**

Tipos de Lavado

**Figura 39:** Personalización de botón de pago en PayPal

En la figura 39, se observa la configuración básica, se elige el tipo de botón que va a observar el cliente, seguido del nombre del artículo “Estación de Auto-lavado” y el Id de artículo en este campo se escribe cualquier Id no existe restricciones.

Luego se escribe los dos servicios que brinda la estación de Auto-lavado, Lavado Simple con un precio de \$ 1.50 y el Lavado Doble a \$ 2.00, también se debe elegir el tipo de moneda. Posteriormente en la Vista que tendrá el cliente se encuentra el icono personalizado.

Los demás ajustes se indican en el **Anexo 4**.

Una vez realizado todos los ajustes se da clic en crear botón, y los aparece la ventana siguiente mostrada en la figura 40, que contiene el código HTML de toda la configuración realiza y que es utilizada en el desarrollo de las páginas web del prototipo.

Cómo pegar el código del botón en el editor de su sitio web:

El código debe pegarse en la vista "código" donde pueda ver y editar HTML.

1. En el editor de sitio web o página de administración, abra la página donde desea incorporar su botón.
2. Busque una opción para ver o editar HTML.
3. Encuentre la sección de la página donde desea que aparezca el botón.
4. Dé clic con el botón derecho y pegue el código de su botón en el HTML.
5. Guarde y publique la página. (La función de vista previa en su editor quizá no muestre el código de botón correctamente.)
6. Pruebe el botón para asegurarse que quede vinculado con una página de pagos PayPal.

¿Necesita más ayuda? [Haga clic aquí](#) para obtener más información.



Figura 40: Código HTML de botón de pago en PayPal

### 3.13.4. Prueba del botón de pago en PayPal con PHP

Creamos un archivo. php en la Raspberry Pi, con el comando: **sudo nano /var/www/html/paypal.php** y pegamos el código de la figura 40.

Ingresamos la dirección IP de la Raspberry Pi, seguido del nombre del archivo creado en PHP de la siguiente manera: **192.168.1.200/paypal.php** y observamos la siguiente página.

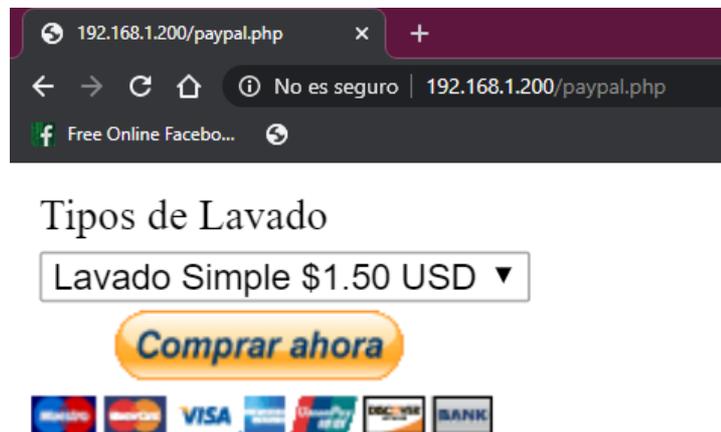
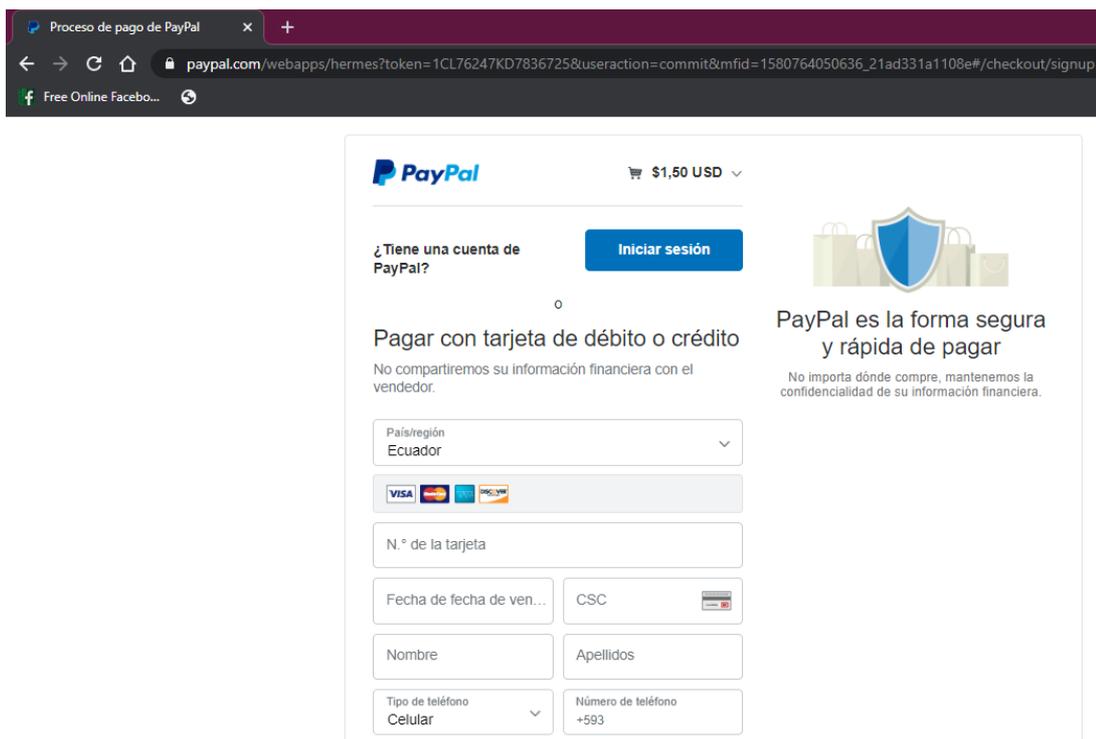


Figura 41: Página web de pago automático en PayPal

El cliente selecciona el tipo del lavado con el costo a pagar y da clic en Comprar ahora y se redirige a la siguiente página que se observa en la figura 42.



**Figura 42:** Pago del servicio de lavado en PayPal

En esta página el usuario, el propietario del vehículo paga por el servicio de lavado, ingresa los datos personales de su tarjeta de debito o crédito, también puede pagar con la cuenta de PayPal claro si tiene dinero en dicha cuenta.

### **3.14. Comunicación del PLC S7-1200 con la Raspberry Pi**

Mediante el lenguaje de programación Python, se instaló la librería Snap 7 en la raspberry Pi, que permite la escritura y lectura de datos del PLC s7-1200 de Siemens. Esta comunicación se denomina como protocolo Ethernet S7. La presente comunicación está configurada con tres componentes: cliente, servidor y compañero.

Las ventajas de usar la librería Snap 7 son varias, debido a que está escrita en C++, para leer los datos del PLC compatibles con Ethernet, mientras las solicitudes a Ethernet no estén restringidas. En el trabajo realizado, los datos se escriben y se leen a través de variables de tipo bits, Word y Dword.

A continuación, en la Figura 43, se indica la condición del cliente Snap7 para escribir datos de tipo bit en el código base que se ejecutó en la raspberry Pi en el programador Python. Todo el código realizado para escribir datos se puede observar en el **Anexo 1**.

```
pulsador_start.py *x
1  from time import sleep
2  import snap7.client
3  from snap7.util import *
4  from snap7.snap7types import *
5
6  myplc = snap7.client.Client()
7  myplc.connect('192.168.0.7',0,1)
8  print(myplc.get_connected())
9
10 def ReadMemory(p,byte,bit,datatype):
11     result=p.read_area(areas['MK'],0,byte,datatype)
12     if datatype == S7WLBit:
13         return get_bool(result,0,1)
```

Figura 43: Código Python para escritura de datos.

Para la lectura de datos del PLC s7-1200, se utilizó datos de salidas, contadores y temporizadores, que utilizan variables de tipo Word, Dword y Bit. La estructura del cliente es la misma que la escritura de datos. El código programado se puede observar en el **Anexo 2**.

```
6  def ReadOutput(dev,bytebit):
7  byte,bit=bytebit.split('.')
8  byte,bit=int(byte),int(bit)
9  data = dev.read_area(0x82,0,byte,1)
10 status = get_bool(data,byte,bit)
11     return status
12
13 def ReadMemory(p,byte,bit,datatype):
14     result=myplc.read_area(areas['MK'],0,byte,datatype)
15     if datatype == S7WLBit:
16         return get_bool(result,0,bit)
17     elif datatype == S7WLByte or datatype == S7WLWord:
18         return get_int(result,0)
```

Figura 44: Código Python para lectura de datos.

### 3.15. Programación de las páginas web con PHP

PHP es un lenguaje de código abierto, especialmente utilizado en el desarrollo de entornos y páginas web, con la facilidad de ser incrustado en HTML. Permite generar páginas con contenido dinámico y trabajar con base de datos.

Para el control y el monitoreo del prototipo de la estación de Auto-lavado se diseñó y programo 5 archivos.php.

La función principal del primer archivo **index.php** es, dar a elegir al usuario si desea ingresar o registrarse para el servicio. El código de programación se muestra en el **Anexo 5**.

En la página web del segundo archivo programado **signup.php** el usuario se registra y todos los datos ingresados son guardados en la base de datos **phpMyAdmin**. El código se puede observar en el **Anexo 6**.

```
signup.php x
1 <?php require 'database.php'; $message = ''; if (!empty($_POST['email']) &&
2 !empty($_POST['password'])) {
3     $sql = "INSERT INTO usuarios (email, password) VALUES (:email, :password)";
4     $stmt = $conn->prepare($sql);
5     $stmt->bindParam(':email', $_POST['email']);
6     $password = password_hash($_POST['password'], PASSWORD_BCRYPT);
7     $stmt->bindParam(':password', $password);
8     if ($stmt->execute()) {
9         $message = 'Successfully created new user';
10    } else {
11        $message = 'Sorry there must have been an issue creating your account';
12    }
13 }
14 ?>
```

**Figura 45:** Código para guardar el registro en base de datos

En la figura 45, se indica el código con que se realiza el llamado a la base de datos y se guarda el registro de los usuarios, tanto el campo de email como la contraseña. Además de existir un problema de conexión indica un mensaje de advertencia que no se ha creado correctamente la cuenta.

El siguiente archivo **database.php** contiene los datos del servidor para ingresar a la base de datos de **PhpMyAdmin**.

```
database.php x
1 <?php
2 $server='localhost';
3 $username='root';
4 $password='12345678';
5 $database='php_login_database';
6
7 try {
8     $conn=new PDO("mysql:host=$server;dbname=$database;", $username, $password);
9
10 } catch (PDOException $e){
11
12 die('Connected failed: '.$e->getMessage());
13 }
14
15 ?>
```

**Figura 46:** Código de conexión con la base de datos

En la figura 46, se puede observar la configuración del usuario, contraseña y la carpeta en la que se guarda los datos de registro de los usuarios.

El archivo **login.php** es el que da acceso al usuario una vez que se haya culminado el proceso de registro, realiza una comparación entre los datos de acceso y los de la base de datos y si son iguales da la bienvenida el sistema al usuario, de lo contrario no permite el acceso, como se observa en la figura 47. El código se indica en el **Anexo 7**.

```
11  if (!empty($_POST['email']) && !empty($_POST['password'])) {
12      $records = $conn->prepare('SELECT id, email, password FROM usuarios WHERE email = :email');
13      $records->bindParam(':email', $_POST['email']);
14      $records->execute();
15      $results = $records->fetch(PDO::FETCH_ASSOC);
16      $message = '';
17      if (count($results) > 0 && password_verify($_POST['password'], $results['password'])) {
18          $_SESSION['user_id'] = $results['id'];
19          header("Location: /login");
20      } else {
21          $message = 'Sorry, those credentials do not match';
22      }
23  }
```

**Figura 47:** Código de acceso al sistema de Auto-lavado

Los otros dos archivos: **logout.php** y **styles.css** se programó para cerrar la sesión y el otro es del estilo, fuente, tipo de texto, color, margen, etc. Para que el diseño web sea agradable para el cliente. El código se muestra en el **Anexo 8** y **Anexo 9** respectivamente.

### 3.15.1. Escritura de datos del PLC mediante página web

Una vez programado en Python los scripts, para escribir datos de tipo bits para los diferentes pulsadores tanto de emergencia, paro y marcha en el PLC s7-1200 ver **Anexo 1**, se procede a importar todos los scripts en el código de PHP del archivo **s7\_1200.php**.

```
42  </body>
43  </html>
44  <?php
45  // Funciones PHP del pin GPIO 7
46
47  if ($_POST[Emergencia]) {
48      $a- exec('sudo python /home/pi/Tesis/pulsador_emergencia.py');
49      echo $a;
50  }
51  if ($_POST[Stop]) {
52      $a- exec('sudo python /home/pi/Tesis/pulsador_stop.py');
53      echo $a;
54  }
```

**Figura 48:** Importar scripts de Python en PHP

Para importar el archivo de programación en el código de PHP se debe ingresar la dirección en la que se encuentra el script de Python en la Raspberry Pi, como se observa en la figura 48.

El código del archivo **s7\_1200.php** se indica en el **Anexo 10**.

### 3.15.2. Lectura de datos del PLC mediante página web

Los datos son leídos sin la necesidad de recargar la página para esto se utiliza la herramienta de Jquery, que permite que los datos se actualicen cada cierto tiempo de acuerdo a la programación.

```
12     <script
13     src="https://code.jquery.com/jquery-3.2.1.js"
14     integrity="sha256-DZAnKJ/6XZ9si04Hgrsxu/8s717jcIzLy3oi35EouyE="
15     crossorigin="anonymous"></script>
```

**Figura 49:** Actualizar datos con Jquery

El código de Jquery está presente en la página oficial de Jquery CDN, que de acuerdo a la versión a utilizar se selecciona y se ingresa directamente al código de programación de PHP, como se muestra en la figura 49.

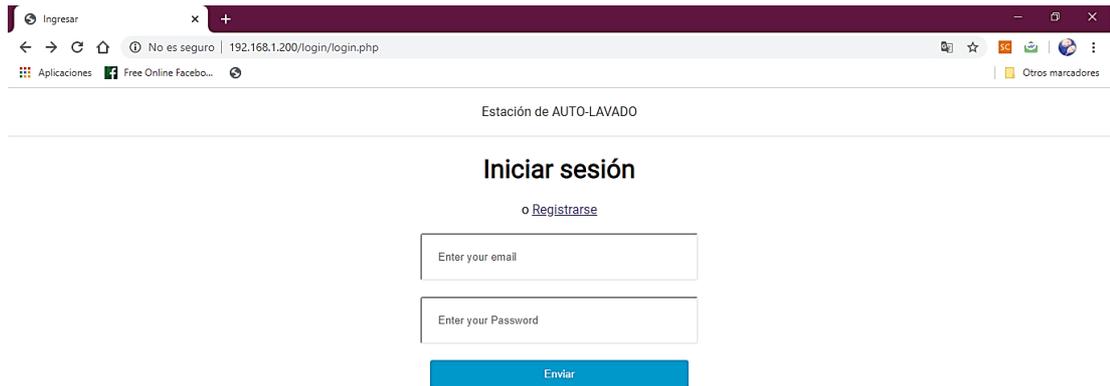
```
30 <script type="text/javascript">
31     $(document).ready(function(){
32         setInterval(
33             function(){
34                 $('#seccionRecargar').load('leerQs.php');
35             },700
36         );
37     });
38 </script>
```

**Figura 50:** Tiempo de actualización de datos con Jquery

En la figura 50, se observa el tiempo en el que los datos de lectura del PLC s7-1200 se van a actualizar de manera automática que es de 700 mili segundos. La programación se encuentra en el **Anexo 11**.

### 3.16. Diseño de la página web para control y adquisición de datos

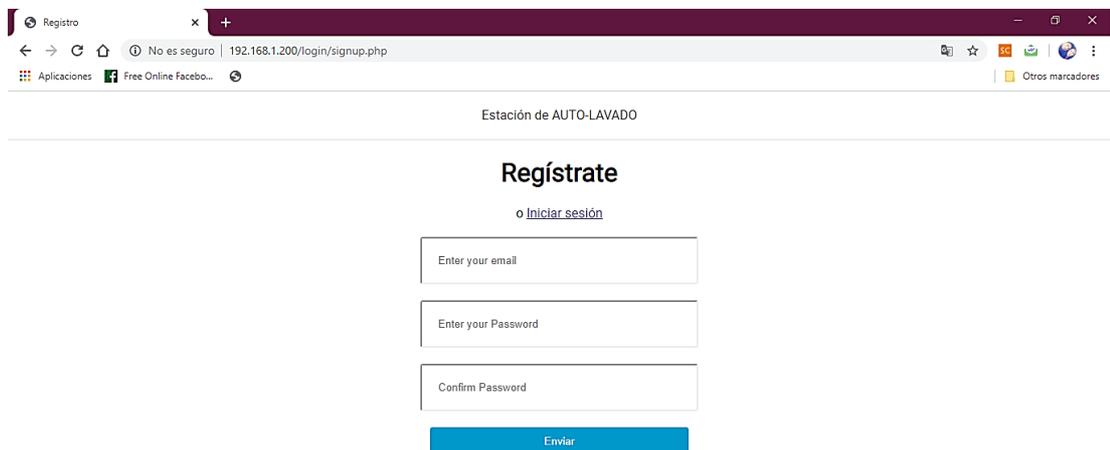
Los resultados obtenidos son las diferentes páginas webs creadas con el lenguaje de programación PHP, que ingresada desde cualquier navegador con la dirección 192.168.1.200/login muestra la pantalla de validación de los usuarios una vez que se hayan registrado, como muestra la figura 51.



**Figura 51:** Página web para acceder al servicio

### 3.16.1. Página para el registro de usuarios

En esta página los usuarios proceden a su registro para después acceder al servicio.

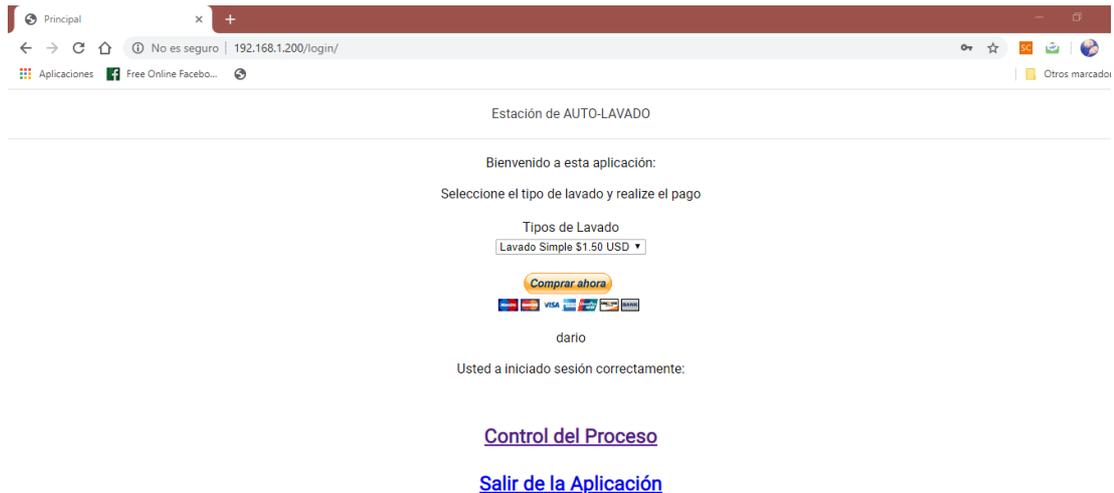


**Figura 52:** Página web de registro

La página muestra tres campos: el correo, contraseña y la confirmación de la contraseña, una vez llenado los campos de manera correcta, se muestra un mensaje que su registro se realizó de manera correcta.

### 3.16.2. Página de bienvenida y pago automático

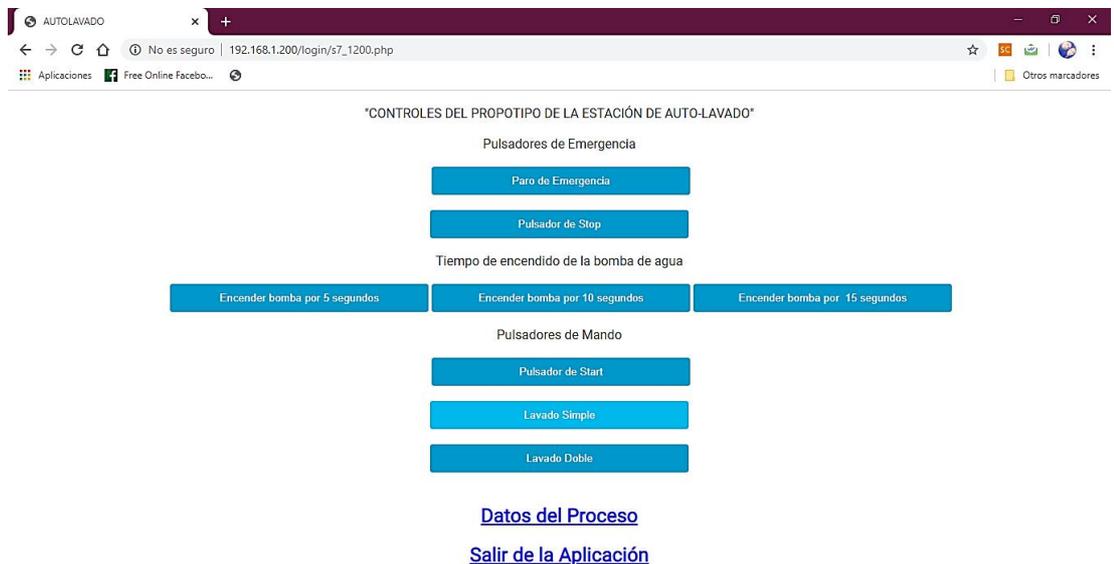
Una vez iniciada la sesión por parte del cliente, es re direccionado a la página mostrada en la figura 53, en donde el usuario selecciona el tipo de lavado y procede a realizar el pago. Si el pago fue realizado con éxito es dirigido a la página de control del procedo de Auto-lavado ver figura 54, si el usuario realizado el pago de lavado simple, entonces el botón de lavado doble no aparecerá en la página, lo mismo sucede para el pago de lavado doble.



**Figura 53:** Página web de bienvenida y pago automático

### 3.16.3. Página de control del proceso de Auto-lavado

En la figura 54, se observa los controles de la estación de auto-lavado, contiene todos los pulsadores tanto de marcha, emergencia y paro. El usuario debe seleccionar el tiempo de encendido de la bomba de agua, posteriormente encender el sistema con el pulsador de Start y por ultimo seleccionar el tipo de lavado que requiere en el servicio.



**Figura 54:** Página web para el control del auto-lavado

### 3.16.4. Página de monitorización de variables

La última página web muestra los datos digitales de los estados de las salidas del PLC Siemens s7-1200, que están conectadas a los distintos actuadores dentro del sistema de lavado, el estado de cada salida cambiara de True = Encendido a False = Apagado,

también cuenta con un contador que muestra el número de autos lavados al final de cada día.

## Datos de los actuadores del sistema

Sistema encendido: True  
Motor adelante : True  
Bomba de agua : False  
Electro - valvulas : False  
Motor regreso : False  
Rodillos verticales: False  
Rodillos horizontal: False  
Tiempo transcurrido: 15000  
N de Autos lavados: 8

[Atras](#)

[Salir de la Aplicación](#)

Figura 55: Página web de monitorización

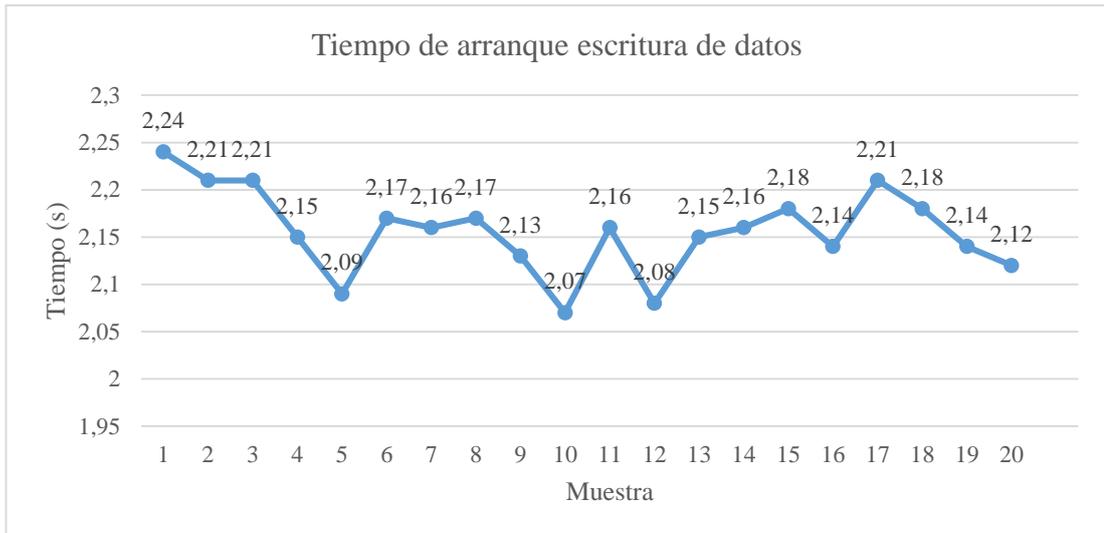
### 3.17. Análisis y discusión de resultados

En esta parte final del proyecto de investigación se detallan los resultados obtenidos al momento de realizar las pruebas de funcionamiento, se realizó un análisis con una muestra de tiempos del arranque del sistema del prototipo de la estación de Auto-lavado específicamente del controlador PLC s7-1200 al momento de enviar y recibir datos de la página web diseñada, además se indica el desempeño de la CPU de la Raspberry Pi en el momento en el que se está ejecutando todo el sistema.

#### 3.17.1. Análisis del arranque de escritura de datos

Se tomó un total de 20 muestras de los tiempos de arranque de la escritura de datos, el tiempo se tomó desde que se manda a ejecutar el proceso desde la página web, hasta que la plataforma del prototipo se comience a desplazar de manera lineal, que es accionado por el motor delantero.

En la figura 56, se indica la muestra de los tiempos tomados al momento de ejecutar el arranque del prototipo.



**Figura 56:** Tiempo de arranque de escritura de datos

### Media aritmética

$$\bar{x} = \frac{\sum_{i=1}^n X_i}{n}$$

**Donde:**

$n$  = Número de muestras

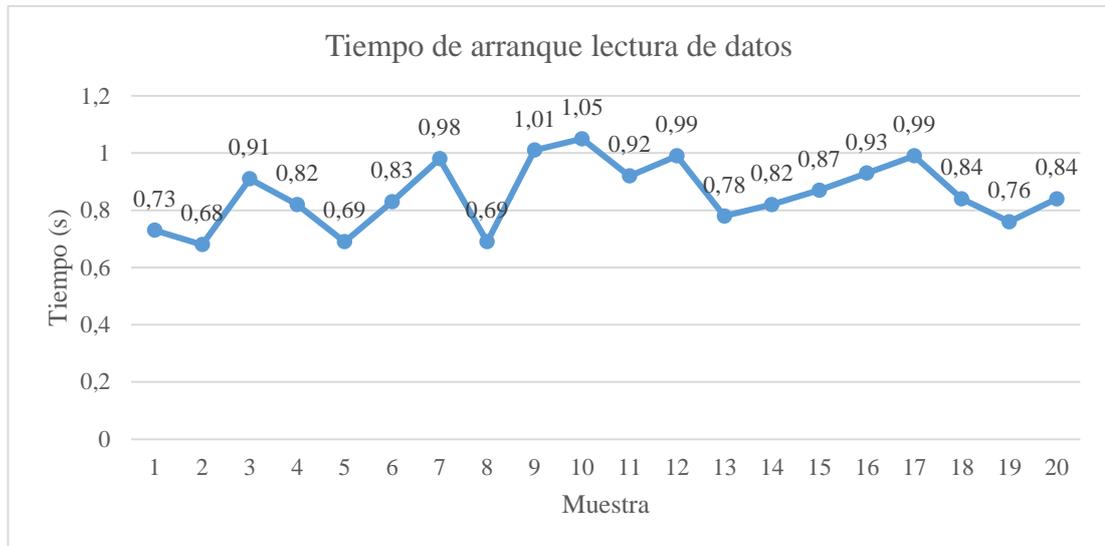
$X_i$  = Valores de cada muestra

$\bar{x}$  = Media aritmética

$$\bar{x} = 2.16 \text{ (seg)}$$

El tiempo promedio estimado que se requiere para el arranque de la escritura de datos, desde la página web hasta el PLC s7-1200 es de 2.16 segundos, considerando este tiempo como rápido, esta velocidad también es obtenida debido a las diferentes características de brinda la Raspberry Pi 3B, que cuenta con un puerto ethernet que puede transmitir con una velocidad de 300 Mbps (Megabits/ segundo) teóricamente, además posee un procesador de 1.4 GHz y memoria RAM de 1 GB.

### 3.17.2. Análisis del arranque de lectura de datos



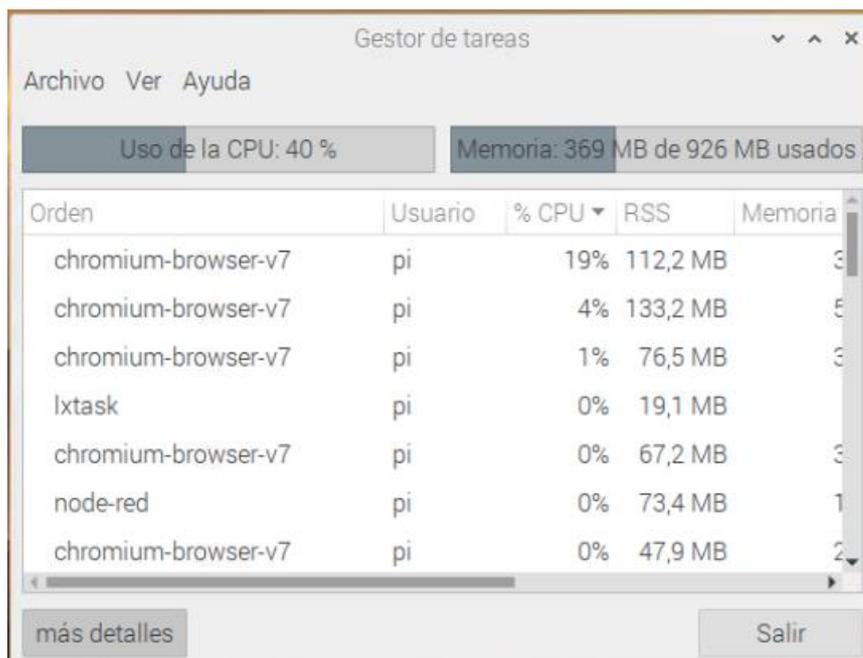
**Figura 57:** Tiempo de arranque de lectura de datos

$$\bar{x} = 0.86 \text{ (seg)}$$

El tiempo de lectura de datos de las salidas del PLC s7-1200 que son conectadas hacia los distintos actuadores es de 0.86 segundos o de 860 milisegundos, que es una velocidad de lectura muy rápida al momento de leer el script de Python desde la página web programado con PHP, además de acuerdo con lo programado en JQuery que permite la actualización de datos sin recargar la página, que fue de 700 milisegundos teóricamente, existe un desfase de 186 milisegundos al momento de realizar las pruebas en conjunto página web-PLC.

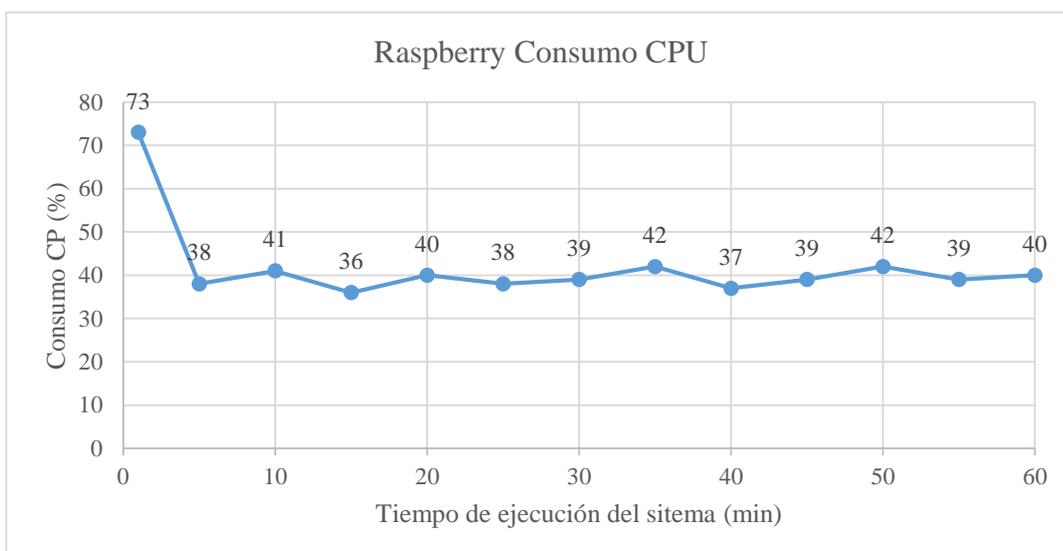
### 3.17.3. Desempeño de la Raspberry Pi con el sistema en ejecución

Una vez puesta en marcha la Raspberry Pi realiza la ejecución de las páginas web programadas (registro, ingreso, pago automático, control del proceso, monitorización) desde la cual se puede enviar y recibir datos por el protocolo Ethernet S7. También el sistema de alarmas por SMS en Node-red se está ejecutando, en la figura 58 se observa el rendimiento de la CPU después de 1 hora (60 minutos), de poner en Marcha el sistema del Auto-lavado.



**Figura 58:** Consumo de la CPU en la Raspberry Pi

El rendimiento de la CPU de la Raspberry Pi varía entre el 36% a 42%, ocupando así un porcentaje moderado en la ejecución del sistema permitiendo fluidez y rapidez en la ejecución y recepción de datos. Existe un pico alto con un porcentaje de consumo del 73% al minuto 1, debido a que todos los dispositivos en este caso la Raspberry Pi al momento del arranque requieren mayor consumo hasta que el sistema llegue a un nivel de estabilidad. En la figura 59 se observa el consumo del CPU.



**Figura 59:** Grafica consumo de la CPU en la Raspberry Pi

## CAPÍTULO IV

### CONCLUSIONES Y RECOMENDACIONES

#### 4.1. Conclusiones

- Se seleccionó el controlador lógico programable Siemens s7-1200 1214 ac/dc/dry, para el desarrollo de esta investigación, ya que posee un mayor número de entradas y salidas digitales, y es ideal para una amplia gama de aplicaciones de automatización, además la comunicación entre la pantalla KTP 600 se realizó con el mismo software de programación, también la conexión con la Raspberry Pi se ejecutó vía ethernet y se obtuvo mayor velocidad en la transición de datos
- La interacción entre clientes en este caso propietarios de los vehículos y el sistema implementado de pago automático mediante el funcionamiento del prototipo de la estación de Auto-lavado se logró a través del diseño y programación tanto del PLC, como de la pantalla HMI y de las páginas web por intermedio del protocolo de comunicación ethernet S7 con la Raspberry Pi, mismas que resultan necesarias para el control y monitoreo del proceso de lavado de autos.
- El rendimiento del mini ordenador Raspberry Pi es muy eficiente en la implementación y en la ejecución de todo el sistema, teniendo en cuenta que ejecuta varias tareas al mismo tiempo, dando como resultado un consumo de la CPU del 40%, la velocidad de respuesta en la escritura y lectura de datos desde que se envía una orden de la página web hasta el PLC, es de 2.16 segundos y de 860 milisegundos respectivamente, esta velocidad es obtenida gracias a las grandes características que posee este modelo de mini ordenador.
- Se consiguió implementar el pago automático para el servicio de lavado de autos en el prototipo, mediante la creación y configuración de un botón de pago inteligente en la plataforma de pagos en línea PayPal, además el mensaje SMS de alerta que es enviado al celular se realizó a través de la liberación de un número propio de Twilio plataforma de comunicación y mensajería y también de la programación en Node-Red.

## 4.2. Recomendaciones

- La primera configuración que se debe realizar después de instalar el sistema operativo raspbian en la Raspberry Pi es la de configurar una IP estática con el comando `sudo nano /etc/dhcpd.conf` para una red de tipo eth0 con una dirección IP que se encuentre en la misma subred que la dirección del PLC.
- Si existe algún error en la comunicación entre la Raspberry Pi y el PLC debe desinstalar e instalar de nuevo la librería de Snap 7, para el desarrollo de este proyecto se utilizó la versión `snap7-full-1.4.2` y no presento ningún problema en su utilización.
- Para verificar si el botón de pago inteligente se configuro de manera correcta, se debe crear primero un archivo `.php` en la Raspberry Pi, y colocar el código HTML del botón, y por ultimo ejecutar el servidor web con archivo utilizando cualquier navegador.
- Para un mejor desempeño de la Raspberry PI, se sugiere utilizar ventiladores o disipadores de calor en el mejor de los casos, o colocar el dispositivo en lugares ventilados y libre del polvo, y para proteger el hardware colocar un case. Tener en cuenta que se están ejecutando varios procesos a la vez.

## BIBLIOGRAFÍA

- [1] M. López, «Industria 4.0 para la monitorización de un proceso industrial,» Universidad Técnica de Ambato, Amabto, 2019.
- [2] P. Esteban, «Los sistemas SCADA en la automatización industrial,» *Tecnología en Marcha*, vol. 28, nº 4, pp. 5-6, 2015.
- [3] J. Villacrés, «Interfaz Hombre – Máquina para el monitoreo y control de los Sistemas de Automatización de la Planta de Producción Acería en la Empresa NOVACERO S.A,» Universidad Técnica de Ambato, Ambato, 2015.
- [4] O. Analuisa, «Sistema electrónico para el monitoreo del almacenamiento de diesel del hostipal general de Ambato,» Universidad Técnica de Ambato, Ambato, 2018.
- [5] S. Bustos, «Desarrollo de un sistema industrial basado en el estándar IEC-61499.,» Universidad Técnica de Ambato, Ambato, 2017.
- [6] O. Gloria, «Sistema de control automatico del proceso de llenado de botellas de agua en la planta Purificadora Ecoagua,» Universidad Técnica de Ambato, Ambato, 2018.
- [7] T. Gallo, «Desarrollo e implementación de internet industrial de las cosas aplicada al laboratorio de PLC'S de la Facultad de Ingeniería Industrial en Procesos de Automatización de la Universidad Técnica de Ambato,» Escuela Superior Pilotécnica de Chimborazo, Riobamba, 2018.
- [8] M. Molina, «Sistema SCADA para la supervisión en tiempo real de medidores industriales de energía en la Empresa Novacero S.,» Universidad Técnica de Ambato, Ambato, 2019.
- [9] R. Villena, «Proyecto de integración de autómatas programables Simatic s7-1200 en red Iot.,» Universidad Politécnica de Valencia, Valencia, 2018.

- [10] W. García, «Sistema de control y monitoreo para bombas de drenaje y vaciado de la Central,» Universidad Técnica de Ambato, Ambato, 2017.
- [11] W. Collay, Luzuriaga y Julio, «Reconstrucción y automatización de una máquina envasadora de agua en la corporación BImarch CIA. LTDA. ubicada en la parroquia Cotaló en el cantón Pelileo,» Escuela Superior Politécnica de Chimborazo, Riobamba, 2017.
- [12] O. Altamirano, «Parametrización automática de las variables de control y su influencia en el proceso de mezclado de compuestos termoplásticos aplicando principios de industria 4.0,» Universidad Técnica de Ambato, Ambato, 2018.
- [13] A. Veintimilla y C. Yunga, «Diseño e implementación de un prototipo de control domótico de bajo costo activado por voz para personas con discapacidad motriz,» Escuela Superior Politécnica de Chimborazo, Riobamba, 2018.
- [14] C. Escobar, «Diseño de sistemas de control industrial de robots basado en industria 4.0,» Universidad Técnica de Ambato, Ambato, 2019.
- [15] F. Baño, «Sistema empotrado de bajo costo para adquisición de datos basados en el estándar de comunicación industrial OPC-UA,» Universidad Técnica de Ambato, Ambato, 2019.
- [16] E. Mifsut y M. Medina, «Curso Mentor Apache,» Gobierno de España, España, 2017.
- [17] J. Torre y R. Burgos, «Sistema web para la gestión y administración de anteproyectos y tesis de grado,» Universidad Politécnica Salesiana sede Guayaquil, Guayaquil, 2016.
- [18] D. Quicaliquín, «implementación de un prototipo: sistema automático de censado inalámbrico basado en rfid para estaciones de control de buses urbanos de riobamba,» Escuela Superior Politécnica de Chimborazo, Riobamba, 2018.

- [19] R. Noguera y L. MinAqua, «Guía de buenas prácticas para instalaciones de lavado de vehículos,» Glam Comunicació S.L., Barcelona , 2016.
- [20] E. Global, «Estaciones de servicio S.L.,» 26 Enero 2016. [En línea]. Available: <https://globalestacionesdeservicio.com/el-proceso-de-lavado-en-un-puente-automatico/#respond>. [Último acceso: 09 Diciembre 2019].
- [21] A. Dansa, «Lavado coches túnel,» Autolavado Aragón, S.L, España, 2017.
- [22] D. Clean Car, «Clean-car Sistema de autolavado automatico. Buenos Aires Argentina.,» 2019. [En línea]. Available: <https://www.clean-car.com.ar/>. [Último acceso: 30 Octubre 2019].
- [23] S. Eyna Car Wahs, «Túneles de lavado para vehículos industriales en alta presión,» Interempresas Media, S.L.U. - Grupo Nova Àgora, 2019. [En línea]. Available: <http://www.interempresas.net/Estaciones-servicio/FeriaVirtual/Producto-Lavados-sin-contacto-Kondor-159615.html>. [Último acceso: 30 Octubre 2019].
- [24] S. Eyna car wash, «Lavado sin contacto Saber SL1,» Madrid – España, 2019. [En línea]. Available: <https://www.eyna.eu/lavado-sin-contacto-saber-sl1/>. [Último acceso: 30 Octubre 2019].
- [25] N. Talavera, «Consultores T.I Servicios Informáticos para la Empresa,» 04 Septiembre 2019. [En línea]. Available: <https://www.inetalavera.es/cashdro-sistemas-de-cobro-automatico>. [Último acceso: 09 Diciembre 2019].
- [26] E. Telematel, «ERP Telematel Gestor de Cobro,» 10 Octubre 2019. [En línea]. Available: <https://www.telematel.com/distribucion/soluciones/terminal-cobro-efectivo/>. [Último acceso: 09 Diciembre 2019].

- [27] P. Burget, «BURGUETSISTEMAS S.L.,» 2019. [En línea]. Available: <http://www.burguet.es/productos/cobro/>. [Último acceso: 09 Diciembre 2019].
- [28] G. Cashkontrol, «Copyright c 2019 Máquinas de Cobro Automático,» 2019. [En línea]. Available: <https://cashkontrol.com/cashlogy/>. [Último acceso: 09 Diciembre 2019].
- [29] C. Twilio, «Twilio - API de comunicación para SMS, voz y video en sus aplicaciones web y moviles.,» 2020. [En línea]. Available: <https://www.twilio.com/what-is-cloud-communications>. [Último acceso: 03 Enero 2020].
- [30] F. Ortiz, «PayPal mejora su acuerdo con Mastercard para llegar a las tiendas físicas,» 07 Septiembre 2017. [En línea]. Available: <https://www.muycomputerpro.com/2016/09/07/paypal-mastercard-tiendas-fisicas>. [Último acceso: 02 Enero 2020].

## ANEXOS

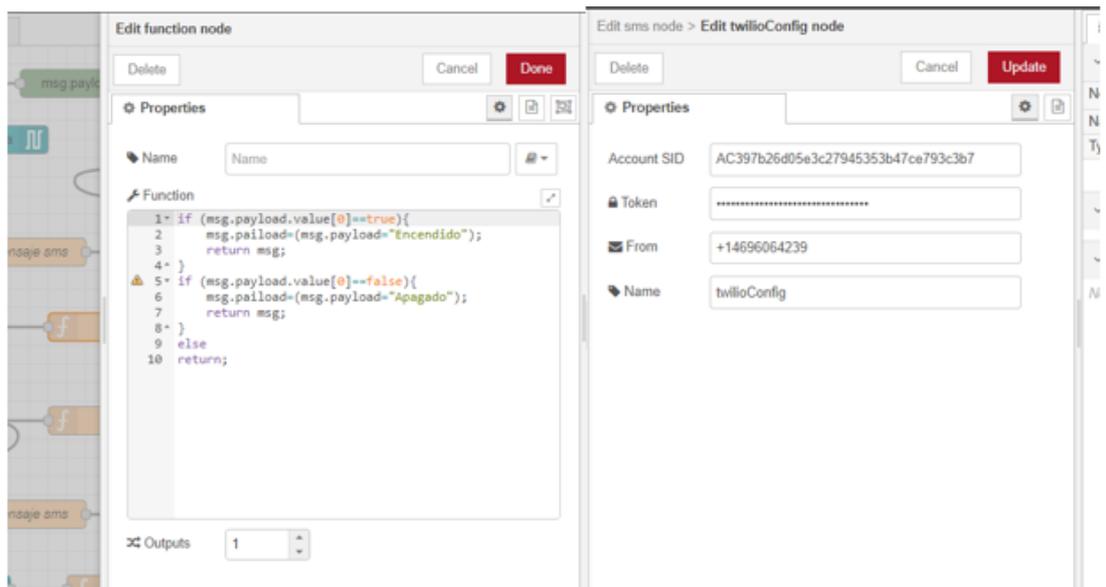
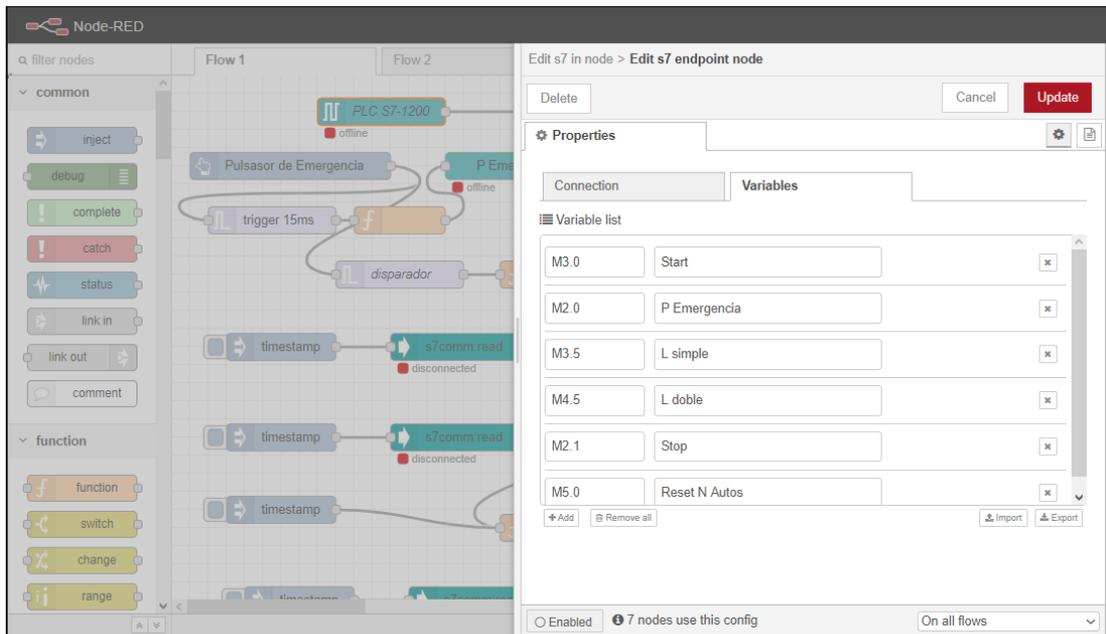
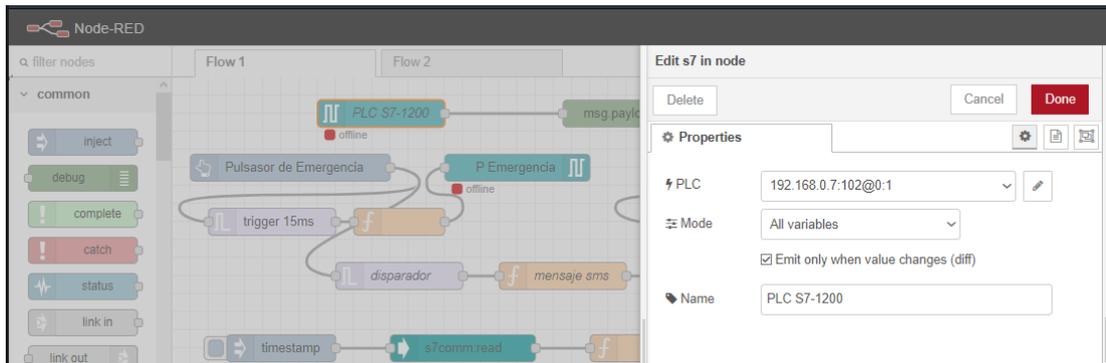
### Anexo 1. Código en Python para escritura de datos en PLC s7-1200.

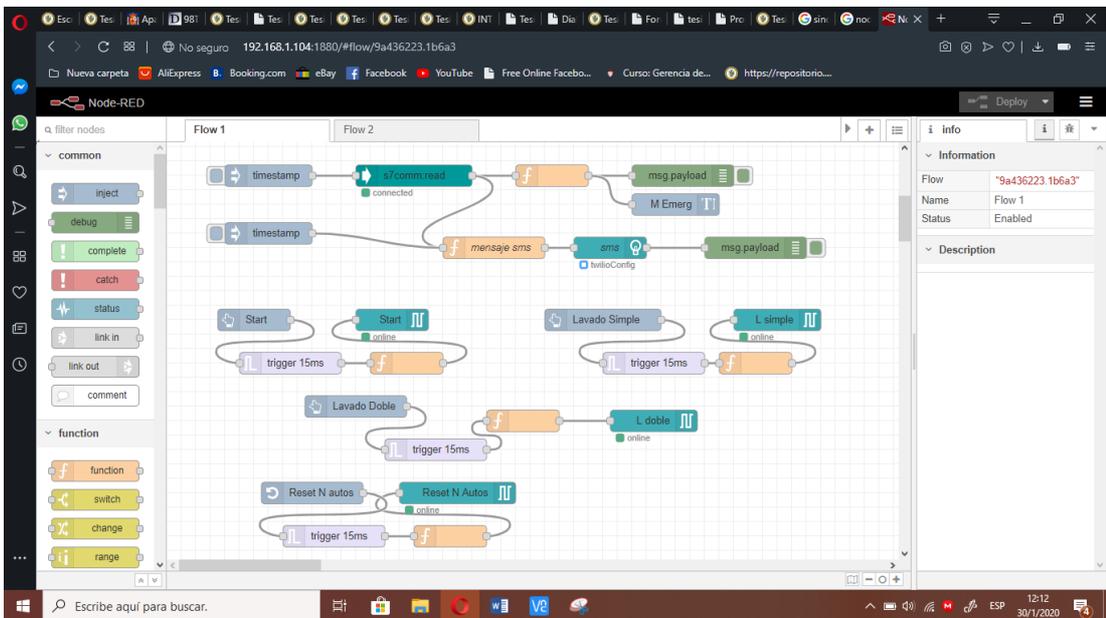
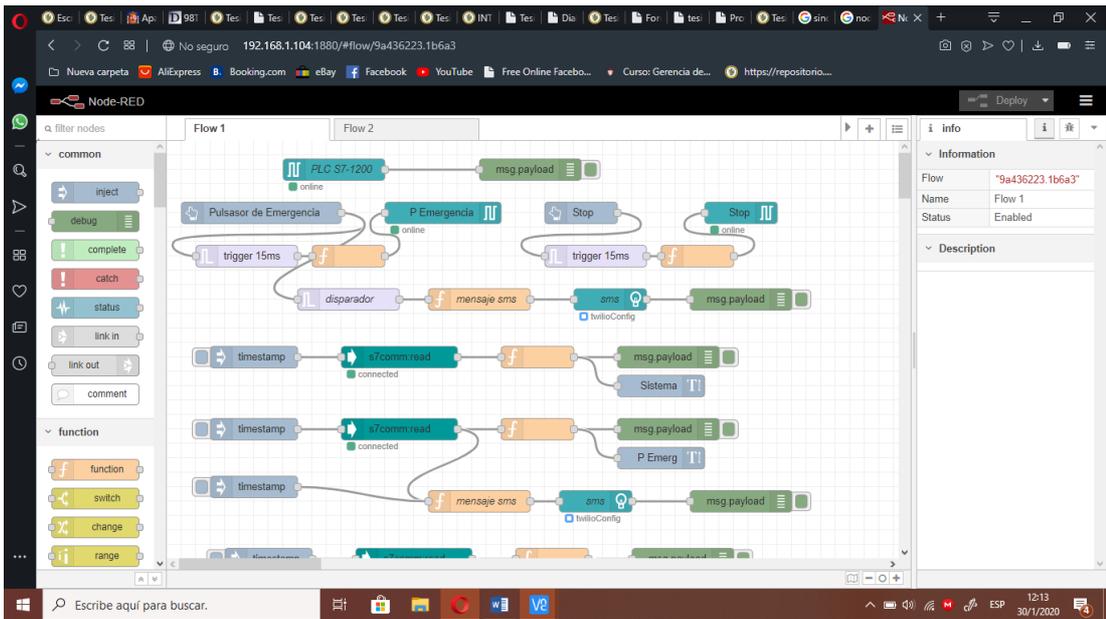
```
pulsador_start.py ✕
1 from time import sleep
2 import snap7.client
3 from snap7.util import *
4 from snap7.snap7types import *
5 |
6 def ReadMemory(p,byte,bit,datatype):
7     result=p.read_area(areas['MK'],0,byte,datatype)
8     if datatype == S7WLBit:
9         return get_bool(result,0,1)
10 #     elif datatype == S7WLByte or datatype == S7WLWord:
11 #         return get_int(result,0)
12 #     elif datatype == S7WLReal:
13 #         return get_real(result,0)
14 # elif datatype == S7LDWord:
15 #     return get_dword(result,0)
16     else:
17         return None
18 def WriteMemory(p,byte,bit,datatype,value):
19     result=p.read_area(areas['MK'],0,byte,datatype)
20     if datatype == S7WLBit:
21         set_bool(result,0,bit,value)
22 #     elif datatype == S7WLByte or datatype == S7WLWord:
23 #         set_int(result,0,value)
24 #     elif datatype == S7LDReal:
25 #         set_real(result,0,value)
26 # elif datatype == S7LDWord:
27 #         set_dword(result,0,value)
28     p.write_area(areas["MK"],0,byte,result)
29
30 myplc = snap7.client.Client()
31 myplc.connect('192.168.0.7',0,1)
32 print(myplc.get_connected())
33 WriteMemory(myplc,3,0,S7WLBit,True)
34 sleep(0.5)
35 WriteMemory(myplc,3,0,S7WLBit,False)
```

## Anexo 2. Código en Python para lectura de datos en PLC s7-1200.

```
leerQ5sistema.py *  
1 from time import sleep  
2 import snap7.client as c  
3 from snap7.util import *  
4 from snap7.snap7types import *  
5  
6 def ReadOutput(dev,bytebit):  
7     byte,bit=bytebit.split('.')  
8     byte,bit=int(byte),int(bit)  
9     data = dev.read_area(0x82,0,byte,1)  
10    status = get_bool(data,byte,bit)  
11    return status  
12  
13 def ReadMemory(p,byte,bit,datatype):  
14    result=myplc.read_area(areas['MK'],0,byte,datatype)  
15    if datatype == S7WLBit:  
16        return get_bool(result,0,bit)  
17    elif datatype == S7WLByte or datatype == S7WLWord:  
18        return get_int(result,0)  
19    elif datatype == S7WLReal:  
20        return get_real(result,0)  
21    elif datatype == S7WLDWord:  
22        return get_dword(result,0)  
23    else:  
24        return None  
25  
26 myplc= c.Client()  
27 myplc.connect('192.168.0.7',0,1)  
28 #print(myplc.get_connected())  
29 print "Sistema encendido:"  
30 print ReadOutput(myplc,'0.5')  
31 print "<br>"  
32 print "&nbsp;Motor &nbsp;adelante &nbsp;:"  
33 print ReadOutput(myplc,'0.2')  
34 print "<br>"  
35 print "&nbsp;Bomba &nbsp;de &nbsp;agua&nbsp;:"  
36 print ReadOutput(myplc,'0.4')  
37 print "<br>"  
38 print "&nbsp;Electro&nbsp;-&nbsp;valvulas&nbsp;:"  
39 print ReadOutput(myplc,'0.4')  
40 print "<br>"  
41 print "&nbsp;Motor &nbsp;&nbsp;&nbsp;regreso&nbsp; &nbsp;:"  
42 print ReadOutput(myplc,'0.0')  
43 print "<br>"  
44 print "Rodillos verticales:"  
45 print ReadOutput(myplc,'0.1')  
46 print "<br>"  
47 print "Rodillos horizontal:"  
48 print ReadOutput(myplc,'0.1')  
49 print "<br>"  
50 print "Tiempo transcurrido:"  
51 print ReadMemory(myplc, 20, 0, S7WLDWord)  
52 print "<br>"  
53 print "N de Autos lavados:"  
54 print ReadMemory(myplc, 40, 0, S7WLDWord)
```

### Anexo 3. Flujo de programación del mensaje de alarma.





## Anexo 4. Configuraciones del botón de pago en PayPal

The image shows a series of screenshots from a web browser displaying the PayPal Seller Tools interface. The top screenshot shows the main dashboard with navigation tabs like 'Resumen', 'Movimientos', 'Enviar y solicitar', 'Cartera', 'Promociones', and 'Ayuda'. Below this, there's a section titled 'Vender en línea' (Sell online) with three main options: 'Botones de PayPal' (PayPal buttons), 'Preferencias del sitio web' (Website preferences), and 'Notificaciones de pago instantáneo' (Instant payment notifications).

The middle screenshot shows the 'Mis botones guardados' (My saved buttons) page. It features a table of saved buttons and a sidebar with 'Artículos relacionados' (Related items). The table has columns for 'Nombre del artículo' (Item name), 'Cant. disponible' (Available quantity), 'Precio' (Price), and 'Acción' (Action). The buttons listed include 'Auto-lavado', 'Estación de Auto-lavado', 'Botón Comprar ahora de ejemplo', 'Ejemplo de Botón Añadir al carrito', and 'Botón de suscripción de ejemplo'. The 'Acción' column contains dropdown menus with 'Acción' selected.

The bottom screenshot shows the configuration form for a button. It is titled 'Paso 1: Elegir un tipo de botón e introducir detalles del pago' (Step 1: Choose a button type and enter payment details). The form includes fields for 'Nombre del artículo' (Estación de Auto-lavado) and 'Id. de artículo (opcional)' (ES05). Under 'Personalizar botón' (Customize button), there are checkboxes for 'Añadir menú desplegable con el precio o la opción' (Add dropdown menu with price or option) and 'Añadir campo de texto' (Add text field). The 'Vista que tendrá el cliente' (View the customer will see) section shows a preview of the button with a 'Buy Now' button and a price of \$1.50 USD.

Editar botón de pago de PayPal

paypal.com/ec/cgi-bin/webscr?cmd=\_flow&SESSION=5iqY6tz9HSqjgFwQu8y17KoJADxtLIY9eFyrP9KL...y5fh3ACN38nu73lsK&dispatch=5885d80a13c0db1f8...

Nombre de opción de menú	Precio	Divisa
Lavado Simple	1,50	USD
Lavado Doble	2,00	USD

[Añadir otra opción](#) [Eliminar opción](#)

Añadir menú desplegable [Ejemplo](#)

Añadir campo de texto [Ejemplo](#)

[Personalizar texto o aspecto \(opcional\)](#)

Envío

Utilizar importe específico:  USD [divisa](#)

Impuesto:

Utilizar tasa de impuesto  %

Identificación de cuenta de comercio [Más información](#)

Usar mi Id de cuenta de comercio seguro

Utilizar mi dirección de correo electrónico principal [darioquilumba31@gmail.com](mailto:darioquilumba31@gmail.com)

[Paso 2: Realizar un seguimiento del inventario, de las ganancias y de las pérdidas \(opcional\)](#)

[Paso 3: Personalizar funcionalidades avanzadas \(opcional\)](#)

[Quiénes somos](#) [Cuentas](#) [Comisiones](#) [Privacidad](#) [Centro de seguridad](#) [Contactar](#) [Contratos legales](#)

Copyright © 1999-2020 PayPal. Todos los derechos reservados.  
Nota para el usuario: PayPal Fin. Ltd., titular de la instalación de valor

Editar botón de pago de PayPal

paypal.com/ec/cgi-bin/webscr?cmd=\_flow&SESSION=5iqY6tz9HSqjgFwQu8y17KoJADxtLIY9eFyrP9KL...y5fh3ACN38nu73lsK&dispatch=5885d80a13c0db1f8...

**Paso 2: Realizar un seguimiento del inventario, de las ganancias y de las pérdidas (opcional)**

Guardar el botón en PayPal

- Proteger sus botones contra cambios fraudulentos
- Añadir botones automáticamente a "Mis botones guardados" en su perfil de PayPal
- Crear fácilmente botones parecidos
- Editar sus botones con las herramientas de PayPal

Realizar un seguimiento del inventario  
No venda artículos de los que ya no haya existencias: reciba una alerta por correo electrónico cuando el inventario esté bajo.

Realizar un seguimiento de ganancias y pérdidas  
Consulte un informe de ganancias y pérdidas por producto o servicio.

Por artículo: Estación de Auto-lavado	Id. de artículo	Cantidad en existencia	Cantidad para la alerta (opcional) <a href="#">¿Qué es esto?</a>	Precio
<input type="checkbox"/>	<input type="text" value="ES85"/>	<input type="text"/>	<input type="text"/>	USD

Por opción (en menú desplegable)

¿Los clientes pueden comprar un artículo cuando ya no quedan existencias?

Sí, los clientes pueden comprar el artículo como siempre.

No, no dejar que los clientes compren el artículo. [Ver más](#)

Llevar a los clientes a una página específica cuando hagan clic en Continuar compra en la página del "artículo sin existencias"

Ejemplo: <http://www.milendacomparativa.com>

[Paso 3: Personalizar funcionalidades avanzadas \(opcional\)](#)

Editar botón de pago de PayPal

paypal.com/ec/cgi-bin/webscr?cmd=\_flow&SESSION=5iqY6tz9HSqjgFwQu8y17KoJADxtLIY9eFyrP9KL...y5fh3ACN38nu73lsK&dispatch=5885d80a13c0db1f8...

**Paso 3: Personalizar funcionalidades avanzadas (opcional)**

Personalizar páginas del proceso de pago

Si es un usuario avanzado, en esta sección, puede personalizar las páginas del proceso de pago para sus clientes, dar fluidez al proceso de pago, etc.

¿Quiere permitir que el cliente cambie las cantidades en el pedido?

Sí

No

¿El cliente puede agregar instrucciones especiales en un mensaje para usted?

Sí

No

¿Necesita la dirección de envío del cliente?

Sí

No

Dirija a los clientes a este URL cuando cancelen su pago

Ejemplo: <https://www.mystore.com/cancel>

Dirija a los clientes a este URL cuando finalicen su pago

Ejemplo: <https://www.mystore.com/success>

Variables avanzadas [¿Qué es esto?](#)

Use un corte de línea entre cada variable. Las variables aparecerán en el código HTML del botón. [Más información](#)

## Anexo 5. Código web de programación login.php

```
login.php x
1 <?php
2 session_start();
3
4 require 'database.php';
5
6 if (isset($_SESSION['user_id'])) {
7     header('Location: /login');
8 }
9
10
11 if (!empty($_POST['email']) && !empty($_POST['password'])) {
12     $records = $conn->prepare('SELECT id, email, password FROM usuarios WHERE email = :email');
13     $records->bindParam(':email', $_POST['email']);
14     $records->execute();
15     $results = $records->fetch(PDO::FETCH_ASSOC);
16     $message = '';
17     if (count($results) > 0 && password_verify($_POST['password'], $results['password'])) {
18         $_SESSION['user_id'] = $results['id'];
19         header("Location: /login");
20     } else {
21         $message = 'Sorry, those credentials do not match!';
22     }
23 }
24 ?>
25
26 <!DOCTYPE html>
27 <html>
28 <head>
29 <meta charset="utf-8">
30 <title>Ingresar</title>
31
32 <link href="https://fonts.googleapis.com/css?family=Roboto&display=swap" rel="stylesheet">
33 <link rel="stylesheet" href="assets/css/styles.css">
34 </head>
35 <body>
36
37 <header>
38 <a href="/login"> Estación de AUTO-LAVADO</a>
39 </header>
40
41 <?php if(!empty($message)): ?>
42 <p <?= $message ?></p>
43 <?php endif; ?>
44
45 <h1>Login</h1>
46 <span>or <a href="signup.php">SignUp</a></span>
47 <form action="login.php" method="POST">
48
49
50
51 <input name="email" type="text" placeholder="Enter your email">
52 <input name="password" type="password" placeholder="Enter your Password">
53 <input type="submit" value="Submit">
54 </form>
55 </body>
56 </html>
57
```

## Anexo 6. Código web de programación signup.php

```
signup.php ✖
1 <?php require 'database.php'; $message = ''; if (!empty($_POST['email']) &&
2 !empty($_POST['password'])) {
3     $sql = "INSERT INTO usuarios (email, password) VALUES (:email, :password)";
4     $stmt = $conn->prepare($sql);
5     $stmt->bindParam(':email', $_POST['email']);
6     $password = password_hash($_POST['password'], PASSWORD_BCRYPT);
7     $stmt->bindParam(':password', $password);
8     if ($stmt->execute()) {
9         $message = 'Successfully created new user';
10    } else {
11        $message = 'Sorry there must have been an issue creating your account';
12    }
13 }
14 ?>
15 <!DOCTYPE html>
16 <html>
17 <head>
18 <title>Registro</title>
19 <meta charset="utf-8">
20 <link href="https://fonts.googleapis.com/css?family=Roboto&display=swap" rel="stylesheet">
21 <link rel="stylesheet" href="assets/css/styles.css">
22 </head>
23 <body>
24
25 <header>
26 <a href="/login"> Estación de AUTO-LAVADO</a>
27 </header>
28
29
30 <?php if(!empty($message)): ?>
31 <p> <?= $message ?></p>
32 <?php endif; ?>
33
34 <h1>SignUp</h1>
35 <span>or <a href="login.php">Login</a></span>
36
37 <form action="signup.php" method="post">
38 <input name="email" type="text" placeholder="Enter your email">
39 <input name="password" type="password" placeholder="Enter your Password">
40 <input name="confirm_password" type="password" placeholder="Confirm Password">
41 <input type="submit" value="Submit">
42 </form>
43
44 </body>
45 </html>
```

## Anexo 7. Código web de programación index.php

```
login.php x
1 <?php
2     session_start();
3
4     require 'database.php';
5
6     if (isset($_SESSION['user_id'])) {
7         header('Location: /login');
8     }
9
10
11     if (!empty($_POST['email']) && !empty($_POST['password'])) {
12         $records = $conn->prepare('SELECT id, email, password FROM usuarios WHERE email = :email');
13         $records->bindParam(':email', $_POST['email']);
14         $records->execute();
15         $results = $records->fetch(PDO::FETCH_ASSOC);
16         $message = '';
17         if (count($results) > 0 && password_verify($_POST['password'], $results['password'])) {
18             $_SESSION['user_id'] = $results['id'];
19             header("Location: /login");
20         } else {
21             $message = 'Sorry, those credentials do not match';
22         }
23     }
24 ?>
25
26 <!DOCTYPE html>
27 <html>
28     <head>
29         <meta charset="utf-8">
30         <title>Ingresar</title>
31
32         <link href="https://fonts.googleapis.com/css?family=Roboto&display=swap" rel="stylesheet">
33         <link rel="stylesheet" href="assets/css/styles.css">
34     </head>
35     <body>
36
37     <header>
38     <a href="/login"> Estación de AUTO-LAVADO</a>
39     </header>
40
41     <?php if(!empty($message)): ?>
42     <p <?= $message ?></p>
43     <?php endif; ?>
44
45     <h1>Login</h1>
46     <span>or <a href="signup.php">SignUp</a></span>
47     <form action="login.php" method="POST">
48
49
50
51     <input name="email" type="text" placeholder="Enter your email">
52     <input name="password" type="password" placeholder="Enter your Password">
53     <input type="submit" value="Submit">
54     </form>
55 </body>
56 </html>
57
```

## Anexo 8. Código web de programación logout.php

```
logout.php x
1 <?php
2     session_start();
3     session_unset();
4     session_destroy();
5     header('Location: /login');
6 ?>
7
8 |
```

## Anexo 9. Código web de programación styles.css

```
styles.css x
1 body {
2     margin: 0;
3     padding: 0;
4     font-family: 'Roboto', sans-serif;
5     text-align: center;
6 }
7
8 input[type="text"], input[type="password"]{
9     outline: none;
10    padding: 20px;
11    display: block;
12    width: 300px;
13    border-radius: 3px;
14    borde: 1px solid #eee;
15    margin: 20px auto;
16 }
17 input[type="submit"]{
18    padding: 10px;
19    color: #fff;
20    background: #0098cb;
21    width: 320px;
22    magin-top: 0;
23    border: 0;
24    border-radius: 3px;
25
26    cursor: pointer;
27 }
28 input[type="submit"]:hover {
29    background-color: #00b8eb;
30 }
31 header {
32    border-bottom: 2px solid #eee;
33    padding: 20px 0;
34    margin-bottom: 10px;
35    width: 100%;
36    text-align: center;
37 }
38
39
40 header a {
41    text-decoration: none;
42    color: #333;
43 }
```

## Anexo 10. Código para importa scripts de Python en PHP

```
s7_1200.php**  
1 <html>  
2 <head>  
3  
4 <meta charset="utf-8">  
5 <title>AUTOLAVADO</title>  
6 <link href="https://fonts.googleapis.com/css?family=Roboto&display=swap" rel="stylesheet">  
7 <link rel="stylesheet" href="assets/css/styles.css">  
8  
9 <!--index.php-->  
10  
11 </head>  
12 <body>  
13  
14 <br>  
15  
16 "CONTROLES DEL PROPOTIPO DE LA ESTACIÓN DE AUTO-LAVADO" <br> <br>  
17  
18 <!--GPIO17-->  
19 <form action="" method="post">  
20 Pulsadores de Emergencia <br><br>  
21 &nbsp;<input type="submit" name='Emergencia' value="Paro de Emergencia">  
22 <br><br>  
23 <input type="submit" name='Stop' value="Pulsador de Stop">  
24 <br> <br>  
25 Tiempo de encendido de la bomba de agua <br><br>  
26 &nbsp;<input type="submit" name='bomba5seg' value="Encender bomba por 5 segundos">  
27 <input type="submit" name='bomba10seg' value="Encender bomba por 10 segundos">  
28 <input type="submit" name='bomba15seg' value="Encender bomba por 15 segundos">  
29 <br> <br>  
30 Pulsadores de Mando <br><br> &nbsp;<input type="submit" name='Start' value="Pulsador de Start">  
31 <br><br>  
32 <input type="submit" name='Simple' value="Lavado Simple">  
33 <br><br>  
34 <input type="submit" name='Doble' value="Lavado Doble">  
35 <br><br>  
36 <h2>  
37 <a href="leerQplc.php">Datos del Proceso</a>  
38 </h2>  
39 <h2>  
40 <a href="logout.php">Salir de la Aplicación</a>  
41 </h2>  
42  
43  
44 </body>  
45 </html>  
46 <?php  
47 // Funciones PHP del pin GPIO 7  
48  
49 if ($_POST[Emergencia]) {  
50 $a- exec('sudo python /home/pi/Tesis/pulsador_emergencia.py');  
51 echo $a;  
52 }  
53 if ($_POST[Stop]) {  
54 $a- exec('sudo python /home/pi/Tesis/pulsador_stop.py');  
55 }  
56 if ($_POST[Start]) {  
57 $a- exec('sudo python /home/pi/Tesis/pulsador_start.py');  
58 echo $a;  
59 }  
60 if ($_POST[Simple]) {  
61 $a- exec('sudo python /home/pi/Tesis/write_lavado_simple.py');  
62 echo $a;  
63 }  
64 if ($_POST[Doble]) {  
65 $a- exec('sudo python /home/pi/Tesis/write_lavado_doble.py');  
66 echo $a;  
67 }  
68 if ($_POST[bomba5seg]) {  
69 $a- exec('sudo python /home/pi/Tesis/ONbomba5seg.py');  
70 echo $a;  
71 }  
72 if ($_POST[bomba10seg]) {  
73 $a- exec('sudo python /home/pi/Tesis/ONbomba10seg.py');  
74 echo $a;  
75 }  
76 if ($_POST[bomba15seg]) {  
77 $a- exec('sudo python /home/pi/Tesis/ONbomba15seg.py');  
78 echo $a;  
79 }  
80  
81 ?>
```

## Anexo 11. Código de actualización de datos con Jquery

```
leerQplc.php *x
1
2 <!DOCTYPE html>
3 <html>
4 <head>
5
6
7 <meta charset="utf-8">
8   <link href="https://fonts.googleapis.com/css?family=Roboto&display=swap"
9     rel="stylesheet">
10  <link rel="stylesheet" href="assets/css/styles.css">
11
12    <title>Datos</title>
13    <script
14      src="https://code.jquery.com/jquery-3.2.1.js"
15      integrity="sha256-DZAnKJ/6XZ9si04Hgrsxu/8s717jcIzLy3oi35EouyE="
16      crossorigin="anonymous"></script>
17 </head>
18 <body>
19   <h2>Datos de los actuadores del sistema</h2>
20   <div id="seccionRecargar"></div>
21
22 <h2>
23 <a href="s7_1200.php">Atras</a>
24 </h2>
25 <h2>
26 <a href="logout.php">Salir de la Aplicación</a>
27
28 </body>
29 </html>
30
31 <script type="text/javascript">
32   $(document).ready(function(){
33     setInterval(
34       function(){
35         $('#seccionRecargar').load('leerQs.php');
36       },700
37     );
38   });
39 </script>
40
```

```
leerQs.php *x
1 <?php
2
3 echo "<br>";
4 echo shell_exec('python /home/pi/Tesis/leerQ5sistema.py');
5 echo "<br>";
6 ?>
7
8
```

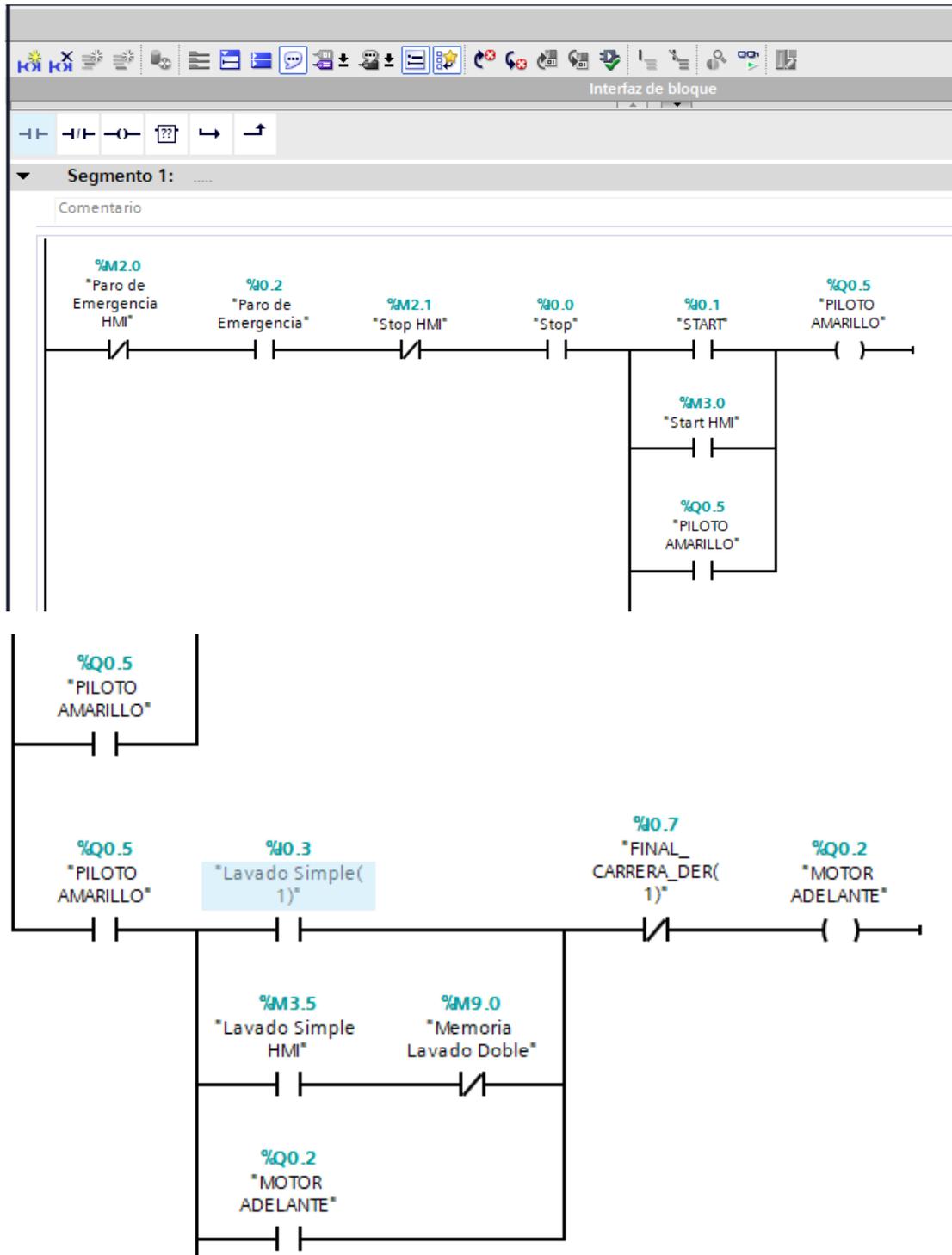
## Anexo 12. Tabla de variables programación Tia Portal V13

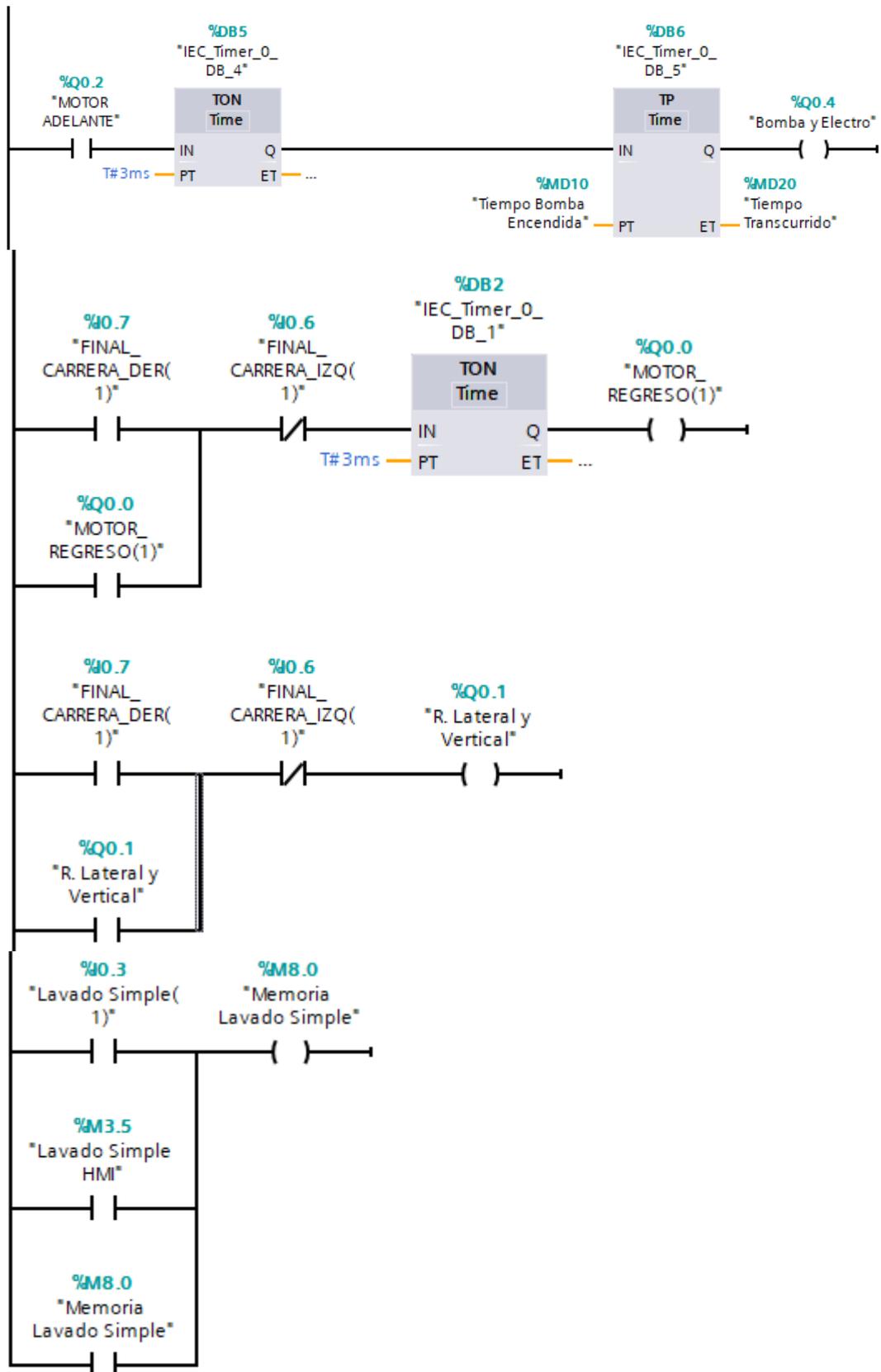
Variables								Constantes de usuario	
Tabla de variables estándar									
	Nombre	Tipo de datos	Dirección	Rema...	Visibl...	Acces...	Comentario		
1	Stop	Bool	%I0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
2	Paro de Emergencia	Bool	%I0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
3	START	Bool	%I0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
4	PILOTO AMARILLO	Bool	%Q0.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
5	Lavado Simple(1)	Bool	%I0.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
6	FINAL_CARRERA_DER(1)	Bool	%I0.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
7	MOTOR ADELANTE	Bool	%Q0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
8	FINAL_CARRERA_IJQ(1)	Bool	%I0.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
9	MOTOR_REGRESO(1)	Bool	%Q0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
10	Paro de Emergencia HMI	Bool	%M2.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
11	Stop HMI	Bool	%M2.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
12	Start HMI	Bool	%M3.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
13	Tiempo Transcurrido	DWord	%MD20	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
14	Lavado Simple HMI	Bool	%M3.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
15	Tiempo Bomba Encendida	DWord	%MD10	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
16	Bomba y Electro	Bool	%Q0.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
17	R. Lateral y Vertical	Bool	%Q0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
18	Tag_3	Bool	%M0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
19	Tag_4	Bool	%M0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
20	Tag_1	Bool	%M0.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
21	Lavado doble	Bool	%I0.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
22	Lavado doble HMI	Bool	%M4.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
23	Contador	Bool	%M0.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
24	Tag_2	Bool	%I0.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
25	Memoria Lavado Simple	Bool	%M8.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
26	Memoria Lavado Doble	Bool	%M9.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
27	Contador(1)	DWord	%MD30	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
28	N Autos Lavados	DWord	%MD40	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
29	Reset Lavado Autos	Bool	%M5.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
30	<Agregar>			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			

## Anexo 13. Tabla de variables HMI Tia Portal V13

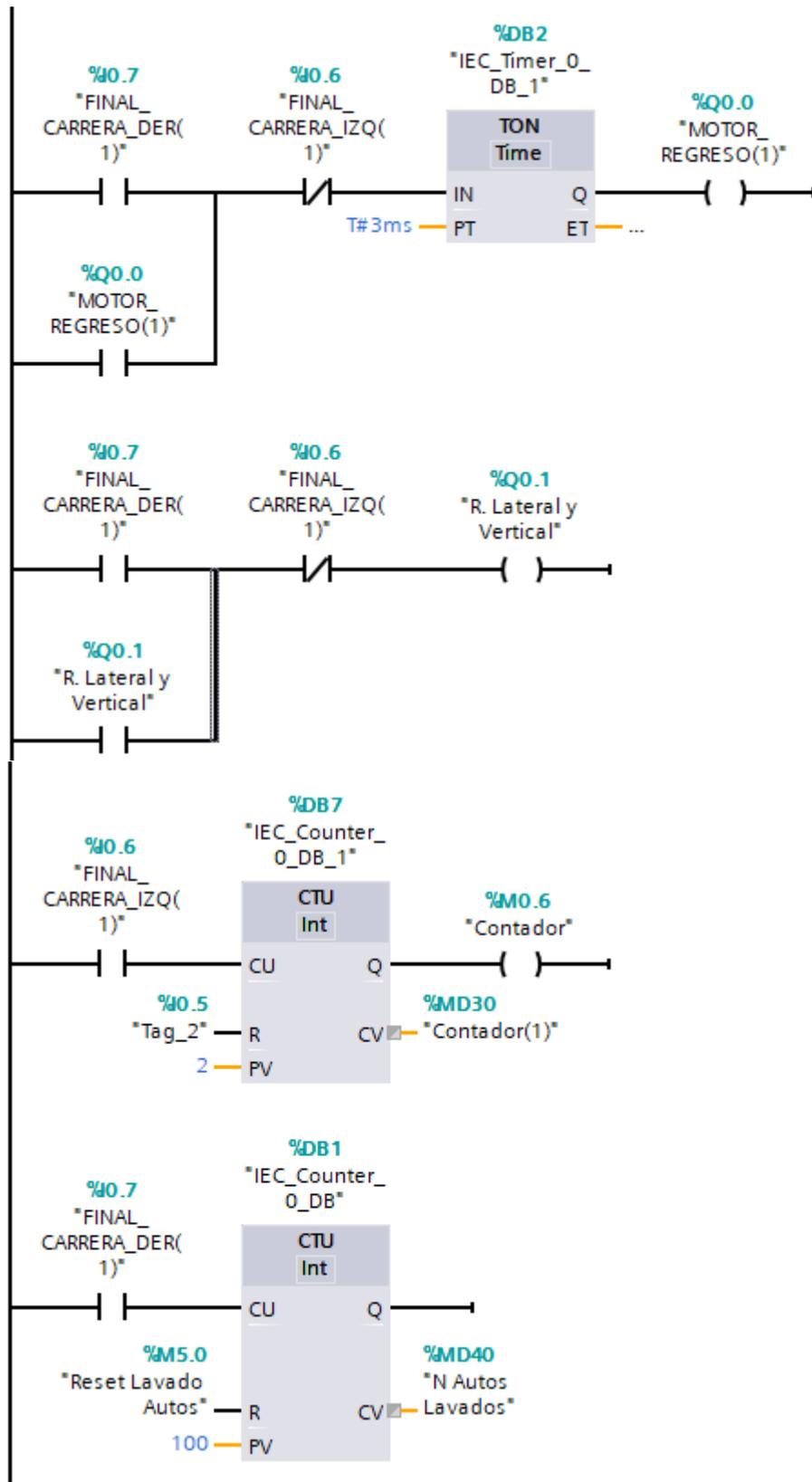
Autolavado con HMI > HMI_1 [KTP600 Basic color PN] > Variables HMI > Tabla de variables estándar [15]					
Tabla de variables estándar					
	Nombre	Tipo de datos	Conexión	Nombre PLC	Variable PLC
<input checked="" type="checkbox"/>	Bomba y Electro	Bool	HMI_Conexión_1	PLC_1	*Bomba y Electro*
<input checked="" type="checkbox"/>	Contador(1)	DWord	HMI_Conexión_1	PLC_1	Contador(1)
<input checked="" type="checkbox"/>	Lavado doble HMI	Bool	HMI_Conexión_1	PLC_1	*Lavado doble HMI*
<input checked="" type="checkbox"/>	Lavado Simple HMI	Bool	HMI_Conexión_1	PLC_1	*Lavado Simple HMI*
<input checked="" type="checkbox"/>	MOTOR ADELANTE	Bool	HMI_Conexión_1	PLC_1	*MOTOR ADELANTE*
<input checked="" type="checkbox"/>	MOTOR_REGRESO(1)	Bool	HMI_Conexión_1	PLC_1	MOTOR_REGRESO(1)
<input checked="" type="checkbox"/>	Número_imagen_variable	UInt	<Variable intern...		<No definido>
<input checked="" type="checkbox"/>	Paro de Emergencia HMI	Bool	HMI_Conexión_1	PLC_1	*Paro de Emergencia HMI*
<input checked="" type="checkbox"/>	PILOTO AMARILLO	Bool	HMI_Conexión_1	PLC_1	*PILOTO AMARILLO*
<input checked="" type="checkbox"/>	Rodillo horixontl	Bool	HMI_Conexión_1	PLC_1	*R. Lateral y Vertical*
<input checked="" type="checkbox"/>	Start HMI	Bool	HMI_Conexión_1	PLC_1	*Start HMI*
<input checked="" type="checkbox"/>	Stop HMI	Bool	HMI_Conexión_1	PLC_1	*Stop HMI*
<input checked="" type="checkbox"/>	Tag_2	Bool	HMI_Conexión_1	PLC_1	Tag_2
<input checked="" type="checkbox"/>	Tiempo Bomba Encendida	DWord	HMI_Conexión_1	PLC_1	*Tiempo Bomba Encendi...
<input checked="" type="checkbox"/>	Tiempo Transcurrido	DWord	HMI_Conexión_1	PLC_1	*Tiempo Transcurrido*

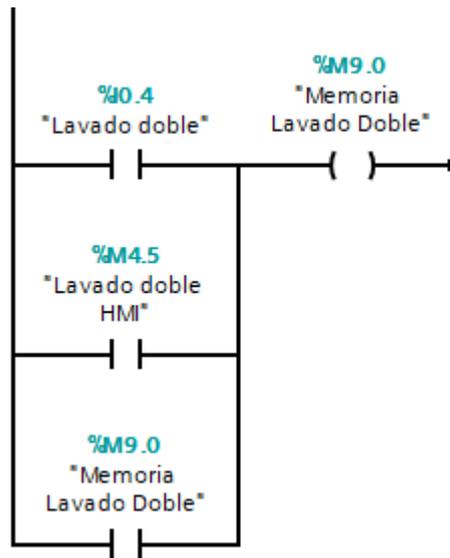
## Anexo 14. Programación lavado simple en Tia Portal V13



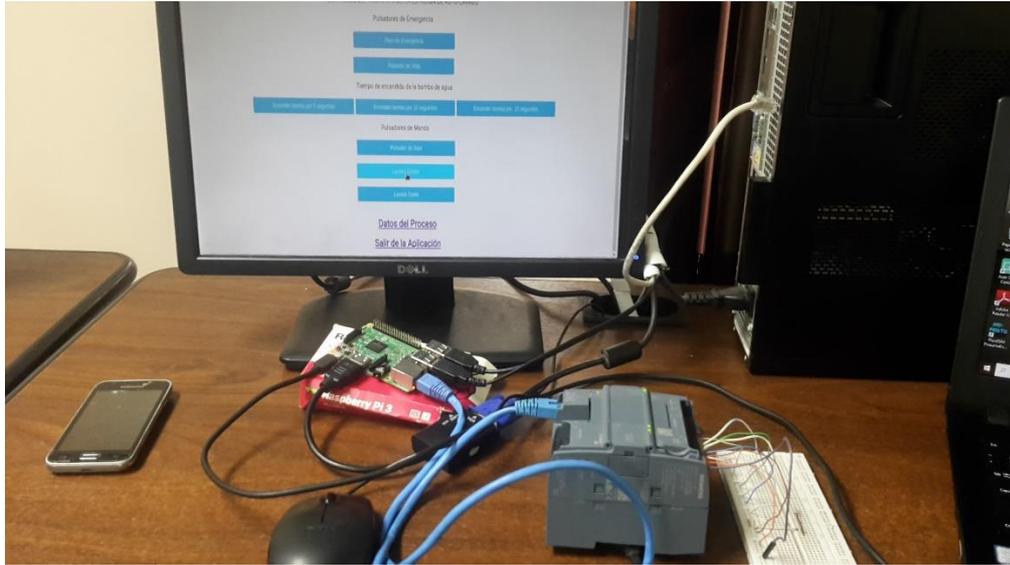






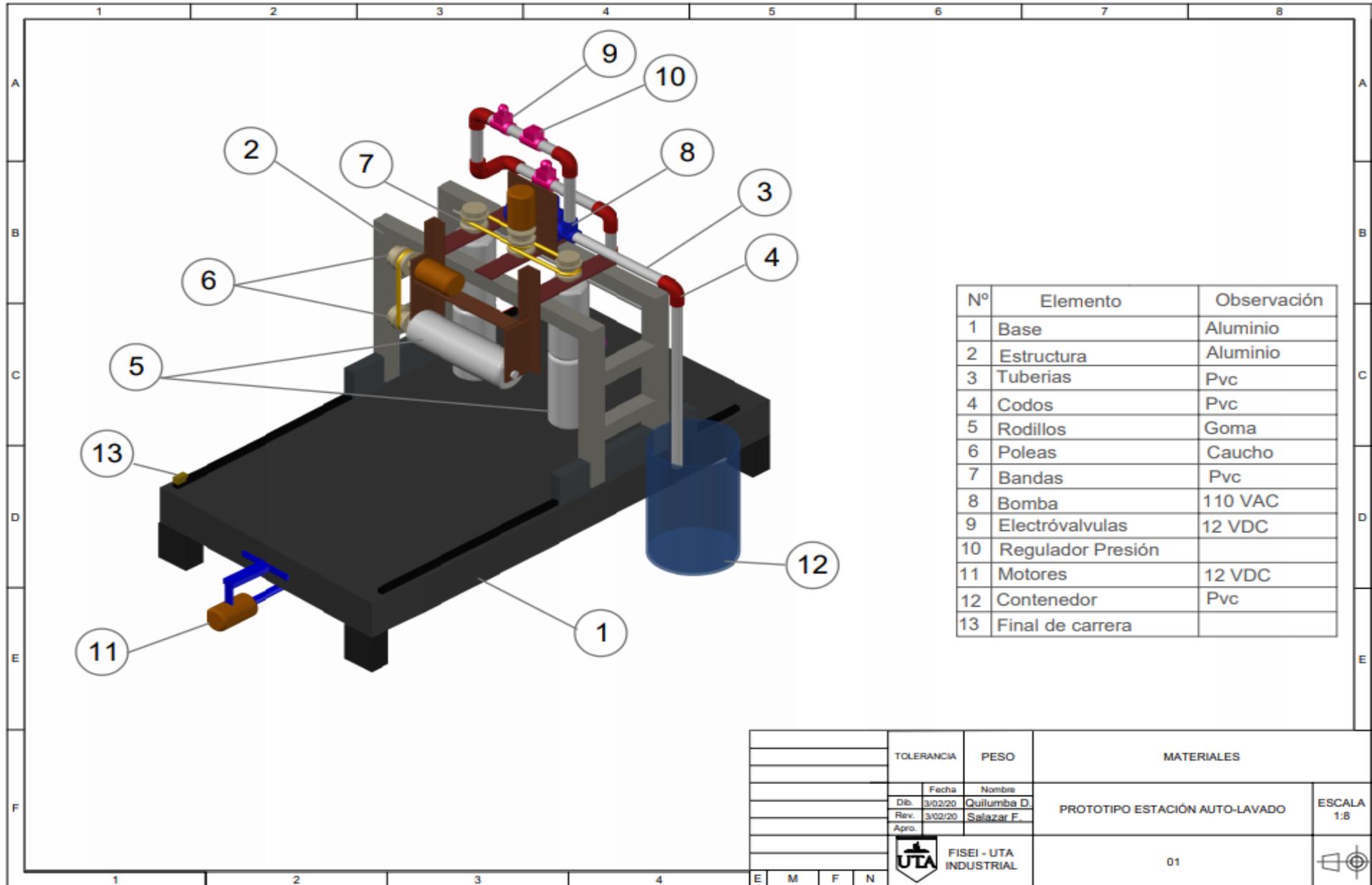


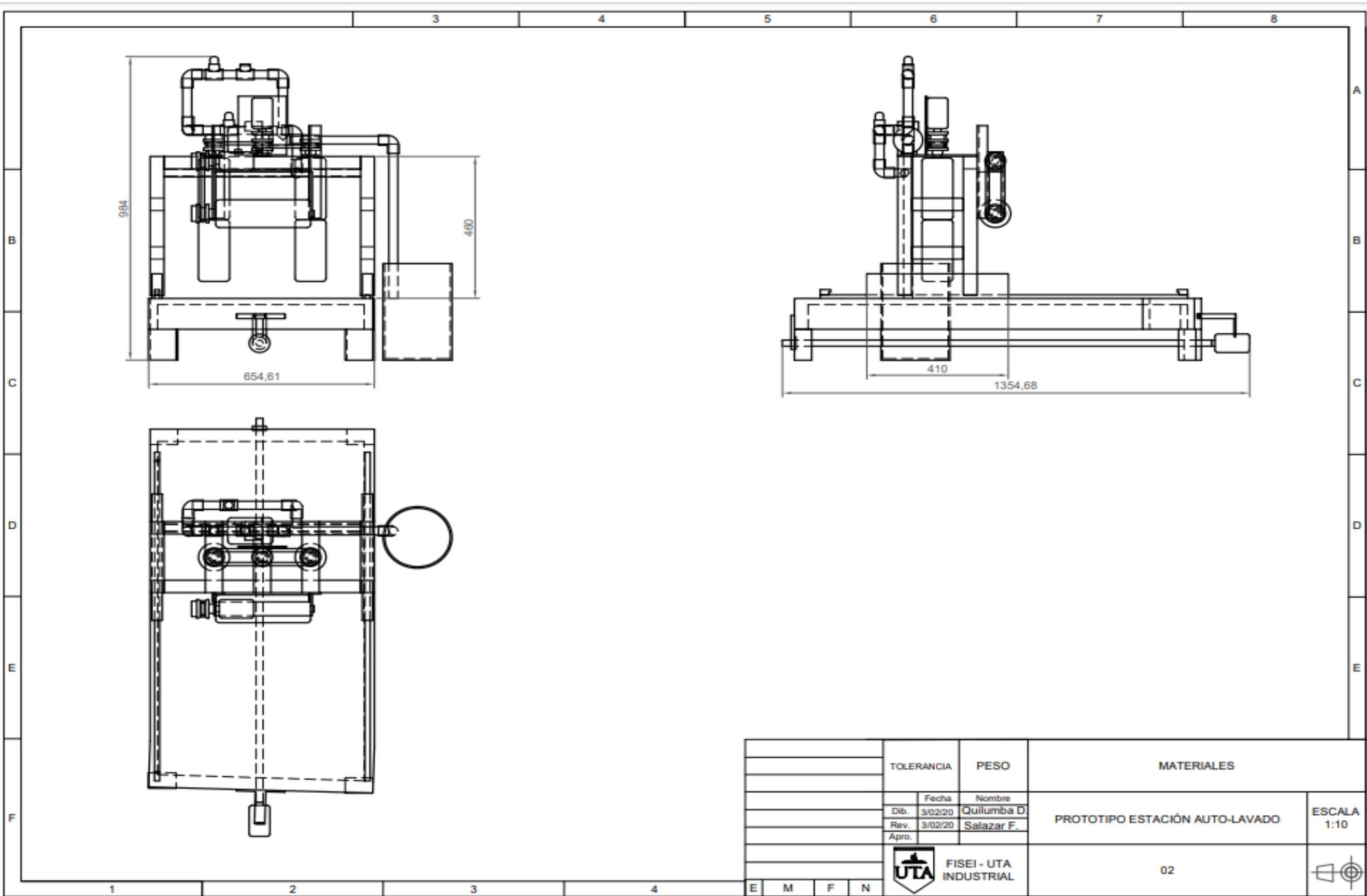
## Anexo 16. Pruebas de funcionamiento del prototipo





### Anexo 17. Planos AutoCAD





	TOLERANCIA	PESO	MATERIALES	
	Fecha	Nombre	PROTOTIPO ESTACIÓN AUTO-LAVADO	ESCALA 1:10
	Dib. 3/02/20	Quilumba D		
	Rev. 3/02/20	Salazar F.		
	Apro.			
	 FISEI - UTA INDUSTRIAL		02	
E	M	F	N	