



UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE CUENCA
CARRERA DE INGENIERÍA MECATRÓNICA

SISTEMA DE MONITOREO A TRAVÉS DE REALIDAD AUMENTADA
DEL MÓDULO SORTING AND HANDING DEL LABORATORIO DE MPS

Trabajo de titulación previo a la obtención
del título de Ingeniero Mecatrónico

AUTORES: JOAN DAVID ENCARNACIÓN DÍAZ
FÉLIX ADRIÁN TORRES TORRES

TUTOR: ING. PAÚL ANDRÉS CHASI PESANTEZ, MSc.

CO-TUTOR: ING. ANGEL EUGENIO CÁRDENAS CADME, MSc.

Cuenca – Ecuador

2022

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Nosotros, Joan David Encarnación Díaz con documento de identificación N° 1104595671 y Félix Adrián Torres Torres, con documento de identificación N° 0107529380; manifestamos que:

Somos los autores y responsables del presente trabajo; y, autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Cuenca, 28 de julio del 2022

Atentamente,



Joan David Encarnación Díaz
1104595671



Félix Adrián Torres Torres
0107529380

CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA

Nosotros, Joan David Encarnación Díaz con documento de identificación N° 1104595671 y Félix Adrián Torres Torres con documento de identificación N° 0107529380, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del Proyecto Técnico: SISTEMA DE MONITOREO A TRAVÉS DE REALIDAD AUMENTADA DEL MÓDULO SORTING AND HANDING DEL LABORATORIO DE MPS, el cual ha sido desarrollado para optar por el título de: Ingeniero Mecatrónico, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Cuenca, 28 de julio del 2022

Atentamente,



Joan David Encarnación Díaz
1104595671



Félix Adrián Torres Torres
0107529380

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Paúl Andrés Chasi Pesantez, con documento de identificación N° 0103652095, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: SISTEMA DE MONITOREO A TRAVÉS DE REALIDAD AUMENTADA DEL MÓDULO SORTING AND HANDING DEL LABORATORIO DE MPS, realizado por Joan David Encarnación Díaz con documento de identificación N° 1104595671 y Félix Adrián Torres Torres con documento de identificación N° 0107529380, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto Técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Cuenca, 28 de julio del 2022

Atentamente,



Ing. Paúl Andrés Chasi Pesantez, MSc.
0103652095

Dedicatoria

Joan David Encarnación Díaz

El presente proyecto de titulación está dedicado:

A mi madre y hermana por brindarme el apoyo incondicional en cada meta que me he propuesto. De igual manera al resto de mi familia por siempre confiar en mí y apoyarme en cada sueño que he tenido.

Finalmente, quiero agradecer a Dios por darme la oportunidad de experimentar las diferentes adversidades que presenta la vida, ayudándome a levantar en cada tropiezo, haciéndome más fuerte y sabio frente a cada problema que se presenta.

Félix Adrián Torres Torres

Este proyecto de titulación va dedicado a todas las personas que han estado en este transcurso de mi vida; el cual, han sido de vital importancia para que pueda cumplir esta meta propuesta años atrás, primeramente a mis padres, ya que ellos me han apoyado de manera moral, sentimental, emocional y sin menospreciar económicamente. Muchas gracias por esta ardua trayectoria.

De igual manera a mis hermanos, por el hecho de que gracias a sus consejos he podido salir adelante como persona.

Agradecimientos

Joan David Encarnación Díaz

Agradezco a mi madre por brindarme el apoyo necesario para culminar este proyecto de titulación, ya que con su ejemplo de superación y motivación me ayudo a alcanzar esta meta propuesta, que forma parte de un sueño cumplido. A mi familia y amigos por estar conmigo brindándome de su apoyo.

Al Ing. Paúl Chasi y Eugenio Cárdenas, ya que con su guía y consejo pudimos solventar algunas dudas de manera exitosa para la realización del presente proyecto de titulación, puesto que sin su ayuda no se hubiera terminado a tiempo. Así también al Ing. Luis López. Y a todas las personas que han aportado con su conocimiento de manera directa o indirecta.

Félix Adrián Torres Torres

Agradezco a mis padres Nube y Félix por brindarme el apoyo necesario para culminar este proyecto de titulación, ya que con su ejemplo de superación y motivación me ayudo a alcanzar esta meta propuesta, que forma parte un sueño cumplido. A mis hermanos y amigos por estar conmigo en todo momento y haberme aconsejado para bien.

Al Ing. Paúl Chasi, ya que con su guía, dedicación y conocimiento de la materia se pudo culminar de manera exitosa este proyecto de titulación, puesto que sin su ayuda no se hubiera terminado a tiempo. Así también al Ing. Eugenio Cárdenas, Ing. Luis López. Y a todas las personas que han aportado con su conocimiento de manera directa o indirecta que han sido de gran ayuda.

Este documento fue realizado enteramente en L^AT_EX

Índice

Certificado de responsabilidad y autoría del trabajo de titulación	I
Certificado de cesión de derechos de autor del trabajo de titulación a la Universidad Politécnica Salesiana	II
Certificado de dirección del trabajo de titulación	III
Dedicatoria	IV
Agradecimientos	V
Resumen	XIII
Abstract	XIV
1. Introducción	1
2. Problema	2
2.1. Antecedentes	2
2.2. Descripción del problema	2
2.3. Importancia y alcances	3
2.4. Delimitación	3
2.4.1. Espacial o geográfica	3
2.4.2. Temporal	3
2.4.3. Sectorial o institucional	4
2.4.4. Problema general	4
2.5. Problemas Específicos	4
3. Objetivos	4
3.1. Objetivo General	4
3.2. Objetivos Específicos	5
4. Hipótesis	5
4.1. Hipótesis General	5
4.2. Hipótesis Específicas	5

5. Marco Teórico	6
5.1. Sistema de Producción Modular - MPS 500	6
5.2. Módulo Handling o de Manejo	8
5.2.1. Principio de Funcionamiento	9
5.3. Módulo Sorting o de Clasificación	10
5.3.1. Principio de Funcionamiento	11
5.4. Controlador Lógico Programable (PLC)	12
5.4.1. Unidad Central de Procesos (CPU)	12
5.4.2. Interfaces de entrada y salida	12
5.5. Protocolos de Comunicación MPS	13
5.5.1. Protocolo Interfaz Multipunto (MPI)	13
5.5.2. Protocolo de comunicación Profibus	14
5.6. Funcionamiento IoT	15
5.6.1. Modelos de conectividad	16
5.6.2. Plataformas Basadas en la Nube.	18
5.7. Realidad Aumentada	19
5.7.1. Basado en características	20
5.7.2. Basado en Modelo	20
5.7.3. SDKs Para el desarrollo de realidad Aumentada	20
6. Marco metodológico	21
6.1. Diseño de un sistema de adquisición de las señales de los módulos Sorting and Handling para el monitoreo y visualización de datos en línea	22
6.1.1. Programación de PLC's mediante el Software de TIA PORTAL	22
6.1.2. Diseño de un circuito electrónico para la comunicación mediante una CPU o System on a chip	23
6.1.3. Programación para procesamiento de señales en la nube	24
6.2. Configuración de la base de datos en la nube para el acceso de las aplicaciones	27
6.2.1. Selección de la plataforma basada en cloud Computing	27
6.2.2. Configuración de firebase	28
6.2.3. Configuración de la aplicación Web	30
6.2.4. Configuración de Firebase Realtime	33
6.2.5. Incorporación del SDK de Firebase a Unity	35
6.3. Diseño de una aplicación móvil y página web para el monitoreo y visualización de datos en línea a través de Realidad Aumentada	38

6.3.1.	Diseño de etiquetas con información 3D	41
6.3.2.	Actualización de las etiquetas con la información de Firebase	42
6.3.3.	Elaboración del APK para dispositivos Android	43
6.3.4.	Diseño de una aplicación web para la visualización de los datos mediante Firebase	44
6.3.5.	Diseño de las Vistas de la aplicación Web	45
6.4.	Evaluación del desempeño de funcionamiento del sistema de monitoreo y visualización de las señales de los módulos del laboratorio de MPS	45
7.	Resultados	46
7.1.	Diseño de un sistema de adquisición de las señales de los módulos Sorting and Handling para el monitoreo y visualización de datos en línea	46
7.2.	Configuración de la base de datos en la nube para el acceso de las aplicaciones	49
7.3.	Diseño de una aplicación móvil y página web para el monitoreo y visualización de datos en línea a través de Realidad Aumentada	51
7.4.	Evaluación del desempeño de funcionamiento del sistema de monitoreo y visualización de las señales de los módulos del laboratorio de MPS	57
7.5.	Análisis de costos	61
7.5.1.	Costo presupuestado	61
7.5.2.	Costo por actividades	62
7.5.3.	Costo por paquete	63
7.5.4.	Costo total del proyecto	64
8.	Conclusiones	65
9.	Recomendaciones	66
	Referencias	68
	ANEXOS	69

Lista de Tablas

- 1. Direcciones IP del módulo Sorting 22
- 2. Direcciones IP del módulo Handling 23
- 3. Señales adquiridas del módulo Sorting and Handling a monitorizar 47
- 4. Análisis de costo presupuestado de los materiales y equipos 62

Lista de Figuras

1.	Módulos MPS	7
2.	Módulo Handling	9
3.	Módulo Sorting	11
4.	Topología de un protocolo MPI	14
5.	Internet de las Cosas (IoT)	15
6.	Protocolo UDP vs TCP	16
7.	Esquema Electrónico para señales del PLC al CPU	24
8.	Código de programación para comunicación con Firebase	25
9.	Código de programación para definir como entradas a los pines del CPU	25
10.	Código de programación con funciones para clasificar información	26
11.	Código de programación para condicionar valores de input lógicos	27
12.	Plataformas de nube	28
13.	Creación de proyecto Firebase	29
14.	Servicios de Firebase	30
15.	Vinculación de aplicaciones a firebase	31
16.	Credenciales de conexión a Firebase.	32
17.	Registro de la aplicación web.	33
18.	Estructura de la base de datos.	33
19.	Variables de los módulos en Realtime Firebase.	34
20.	Variables de información de los modulos.	35
21.	Reglas de seguridad en Firebase Realtime.	35
22.	Incorporación del paquete de firebase a unity	36
23.	Selección de Dotnet al proyecto.	36
24.	Incorporación de la librería de Firebase realtime	37
25.	Función encargada de tomar datos de Firebase	38
26.	Importacion del SDK de vuforia.	39
27.	Vinculación AR camera	39
28.	Configuración Image Target	41
29.	Etiquetas de Información	42
30.	Objeto de los Image Target	43
31.	Ajustes de la configuración en Android	44
32.	Esquema Electrónico para conexión del PLC a Raspberry pi 3 para el módulo Sorting	48

33.	Esquema Electrónico para conexión del PLC a Raspberry pi 3 para el módulo Handling	49
34.	Variación de estado de señales del módulo Handling	50
35.	Variación de estado de señales del módulo Sorting	50
36.	Información del proyecto visualizado en la página web	51
37.	Información del Dashboard visualizado en la página web	52
38.	Visualización del Home de la página web	53
39.	Variación de estado de sensores y actuadores del módulo Sorting con realidad aumentada una tapa de color negro	54
40.	Variación de estado de sensores y actuadores del módulo Sorting con realidad aumentada una tapa de color metálico	54
41.	Variación de estado de sensores y actuadores del módulo Sorting con realidad aumentada una tapa de color rojo	55
42.	Variación de estado del sensor 1B1 con realidad aumentada	56
43.	Variación de estado del sensor 1B3 con realidad aumentada	56
44.	Resultado de evaluación de pregunta 1	57
45.	Resultado de evaluación de pregunta 2	58
46.	Resultado de evaluación de pregunta 3	59
47.	Resultado de evaluación de pregunta 4	60
48.	Resultado de evaluación de pregunta 5	61
49.	Programación de secuencia lógica del módulo Handling Network 1	70
50.	Programación de secuencia lógica del módulo Handling Network 2	70
51.	Programación de secuencia lógica del módulo Handling Network 3 y 4	71
52.	Programación de secuencia lógica del módulo Handling Network 5 y 6	71
53.	Programación de secuencia lógica del módulo Sorting Network 1	72
54.	Programación de secuencia lógica del módulo Sorting Network 2 y 3	72
55.	Programación de secuencia lógica del módulo Sorting Network 4 y 5	73
56.	Programación de secuencia lógica del módulo Sorting Network 6 y 7	73
57.	Programación de secuencia lógica del módulo Sorting Network 8 y 9	74

Resumen

En el presente trabajo de titulación se propone la implementación de un sistema de monitoreo y visualización de datos de los módulos Sorting and Handling del laboratorio de Sistemas de Producción Modular (MPS) de la Universidad Politécnica Salesiana a través de realidad aumentada, que sirve como soporte para la implementación y aparición de nuevas tecnologías dentro de la industria 4.0; permitiendo así, la visualización del funcionamiento de sensores y actuadores presentes dentro de los módulos establecidos a monitorear a través de una aplicación web y mediante la visualización a través de realidad aumentada el cual se puede supervisar líneas de producción desde cualquier lugar del mundo con acceso a internet y un dispositivo inteligente.

En primera instancia, el proyecto se centra en la emisión de las señales proporcionadas por el Controlador Lógico Programable (PLC) hacia una base de datos en internet mediante la plataforma de Firebase, lo que permite crear un sistema IoT donde se visualiza la información en línea para supervisar el funcionamiento de la simulación de una línea de producción a través de la flexibilidad incorporada del MPS 500. Es decir, esto genera una red de comunicación a través de una tarjeta electrónica con acceso a internet para el envío de señales de sensores y actuadores presentes en los módulos respectivamente.

Seguidamente, se enlaza los datos obtenidos a través de la tarjeta electrónica a una aplicación web denominado Dashboard, el cual permite visualizar la información proporcionada de una forma más ordenada y explicativa. Luego, se procedió a conectar a la nube de Firebase mediante un kit de desarrollo de software (SDK) implementando co-rutinas y librerías que permitió la comunicación con el motor de videojuegos Unity, donde se procedió a desarrollar elementos en 2D y 3D para aplicaciones móviles implicando la utilización de cámaras en el reconocimiento de imágenes. El trabajo se concluyó con una evaluación de la funcionalidad de la implementación del sistema de monitoreo y visualización de datos en línea de un proceso de producción en la industria, lo cual abarca un enfoque mecatrónico.

Palabras clave: Línea de producción, Realidad Aumentada, Monitoreo y Visualización, Base de datos, Controladores lógicos, Software de ingeniería, Tarjeta electrónica.

Abstract

In the present degree project, it is proposed the implementation of a system of monitoring and visualization of data from the Sorting and Handling modules of the laboratory of Modular Production System (MPS) of the Universidad Politecnica Salesiana through augmented reality, giving to serve as support for the implementation and appearance of new technologies within industry 4.0; thus allowing the visualization of the operation of sensors and actuators present within the modules established to be monitored through a web application and visualization with augmented reality providing the benefit of supervising production lines from anywhere in the world with internet access and a smart device.

In the first instance, the project focuses on the emission of the signals provided by the Programmable Logic Controller (PLC) to a database on the internet through the Firebase platform, which allows you to create a "IoT" system where you can visualize this information online to monitor the operation of the simulation of a line of products through the built-in flexibility of the MPS 500. In other words, this creates a network of communication through an electronic card with internet access for sending signals from sensors and actuators present in the modules respectively.

Next, the data obtained through the electronic card is linked to a web application called Dashboard, which allows viewing the information provided in a orderly and explanatory way. Then, we proceeded to connect to the Firebase cloud through a Software Development Kit (SDK) implementing coroutines and libraries that allowed communication with the developer of Unity, where they proceeded to develop 2D and 3D elements in mobile applications involving the use of cameras. The project was concluded with an evaluation of the functionality of the implementation of the online data monitoring and visualization of a production process in the industry, which encompasses a mechatronic approach.

Keywords: Production line, Augmented Reality, Monitoring and Visualization, Base of data, Logic controllers, Engineering software, and Electronic card.

1. Introducción

En la actualidad, el desarrollo de la tecnología, la aparición de las recientes innovaciones de la electrónica, la informática y las tecnologías de la información y la comunicación (TIC) han proporcionado un crecimiento vertiginoso en cuanto al procesamiento de los sistemas de tratamiento de información, resaltando el surgimiento del Internet de las Cosas (IoT), donde el principal objetivo es establecer una conexión de los objetos físicos con el Internet para interactuar entre sí. (Andrés, 2020)

El uso y la aplicación del IoT puede depender y afectar del modo en el que se utiliza. Dentro del ámbito laboral existe la posibilidad de implementar la monitorización y visualización de señales eléctricas de equipos industriales de manera remota, permitiendo realizar que el sistema sea muy accesible ante la manejabilidad del operador o de la persona involucrada, sin la necesidad que se encuentre presente en el entorno, permitiendo así visualizar esta información desde cualquier otro lugar del planeta. Así también, implementando una nueva tecnología innovadora al momento de relacionar el IoT con la Realidad Aumentada para desarrollar y mejorar el modo de percepción del ser humano.

En el presente proyecto de titulación se realizó la implementación de un sistema de monitorización en dos módulos del laboratorio de Sistema Modular de Producción (MPS) de la Universidad Politécnica Salesiana mediante la determinación de: los procesos de manejo y clasificación presentes en las líneas de producción industriales, el software de ingeniería, redes IoT, Realidad Aumentada (RA), plataformas en la nube como Firebase y la presencia de Controladores Lógicos Programables (PLC's), que se utilizó para el desarrollo del proyecto, Para después continuar con la comunicación y conexión de las señales eléctricas de los sensores y actuadores presentes en cada módulo con una tarjeta electrónica enviando estos datos a la plataforma de la nube y crear una base de datos en Internet, además de visualizar esta información a través de Realidad Aumentada desde un dispositivo móvil, dando así un enfoque mecatrónico determinando la relación con automatización industrial reforzando conceptos para la manipulación y visualización de equipos industriales que se encuentran presentes en campo de la ingeniería, enfocando directamente el área de un ingeniero mecatrónico.

2. Problema

2.1. Antecedentes

El mundo se encuentra en un constante cambio tecnológico, lo cual incide en el desarrollo industrial debido a la implementación de nuevas tecnologías. Estas nuevas tecnologías vienen fuertemente vinculadas a las necesidades que surgen en el ámbito laboral, donde se nota un cambio de paradigmas al momento de aplicar nuevas técnicas, permitiendo categorizar cada revolución como una mejora de la anterior. Es aquí donde la aplicación de nuevas metodologías para la comprensión de los datos surgen como es el caso de la realidad aumentada. El término de realidad aumentada es una tecnología que existe desde 1990 por el investigador Boeing Tom Caudal, donde la primera aplicación que se realizó consiste en superponer los planos de cableados sobre las piezas producidas. Se considera que esta fue una de las primeras aplicaciones dentro del campo laboral para mejorar la interpretación de un sistema y gracias a esto viene se ha visto impulsado en la industria en estas últimas décadas, especialmente a partir de la llegada de la industria 4.0 y el internet de las cosas. (Bejerano, 2014)

Dentro del campo laboral y educativo existen ejemplos donde la integración de realidad aumentada ha sido de gran ayuda al momento de querer explicar algún producto o tema en general. Velux es una compañía que se especializa en ventanas de techo y tragaluz. Esta compañía a partir del 22 de septiembre del 2020 se implementó una aplicación para visualizar a través de realidad aumentada las decoraciones que ellos realizaban con el objetivo de permitir al consumidor una mejor perspectiva del servicio. (Velux, 2020)

Volvo es una marca de automóviles que surgió en Suecia, actualmente está incursionando en el mundo de la realidad aumentada con dos diferentes prototipos de sistemas de realidad aumentada donde el principal objetivo es facilitar el ensamblaje de bloques de motor en sus talleres, para ello uno de los sistemas que plantea es una tableta óptica que utiliza una cámara para mapear y generar un gráfico en 3D superpuesto al plano de enfoque de la cámara.

2.2. Descripción del problema

La utilización de los módulos del laboratorio MPS permiten el monitoreo de las variables durante el funcionamiento del mismo mediante la implementación de un PLC (Controlador Lógico Programable) que se encargará del control de los procesos establecidos en el laboratorio MPS; sin embargo, el presente sistema no permite obtener una experiencia inmersiva al

momento del monitoreo de datos. Por lo cual, es factible poner en práctica y en desarrollo un prototipo de realidad aumentada para visualizar la transferencia de datos mediante una aplicación móvil y web, que permitirá una mejor perspectiva del módulo, ayudando a los educadores a explicar mejor el proceso de funcionamiento.

2.3. Importancia y alcances

- Reducción de costos: por medio de las plataformas de internet, el operario o estudiante puede visualizar la información funcional del laboratorio y esto hace que el costo relacionado con el transporte y el traslado de la persona desde su domicilio hasta la institución reduzca de manera considerable.
- Relación con plataformas de Internet: Debido a que las señales de los sensores y actuadores presentes en los módulos del laboratorio MPS están conectados a una tarjeta electrónica que envía esta información a la nube, el usuario puede tener acceso directo de esta información desde cualquier comodidad de su hogar que tenga internet.
- Innovación tecnológica: el manejo de laboratorios de forma remota mejora la percepción de los estudiantes involucrados debido a que manejan un sistema tecnológico, puesto que existe un gran número de personas que tienen acceso a estos laboratorios.

2.4. Delimitación

El problema de estudio se delimitará en las siguientes dimensiones:

2.4.1. Espacial o geográfica

El trabajo de titulación implementación de un sistema de monitoreo a través de realidad aumentada del módulo “Sorting and Handing” se realizó en el laboratorio de MPS en la Universidad Politécnica Salesiana sede Cuenca, ubicada en la provincia del Azuay con dirección: Calle vieja y Elia Liut.

2.4.2. Temporal

El presente proyecto de titulación se realizó en un tiempo establecido de 6 meses, dividido de manera proporcional con base a los objetivos planteados.

2.4.3. Sectorial o institucional

Partiendo del aval de la Universidad Politécnica Salesiana, el proyecto de titulación se realizó en el laboratorio de Sistema Modular de Producción (MPS) del área de mecánica.

2.4.4. Problema general

¿Es posible monitorear y visualizar las variables del módulo “Sorting and Handling” del laboratorio MPS de la Universidad Politécnica Salesiana a través de realidad aumentada para desarrollar una aplicación móvil mejorando la interpretación de datos en línea?

2.5. Problemas Específicos

- 1.- ¿Se podrá adquirir las variables del módulo “Sorting and Handling” del laboratorio MPS de la Universidad Politécnica Salesiana en el monitoreo y visualización de las variables en el proceso para desarrollar una aplicación móvil mejorando la interpretación de datos en línea?
- 2.- ¿Es factible proponer el diseño de monitoreo y visualización de las variables del módulo “Sorting and Handling” del laboratorio MPS de la Universidad Politécnica Salesiana para desarrollar una aplicación móvil mejorando la interpretación de datos en línea?
- 3.- ¿Es posible implementar una interfaz gráfica y montar protocolos de comunicación para el monitoreo y visualización de las variables del módulo “Sorting and Handling” del laboratorio MPS de la Universidad Politécnica Salesiana, adquiriendo datos en línea?
- 4.- ¿Existe la posibilidad de evaluar el desempeño de la interfaz gráfica y los protocolos de comunicación para el monitoreo y visualización de las variables del módulo “Sorting and Handling” del laboratorio MPS de la Universidad Politécnica Salesiana adquiriendo datos en línea?

3. Objetivos

3.1. Objetivo General

- Monitorear y visualizar las variables del módulo “Sorting and Handling” del laboratorio MPS de la Universidad Politécnica Salesiana a través de realidad aumentada para desarrollar una aplicación móvil mejorando la interpretación de datos en línea.

3.2. Objetivos Específicos

- 1.- Adquirir las variables del módulo “Sorting and Handing” del laboratorio MPS de la Universidad Politécnica Salesiana en el monitoreo y visualización de las variables en el proceso para desarrollar una aplicación móvil mejorando la interpretación de datos en línea.
- 2.- Proponer el diseño de monitoreo y visualización de las variables del módulo “Sorting and Handing” del laboratorio MPS de la Universidad Politécnica Salesiana para desarrollar una aplicación móvil mejorando la interpretación de datos en línea.
- 3.- Implementar una interfaz gráfica y montar protocolos de comunicación para el monitoreo y visualización de las variables del módulo “Sorting and Handing” del laboratorio MPS de la Universidad Politécnica Salesiana adquiriendo datos en línea.
- 4.- Evaluar el desempeño de la interfaz gráfica y los protocolos de comunicación para el monitoreo y visualización de las variables del módulo “Sorting and Handing” del laboratorio MPS de la Universidad Politécnica Salesiana adquiriendo datos en línea.

4. Hipótesis

4.1. Hipótesis General

- Se monitoreará y visualizará las variables del módulo “Sorting and Handing” del laboratorio MPS de la Universidad Politécnica Salesiana a través de realidad aumentada para desarrollar una aplicación móvil mejorando la interpretación de datos en línea.

4.2. Hipótesis Específicas

- 1.- Se adquirirá las variables del módulo “Sorting and Handing” del laboratorio MPS de la Universidad Politécnica Salesiana en el monitoreo y visualización de las variables en el proceso para desarrollar una aplicación móvil mejorando la interpretación de datos en línea.
- 2.- Se propondrá el diseño de monitoreo y visualización de las variables del módulo “Sorting and Handing” del laboratorio MPS de la Universidad Politécnica Salesiana para desarrollar una aplicación móvil mejorando la interpretación de datos en línea.

- 3.- Se implementará una interfaz gráfica y montar protocolos de comunicación para el monitoreo y visualización de las variables del módulo “Sorting and Handing” del laboratorio MPS de la Universidad Politécnica Salesiana adquiriendo datos en línea.
- 4.- Se evaluará el desempeño de la interfaz gráfica y los protocolos de comunicación para el monitoreo y visualización de las variables del módulo “Sorting and Handing” del laboratorio MPS de la Universidad Politécnica Salesiana adquiriendo datos en línea.

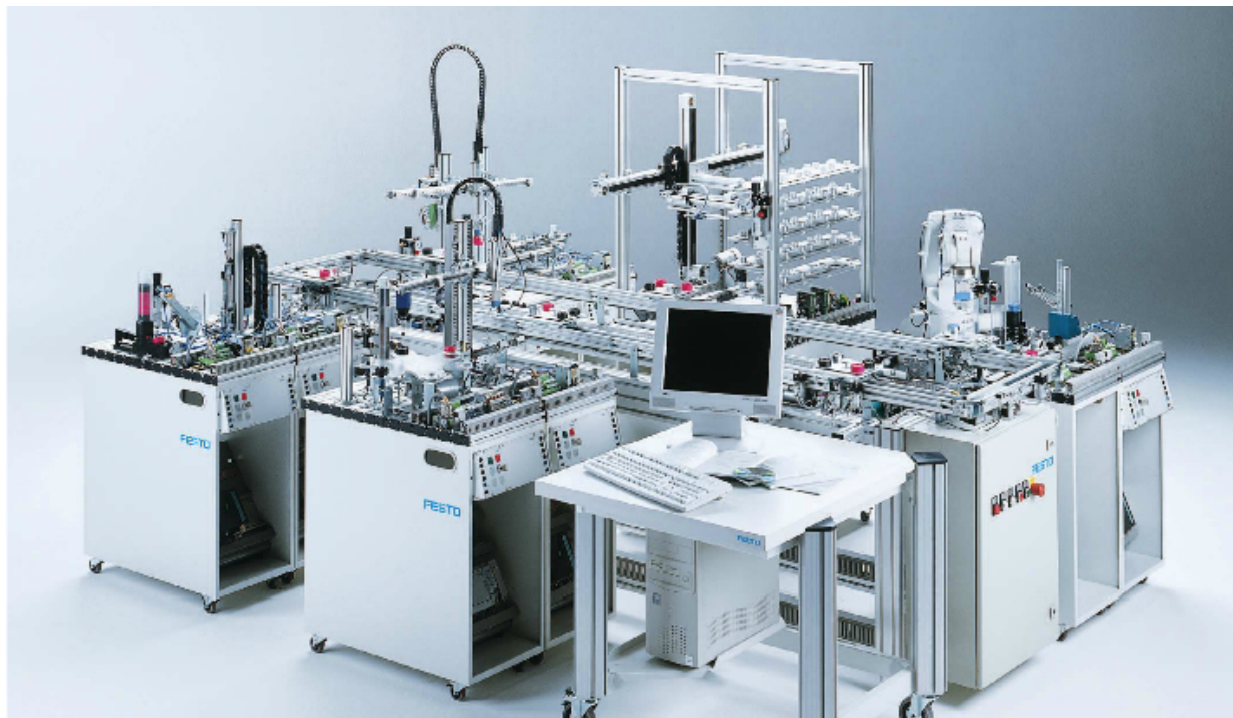
5. Marco Teórico

5.1. Sistema de Producción Modular - MPS 500

Las estaciones del MPS 500 conforman la similitud de una cadena de producción de una fábrica que están formados por módulos individuales. Cada uno de estos módulos cuenta con una función determinada dentro de un proceso, ya sea de: distribución, procesamiento, montaje, verificación o almacenamiento. Donde, a partir de un gran número de estaciones o módulos individuales, se le permite realizar una aplicación o proceso al combinarlos dependiendo a la necesidad requerida. A través de una combinación estructurada de estaciones individuales, se observa en la figura 1 que es posible realizar una propia instalación de producción deseada por el usuario u operario. (FestoDidactic, 2021)

Figura 1

Módulos MPS



Nota: Se observa los diferentes módulos existentes para la implementación del MPS, según FestoDidactic (2021).

Existen los siguientes módulos que son posibles adquirir para realizar combinaciones deseadas:

- Estación de distribución/cinta
- Estación Pick Place
- Estación de Robot
- Estación de Robot con módulos MPS
- Estación de Distribución
- Estación de Medición
- Estación de Embalaje
- Estación de mecanizado

- Estación de Manipulación, neumática
- Estación de Manipulación, eléctrica
- Estación de Prensa con Músculo Neumático
- Estación de Punzonado
- Estación de Separación
- Estación de almacenamiento
- Estación de Clasificación
- Estación de Programación

5.2. Módulo Handling o de Manejo

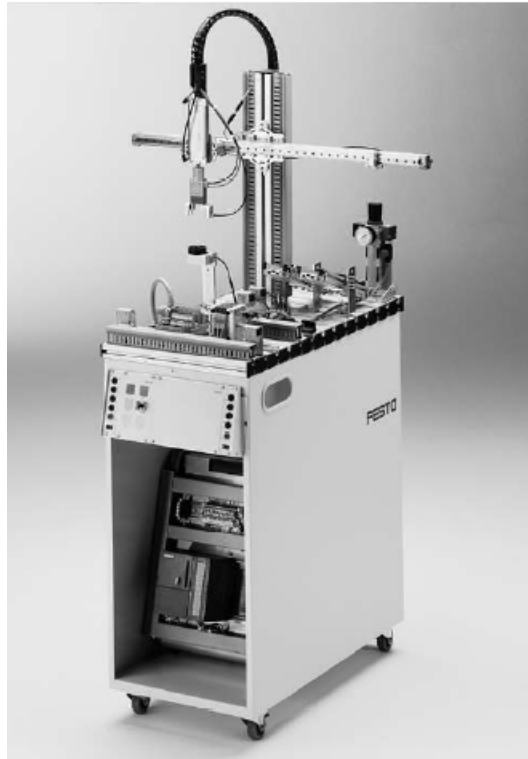
El módulo Handling es una subfunción de flujo de materiales, donde las subfunciones adicionales a este son los módulos de transporte, almacenamiento y de clasificación. Este módulo puede crear, definir o determinar de una disposición espacial, específica o temporal de cuerpos geoméricamente determinados, como se aprecia en la figura 2.

Las funciones principales características de este módulo son:

- Determinar las características materiales de un espacio de trabajo.
- Sacar piezas de trabajo de un receptáculo.
- Depositar piezas de trabajos sobre otros módulos, ya sea en una estación posterior.
- Clasificar por colores metálico, rojo o negro.

Figura 2

Módulo Handling



Nota: Se identifica la estructura y posición del modulo Handling para la implementación del MPS, según SIEMENS (2006b).

5.2.1. Principio de Funcionamiento

El módulo Handling está equipado con un dispositivo de manipulación de dos ejes donde las piezas de trabajo insertadas son detectadas en el dispositivo de retención por un sensor de reflexión de luz óptica. También, recoge las piezas de trabajo del dispositivo de retención con la ayuda de una pinza neumática que cuenta con un sensor óptico. El sensor diferencia entre piezas de trabajo negro o diferente color, donde las piezas son depositadas en diferentes puntos requeridos con base a los criterios establecidos.

Se definen otros criterios de manejo y clasificación si la estación se combina con otras estaciones al cambiar la configuración de los topes mecánicos con base en la necesidad.

5.3. Módulo Sorting o de Clasificación

El módulo Sorting es una función de clasificación que es una subcategoría de la función de manipulación. Este módulo cuenta con una cinta transportadora que tiene bifurcaciones para clasificar las piezas a manipular en función a las características de las piezas que incluyen diferentes derivadores. Las piezas de trabajo a manipular deben transportarse individualmente para que no interfieran en las operaciones de conmutación de los diferentes separadores. Esta estación separa las piezas de trabajo, mediante accionamientos neumáticos, entregando a tres planos inclinados, como se aprecia en la figura 3 más adelante. (SIEMENS, 2006c)

La estación de clasificación únicamente consiste en clasificar piezas a manipular, diferenciándolas por sus características físicas.

El módulo Sorting consta de lo siguiente:

- Módulo transportador de clasificación
- Módulo deslizante
- Placa de perfil
- Control de consola
- Placa PLC

Figura 3

Módulo Sorting



Nota: Se observa la estructura y posición del módulo Sorting de MPS según SIEMENS (2006c).

5.3.1. Principio de Funcionamiento

El módulo Sorting consta con sensores ópticos que se encuentran al inicio de la cinta transportadora que al topar con una unidad neumática en forma de horquilla detecte piezas de color rojo, negro o metálico. Mediante la implementación de 3 sensores de proximidad, el módulo detecta el material o el color de las piezas. Con el fin de detectar el color y material de la pieza se utilizan un sensor de proximidad inductivo y dos sensores ópticos de la siguiente manera:

- El sensor de proximidad inductivo detecta piezas de trabajo metálicas.
- El sensor de reflexión directa detecta piezas a manipular rojas y metálicas.
- La barrera óptica en horquilla detecta todas las piezas.

5.4. Controlador Lógico Programable (PLC)

El PLC es un dispositivo electrónico de operación digital que usa una memoria para el almacenamiento interno de instrucciones que realizan funciones de software como: operaciones aritméticas, temporización, conteo, secuencias lógicas; y ejecuta el proceso mediante módulos de entrada y salida. Por esta razón, el uso de PLC's dentro de la industria en tareas de automatización son bastante utilizadas por su gran versatilidad en usos a nivel industrial. (Borreiro y Paguay, 2010)

Según Daneri (2009) un PLC permite controlar un proceso o realizar una secuencia de manera automática a partir de un programa definido por el usuario. Esto lo realiza a partir de una ejecución cíclica de una secuencia de instrucciones que llegan de la información proporcionada de sensores o actuadores.

El PLC tiene incorporado generalmente las siguientes partes fundamentales:

5.4.1. Unidad Central de Procesos (CPU)

El CPU es el cerebro del PLC y está formado por el procesador y la memoria, donde el procesador se encarga de ejecutar el programa definido por el usuario que se encuentra almacenado en la memoria. Además, el procesador mediante sus puertos de comunicación se comunica con el exterior, adicionalmente tiene la facilidad de realizar funciones de auto-diagnóstico. Para controlar todo el sistema, el procesador necesita de un programa escrito y determinado por el fabricante denominado Firmware, este programa no es accesible para el usuario y se encuentra almacenado en una memoria no volátil que forma parte del bloque de memoria del CPU. (Daneri, 2009)

5.4.2. Interfaces de entrada y salida

Las entradas y salidas (input/output) son las partes del controlador programable que lo vinculan con el campo; como se mencionó, su función es de adaptar las señales de los captadores para que sean reconocidas por la CPU. El PLC realiza las acciones de control mediante sus entradas y salidas. Las entradas son aquellas señales de los dispositivos de campo, como: sensores e interruptores; y las salidas comandan los actuadores del proceso, como: bombas, motores u otros. (Daneri, 2009)

5.5. Protocolos de Comunicación MPS

El laboratorio MPS cuenta con el protocolo de comunicación Profibus para interactuar con los diferentes dispositivos que se encuentran conectados; sin embargo, no es el único protocolo presente, también existen protocolos como el MPI para la lectura y conexión con los diferentes periféricos del laboratorio.

El funcionamiento de los protocolos Profibus y MPI se basan en la arquitectura de red de comunicaciones y transporte de datos, comúnmente conocida como ‘modelo OSI’. Sin embargo, hay que tener en cuenta que no todos los protocolos de comunicación industrial ocupan todas las capas de este modelo, sino que únicamente se centran en determinadas capas.

5.5.1. Protocolo Interfaz Multipunto (MPI)

MPI es un protocolo de comunicación que está disponible en los equipos Siemens S7, se basa en el estándar profibus y soporta las topologías estrella, árbol y bus, las cuales permiten conectar hasta un máximo de 32 participantes, para una mejor comprensión de las topologías observe la figura 4. Este protocolo es principalmente empleado en sistemas SCADA donde se necesita recopilar información de los bancos de memoria que disponen los autómatas o PLC’s. (Torres, 2009)

El principio de funcionamiento de este protocolo se basa en la norma física RS485. Según Weis (s.f.) es un estándar de comunicaciones muy utilizado en aplicaciones de adquisición y control de datos, donde una de sus principales ventajas es que permite incluir varios dispositivos RS485 en el mismo bus, lo que hace posible que varios nodos se conecten entre sí.

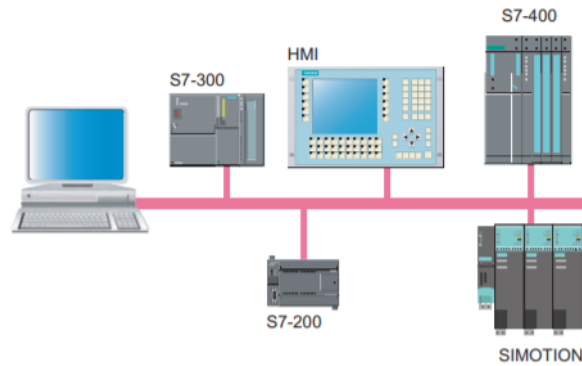
Al ser un protocolo de Siemens, permite una serie de características las cuales destacan las siguientes:

- Tiene un Número máximo de 32 de participantes.
- Cada CPU tiene la posibilidad de un máximo de 8 conexiones de comunicación dinámica para la comunicación básica con un SIMATIC S7/M7-300/-400.
- Cada CPU puede operar un máximo de 4 conexiones de comunicación estática con PG/PC, Sistemas SIMATIC HMI y SIMATIC S7/M7-300/400.
- Velocidad de transmisión de datos desde 187,5 kbit/s hasta 12Mbit/s.

- Posibilidades flexibles de configuración en el bus o estructura en árbol.
- la longitud máxima del cable es de 10 Km.
- tiene una Interfase RS485.

Figura 4

Topología de un protocolo MPI



Nota: Se aprecia la topología de un protocolo MPI según SIEMENS (2006a).

5.5.2. Protocolo de comunicación Profibus

El protocolo Profibus es un estándar de red digital de campo abierto (bus de campo) que se encarga de la comunicación entre sensores y el sistema de control o los controladores. También, es un protocolo mundial compatible con las normas internacionales IEC 61158 y es ampliamente usado en diferentes ambientes industriales. Es muy complejo debido a que está creado a partir de 3 capas del modelo OSI, entrando en detalle profibus proporciona 4 versiones diferentes de la capa física, donde principalmente se emplea una topología de bus con terminación en ambos extremos, permitiendo el acoplamiento y desacoplamiento seguro en las estaciones presentes. (Weigmann y Kilian, 2004)

Las principales versiones de capa física que se encuentra en Profibus son RS485 la cual es un protocolo de transmisión asíncrona con un rango de velocidades que oscila entre los 9.6 Kbit/s y 12 Mbit/s, generalmente el cable implementado en RS485, suele ser par trenzado apantallado para evitar posibles ruidos durante la comunicación. Además, permite conectar un máximo de 127 estaciones con una distancia máxima de 10 km utilizando repetidores; sin embargo, existe otra versión de capa física como Ethernet, ya que es muy accesible de utilizar

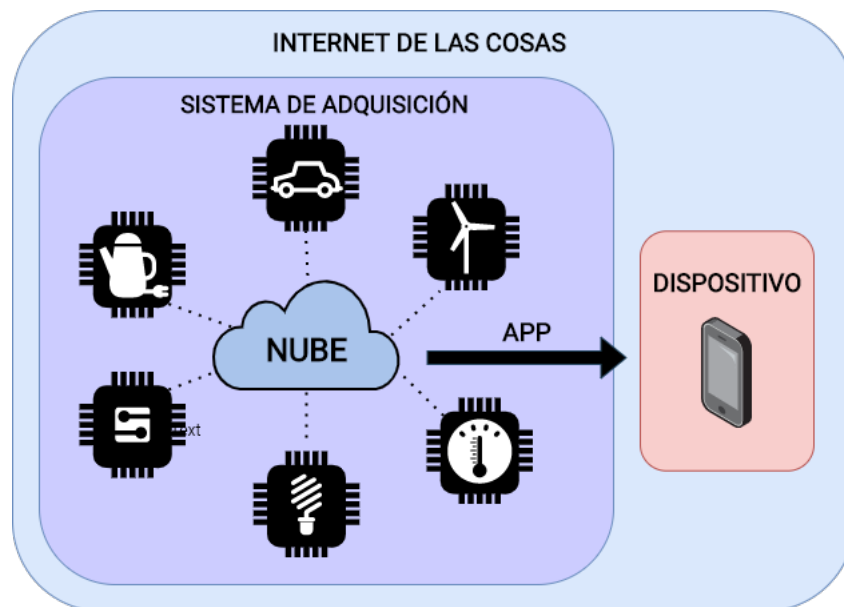
o implementar. El conector que generalmente se emplea para conectarse con cada estación es el DB9. (Weigmann y Kilian, 2004)

5.6. Funcionamiento IoT

La definición de IoT, según Rose (2015) considera que es la agrupación e interconexión de dispositivos y objetos a través de una red, donde todos ellos podrán ser visibles e interactuar por medio de esta. Puede concebirse como una infraestructura global que permite ofrecer servicios de monitorización mediante la conexión de objetos, ya sean físicos o virtuales. También se considera que es una conectividad de red y la capacidad de cómputo que se extienden a objetos, sensores que estén dentro de la industria 4.0, permitiendo que estos dispositivos intercambien datos sin la intervención del ser humano, como se aprecia en la figura 5. (Rose, 2015)

Figura 5

Internet de las Cosas (IoT)



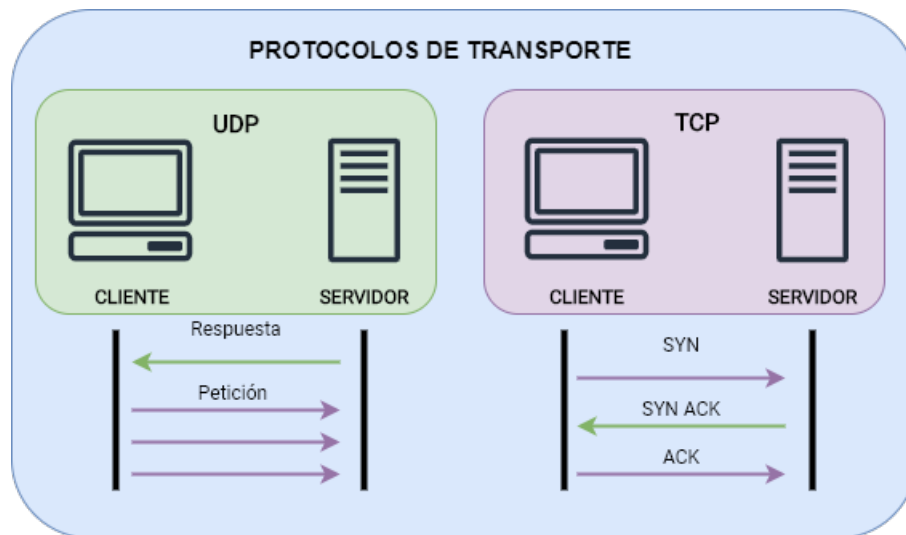
Nota: Estructura base de un sistema IoT.

Dentro de las comunicaciones IoT, la suite de protocolos TCP/IP proporciona una estrategia flexible y escalable en redes. Donde el protocolo IP proporciona direccionamiento y reenvío de paquetes individuales basándose en el origen y destino del paquete, por otro lado, TCP se encarga del proceso de transporte, teniendo en cuenta el flujo de datos, además de los

paquetes perdidos en el proceso, Cabe recalcar que no necesariamente en la capa de transporte se debe emplear TCP también existen alternativas que ofrecen una mayor velocidad como el protocolo UDP, En la figura 6 se observa la comparación de ambos protocolos.

Figura 6

Protocolo UDP vs TCP



Nota: Se aprecia la caracterización de los protocolos UDP y TCD respectivamente.

5.6.1. Modelos de conectividad

Según Rose (2015), el IoT utiliza diferentes modelos de conectividad, cada uno de ellos dependen de sus propias características, entre los modelos disponibles tenemos:

- Device-to-Device (dispositivo a dispositivo).
- Device-to-Cloud (dispositivo a la nube).
- Device-to-Gateway (dispositivo a puerta de enlace).
- Back-End Data-Sharing (intercambio de datos a través del back-end).

Cada uno de estos modelos demuestra la flexibilidad en la forma en que los objetos y dispositivos IoT pueden conectarse y proporcionar un valor para el usuario.

Dentro del IoT existen varios protocolos que intervienen en todo el proceso de envío y recepción de información para comprender el principio de funcionamiento del sistema de

adquisición, para eso se debe ir describiendo el funcionamiento de cada uno de protocolos que están interviniendo.

Protocolos:

- **HTTPS:** Es un protocolo de solicitud-respuesta que se basa en un texto sin estado donde se define la comunicación entre un cliente y un servidor sobre una conexión TCP. Según (Cirani, Ferrari, Picone, y Veltri, 2018) el protocolo HTTPS se caracteriza por:
 - Se utilizan direcciones URL para redireccionar a los recursos que conforman las solicitudes.
 - Los métodos son definidos para proporcionar el lenguaje de comunicación con la capa de aplicación sobre la operación que se está ejecutando en el servidor.
 - Tiene la posibilidad de agregar campos de encabezado para transmitir información por medio de mensaje y así procesar de forma correcta y proporcionada un lenguaje de comunicación.
 - Para informar al cliente sobre el resultado de la que la solicitud ha sido atendido, existen los códigos de estado donde proporciona respuestas de forma estándar, uniforme y descriptivo.

- **MQTT:** Message Queue Telemetry Transport o en español Transporte de Telemetría de Cola de Mensajes es un protocolo de código abierto que se basa en TCP. MQTT se dirige a entornos donde dispositivos y redes que se encuentran restringidas. Dentro del protocolo MQTT aplica filtros de suscripción al flujo de mensajes recibidos para determinar de manera eficaz a qué clientes se debe responder un mensaje. Por lo tanto, una suscripción se considera como una recepción en tiempo real y tiene un carácter no duradero. (Cirani y cols., 2018)

Su funcionamiento empieza cuando los clientes inician un conexión TCP/IP con un servidor central o broker, el cual mantiene un registro de usuarios en línea. Esta conexión se mantiene abierta hasta que el cliente la finaliza.

- **ZIGBEE:** La capa de red ZigBee proporciona la funcionalidad de la capa de red OSI, se basa en el protocolo de enrutamiento de malla a 802.15.4 lo que este protocolo ofrece 16 canales en 2.4 GHz, numerados del 11 al 26, por lo que, para acceder a la red no se puede reservar una banda de ancho ZigBee restringe las direcciones ID de PAN para establecer una comunicación con la red. Las redes zigBee permite a los nodos actualizar y formar rutas a una red, donde Hersent, Boswarthick, y Elloumi (2011) considera los

siguientes roles:

- **Coordinador:** Es un nodo encargado de crear la red, enrutar los paquetes y permitir las conexiones con otros nodos de la red.
- **Router:** Trabaja de manera parecida al coordinador donde los nodos actúan como router dentro de la red y tienen la capacidad de enrutar a los demás paquetes.
- **End Device:** Son los que tienen la capacidad de enviar y recibir paquetes a la red, pero no pueden cumplir la tarea de router, además tienen la facilidad de entrar en un modo sleep para disminuir el rendimiento y alargar la vida útil del router.

5.6.2. Plataformas Basadas en la Nube.

En el montaje de plataformas basadas en el internet de las cosas es necesario tener una base de datos que permita acceder a la información que constantemente se actualizara a partir de la lectura de los dispositivos de campo involucrados en la red IoT. Actualmente, existen varias plataformas que proporcionan este servicio de modo que cualquier persona puede rescatar información de estas bases de datos a partir de cualquier parte del mundo con acceso a internet, entre las plataformas más implementadas tenemos las siguientes.

- **Amazon Web Services (AWS):** Es una plataforma de servicios web que ofrece soluciones para computación, almacenamiento y redes, en diferentes capas de abstracción. Por ejemplo, puede utilizar el almacenamiento a nivel de bloque (un bajo nivel de abstracción) o un almacenamiento de objetos altamente distribuido (un alto nivel de abstracción) para almacenar sus datos. Utiliza estos servicios para alojar sitios web, ejecutar aplicaciones empresariales y extraer enormes cantidades de datos. Los servicios web son accesibles a través de Internet mediante el uso de protocolos web típicos (HTTP) y los utilizan las máquinas o los humanos a través de una interfaz de usuario. Los servicios más destacados proporcionados por AWS, son EC2 que ofrece máquinas virtuales y S3 que ofrece capacidad de almacenamiento. Los servicios en AWS funcionan bien juntos: puede usar estos servicios para replicar su configuración de red local existente o diseñar una nueva configuración desde cero. El modelo de precios de los servicios es de pago por uso.(Wittig y Wittig, s.f.)
- **Firebase:** Se trata de una plataforma web creada por Google, cuya principal función es desarrollar y facilitar la creación de apps de elevada calidad de una forma rápida, con el fin de que aumenta la base de usuarios y ganar más dinero. La plataforma

está importada en la nube y está disponible para diferentes plataformas como iOS, Android y web. Contiene diversas funciones para que cualquier desarrollador pueda combinar y adaptar la plataforma a medida de sus necesidades.(Yahiaoui, s.f.)

- **Microsoft Azure:** Azure es una plataforma informática en la nube de Microsoft, que proporciona una amplia variedad de servicios que se utilizan sin la necesidad de comprar ni aprovisionar su propio hardware. Azure permite el desarrollo rápido de soluciones y proporciona los recursos para realizar tareas que no son factibles en un entorno local. Los servicios de cómputo, almacenamiento, red y aplicaciones de Azure permiten concentrarse en crear excelentes soluciones sin preocuparse por cómo se ensambla la infraestructura física.(Shanan, 2015)

5.7. Realidad Aumentada

En la actualidad existen muchos campos de aplicación de la realidad aumentada, entre los cuales se puede destacar los siguientes: publicidad , información, educación o entretenimiento. la implementación de la realidad aumentada surge de la necesidad de implementar visualizaciones más complejas, tales como Objetos 3D superpuestos en la realidad. (Jara, 2020)

La realidad aumentada está basada en algoritmos de visión artificial, donde busca representar objetos virtuales 3D a partir del mismo punto de vista en cuál las imágenes de la escena real están siendo capturadas por cámaras de seguimiento. De igual forma, se debe considerar la necesidad de convergencia en un mapa estereoscópico de los sistemas inerciales y ópticos presentes en el dispositivo, básicamente se requiere un robusto sistema de seguimiento inercial, estos sistemas de seguimiento inercial comúnmente conocido como (IMU'S) durante el proceso de rastreo de imágenes la cámara debe estar sincronizada con el reloj de forma muy precisa. El sistema necesita saber exactamente qué lectura de IMU corresponde al comienzo de la captura del frame como también al final para determinar una correcta localización del objeto virtual. (Pangilinan, Lukas, y Vasanth, s.f.)

Según Furht (2011), en el proceso de renderización de objetos virtual hay que destacar la implementación de visión artificial; es decir, se ocupan diferentes métodos de visión por computadora, en su mayoría relacionados con el seguimiento de vídeo. Estos métodos suelen constar de dos etapas: seguimiento y reconocimiento. En la parte de seguimiento es común el uso de detección de características, detección de bordes o inclusive otros métodos de procesamiento de imágenes para la detección; sin embargo, se clasifica todos estos modelos en

dos principales clases “basado en características” o “basado en modelo”.

5.7.1. Basado en características

Los métodos basados en características consisten en descubrir la conexión entre las características de la imagen 2D y su marco mundial 3D coordenadas. Dentro de los modelos basados en características destaca World tracking que consiste básicamente en escenas a gran escala donde se requiere realizar el seguimiento de superficies horizontales planas en el entorno del usuario y realizar el método de Mapping para determinar la posición del objeto.

5.7.2. Basado en Modelo

Los métodos basados en modelos se caracteriza por usar como marco de referencia, modelos CAD o plantillas 2D del artículo basadas en distinguibles como por ejemplo modelos 3D o códigos QR. Para el tracking basado en modelo comúnmente se emplean el modelo de “seguimiento de imágenes” que se basa en imágenes 2D de referencia para superponer el objeto. Estos marcadores simples, pero visualmente distintos que se detectan fácilmente incluso con dispositivos de gama baja. Dichos marcadores son fácilmente detectables debido a sus detalles distintivos, bordes de alto contraste y una forma asimétrica; es decir, es una imagen fácilmente reconocible. De esta forma, el software de detección puede determinar qué imagen de marcador está a la vista y la orientación de la cámara en relación con el marcador en el espacio 3D. (Linowes, 2021)

Finalmente, La segunda etapa clave en realidad aumenta se basa en el reconocimiento, en esta etapa se utilizó los datos obtenidos de la primera etapa para reconstruir el sistema de coordenadas en el mundo real. si un punto tiene coordenadas $(x, y, z)^T$ en el marco de coordenadas de la cámara, su proyección sobre el plano de la imagen es $(x/z, y/z, 1)^T$. (Furht, 2011)

5.7.3. SDKs Para el desarrollo de realidad Aumentada

- **AR Core:** Es una plataforma de creación de aplicaciones de Realidad Aumentada para dispositivos móviles como Android. ARCore utiliza tres tecnologías principales para integrar contenido virtual en el mundo a través de la cámara. Utiliza seguimiento de movimiento, comprensión ambiental y estimación de luz. ARCore funciona rastreando la ubicación de un dispositivo a medida que se mueve y construyendo su propia comprensión del mundo real. Puede identificar puntos de interés y lecturas de los sensores del teléfono,

además determina la posición como la dirección del teléfono mientras está en movimiento. (Glover, 2018)

- **Vuforia:** Vuforia contiene uno de los SDK más usados en el desarrollo de realidad aumentada, dispone de muchas herramientas, como el reconocimiento de diferentes tipos de objetos visuales (como cajas, cilindros y planos), reconocimiento de texto y entorno mediante algoritmos de visión artificial, a partir de la combinación de imagen y código QR podemos rastrear objetos. Vuforia dispone de un Escáner de objetos en el cual escanea y crea objetivos de objetos. El proceso de reconocimiento se implementa utilizando la base de datos (almacenamiento local o en la nube). El complemento de Unity es fácil de integrar y muy potente. Todos los complementos y funcionalidades de la plataforma son de uso gratuito, pero incluyen las marcas de agua de Vuforia. (Glover, 2018)
- **AR Toolkit:** Específicamente, ARKit es un sistema VIO, con una detección de plano 2D. VIO rastrea la posición relativa de su dispositivo en el espacio en tiempo real; es decir, su pose se vuelve a calcular entre cada actualización de cuadro en su pantalla, unas 30 o más veces por segundo. Estos cálculos se realizan dos veces, en paralelo. Su pose se rastrea a través del sistema visual (cámara) haciendo coincidir un punto en el mundo real con un píxel en el sensor de la cámara en cada cuadro. Su pose también es rastreada por el sistema de inercia (su acelerómetro y giroscopio, la IMU). La salida de ambos sistemas se combina luego a través de un filtro Kalman que determina cuál de los dos sistemas está proporcionando la mejor estimación de su posición real"(verdad del terreno) y publica esa actualización de pose a través de ARKit SDK. Al igual que el cuentakilómetros de su automóvil registra la distancia que ha recorrido el automóvil, el sistema VIO rastrea la distancia que ha recorrido su iPhone en. (Pangilinan y cols., s.f.)

6. Marco metodológico

A continuación se presenta la metodología de la investigación que se utilizó en el proyecto técnico planteado. Está relacionado con: el enfoque, el alcance, el método y tipo de investigación; además, el proceso que se realizó para el desarrollo.

6.1. Diseño de un sistema de adquisición de las señales de los módulos Sorting and Handling para el monitoreo y visualización de datos en línea

6.1.1. Programación de PLC's mediante el Software de TIA PORTAL

Para la adquisición de las señales presentes en los módulos del MPS, se tuvo que cargar una programación lógica que cumpla con una emulación de una línea de producción. El PLC SIMATIC S7-300 cuenta con las respectivas direcciones para la comunicación con el PLC en función de trabajo con el módulo Sorting, en la tabla 1; y con el módulo Handling en la tabla 2 respectivamente, se aprecia a detalle las direcciones que se utilizaron para la identificación de los sensores y actuadores presentes en la programación.

Tabla 1

Direcciones IP del módulo Sorting

Dirección	Nombre
I 124.0	Sensor óptico de proximidad - PART_AV
I 124.1	Sensor inductivo de reflexión - B2
I 124.2	Sensor óptico de proximidad - B3
I 124.3	Sensor óptico de proximidad - B4
I 124.4	Sensor fin carrera 2B1
I 124.5	Sensor fin carrera 2B2
Q 124.0	Actuador de la banda
Q 124.1	Actuador de desvío de tapas 1 - color rojo
Q 124.2	Actuador de desvío de tapas 2 - color gris
Q 124.3	Actuador de retención de objetos
Q 125.5	Output para comunicación con el módulo Handling

Nota: *Se observa la identificación de las direcciones IP del PLC conectado a cada uno de los sensores y actuadores presentes en el módulo Sorting, la información ha sido tomada por los autores del proyecto.*

Tabla 2*Direcciones IP del módulo Handling*

Dirección	Nombre
I 124.0	Sensor óptico de proximidad - PART_AV
I 124.1	Sensor fin carrera - 1B1
I 124.2	Sensor fin carrera - 1B3
I 124.3	Sensor fin carrera - 1B2
I 124.4	Sensor fin carrera - 2B1
I124.5	Sensor fin carrera - 2B2
I124.6	Sensor óptico de proximidad - gripper
Q 124.0	Accionamiento actuador desplazamiento hacia la izquierda
Q 124.1	Accionamiento actuador desplazamiento hacia la derecha
Q 124.2	Accionamiento actuador desplazamiento vertical
Q 124.3	Actuador activación del Gripper
I 125.7	Input para comunicación con el módulo Sorting

Nota: Se observa la identificación de las direcciones IP del PLC conectado a cada uno de los sensores y actuadores presentes en el módulo Handling, la información ha sido tomada por los autores del proyecto.

A partir de las direcciones establecidas en el PLC se realizó la programación en TIA PORTAL que se visualiza en el ANEXO A, permitiendo que el módulo Handling recoja la tapa desde la banda transportadora hasta el módulo Sorting para que realice la clasificación correspondiente a cada color de tapa.

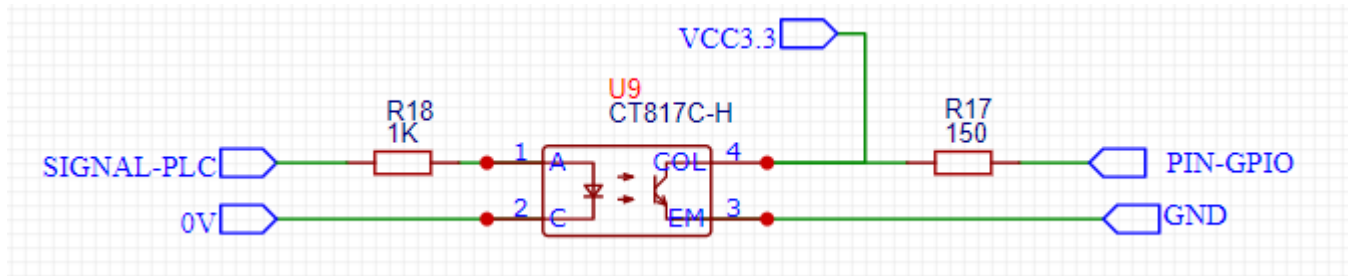
6.1.2. Diseño de un circuito electrónico para la comunicación mediante una CPU o System on a chip

Para obtener las señales y procesar a la nube se realizó un circuito electrónico para conectar los datos de los PLC de cada módulo a la tarjeta electrónica o CPU, debido a que cada PLC trabaja a 24V y 0.9A, y la tarjeta de adquisición que se seleccionó fue Raspberry Pi 3 para el desarrollo de este proyecto debido a que cuenta un procesador que trabaja a 2.4 GHz y tiene conectividad a Internet en su placa integrada y sus header GPIO funcionan a 3.3V y 0.04A. Identificando este cambio de tensión se observa en la figura 7 el esquema electrónico donde se implementó un circuito aislador para que la señal que es emitida por el PLC sea detectado

mediante la activación de un fototransistor, haciendo que se cierre el circuito permitiendo separar la activación del PLC y la emisión de señales con el voltaje y amperaje requerido para el Raspberry Pi 3.

Figura 7

Esquema Electrónico para señales del PLC al CPU



Nota: Se visualiza el esquema utilizado para la recepción de una señal captada del PLC a la tarjeta de adquisición o CPU.

Cabe mencionar que el mismo esquema electrónico planteado fue utilizado para cada una de las señales a monitorizar en línea.

6.1.3. Programación para procesamiento de señales en la nube

En esta sección, para el envío de las señales captadas del PLC a la nube se procedió a desarrollar un programa en Python, ya que es un software libre y cuenta con muchos paquetes y librerías que permiten comunicar a servidores como Firebase con mucha facilidad, el programa fue incorporado dentro del sistema operativo de la tarjeta de adquisición o CPU. A continuación se indica el procedimiento que se realizó:

1. **Definición de librerías:** Mediante el comando import se adjunta las librerías de Firebase permitiendo que la CPU se conecte con el Firebase, como se observa en la figura 8.
2. **Definición de variables a los pines del GPIO:** En la figura 8 se aprecia la declaración de las variables con su respectivo pin de GPIO que van a estar conectados mediante cable de bus de datos al esquema electrónico planteado anteriormente.

Figura 8

Código de programación para comunicación con Firebase

```
main.py > ...
1 import firebase_admin
2 from firebase_admin import credentials
3 from firebase_admin import db
4 import RPi.GPIO as GPIO
5 import time
6 cred = credentials.Certificate("claves-firebase.json")
7 GPIO.setmode(GPIO.BOARD)
8
9 #Pines del modulo Sorting
10 pin0 = 11 #I0 #PART-AV #SENSOR1
11 pin1 = 13 #I1 #B3 #SENSOR2
12 pin2 = 15 #I2 #B2 #SENSOR3
13 pin3 = 37 #I3 #B4 #SENSOR4
14 pin4 = 35 #Q0 #BANDA
15 pin5 = 33 #Q1 #ACTUADOR TAPAS ROJAS\
16 pin6 = 31 #Q2 #ACTUADOR TAPAS METALICAS
17 pin7 = 29 #Q3 #ACTUADOR STOP
18
19 #Pines del modulo Handling
20 pin1_1 = 12 #I1 # FIN CARRERA 1B1
21 pin1_2 = 16 #I2 # FIN CARRERA 1B3
22 pin1_3 = 18 #I6 # SENSOR GRIPER
23 pin1_4 = 22 #Q0 # DESPLAZAMIENTO IZQUIERDO
24 pin1_5 = 36 #Q1 # DESPLAZAMIENTO DERECHA
25 pin1_6 = 38 #Q2 # BAJA GRIPPER
26 pin1_7 = 40 #Q3 # ACTUADOR GRIPPER
```

Nota: Se aprecia el código de programación para importar librerías de Firebase con Python.

3. **Definición de los pines de la tarjeta de adquisición como inputs:** Luego de importar las librerías que permite comunicar Python con Firebase, se procedió a definir la variable declarada con el correspondiente como inputs para la lectura de datos, observando en la figura 9.

Figura 9

Código de programación para definir como entradas a los pines del CPU

```
27 #Configuración GPIO
28 GPIO.setup(pin1_1, GPIO.IN)
29 GPIO.setup(pin1_2, GPIO.IN)
30 GPIO.setup(pin1_3, GPIO.IN)
31 GPIO.setup(pin1_4, GPIO.IN)
32 GPIO.setup(pin1_5, GPIO.IN)
33 GPIO.setup(pin1_6, GPIO.IN)
34 GPIO.setup(pin1_7, GPIO.IN)
35 GPIO.setup(pin0, GPIO.IN)
36 GPIO.setup(pin1, GPIO.IN)
37 GPIO.setup(pin2, GPIO.IN)
38 GPIO.setup(pin3, GPIO.IN)
39 GPIO.setup(pin4, GPIO.IN)
40 GPIO.setup(pin5, GPIO.IN)
41 GPIO.setup(pin6, GPIO.IN)
42 GPIO.setup(pin7, GPIO.IN)
```

Nota: Se visualiza el código de programación para configurar los pines GPIO de la tarjeta de adquisición o CPU.

4. **Creación de funciones para enviar datos:** Para la emisión de datos se definió 3 tres funciones, el cual, mediante con el URL otorgado por Firebase, permitió la comunicación y el envío de los datos correspondientes de los módulos Sorting and Handling respectivamente. Se observa en la figura 10 como se declaró las funciones para clasificar las señales de cada módulo trabajado.

Figura 10

Código de programación con funciones para clasificar información

```
45 firebase_admin.initialize_app(cred,{
46     'databaseURL': 'https://thesis-8cf59-default-rtdb.firebaseio.com/'
47 })
48 def enviar_sorting(entrada,dato):
49     users_ref = db.reference('/RASPBERRY-1/sorting/')
50     users_ref.update({
51         str(entrada): bool(dato)
52     })
53 def enviar_handling(entrada,dato):
54     users_ref = db.reference('/RASPBERRY-1/handling/')
55     users_ref.update({
56         str(entrada): bool(dato)
57     })
58 def enviar_informacion(entrada,dato):
59     users_ref = db.reference('/RASPBERRY-1/informacion/')
60     users_ref.update({
61         str(entrada): int(dato)
62     })
63 )
```

Nota: Se visualiza el código de programación para definir funciones que permita ordenar las variables obtenidas de cada módulo.

5. **Creación de condiciones para procesar los datos:** Luego de definir todos los parámetros que competen para la emisión de las señales captadas del PLC, se creó una condición el cual indica cuando el pin definido como input detecte un “0” lógico, este inmediatamente envía la palabra “True” a la plataforma de Firebase; de igual manera, cuando el pin detecte un “1” lógico se envía la palabra “False”. Se aprecia en la figura 11 la programación para la configuración de la condición.

Figura 11

Código de programación para condicionar valores de input lógicos

```
89     if sensor1 == 0:
90         enviar_sorting('I0',True)
91     if sensor1 == 1:
92         enviar_sorting('I0',False)
93
94
95     if sensor2 == 0:
96         enviar_sorting('I1',True)
97     if sensor2 == 1:
98         enviar_sorting('I1',False)
99
100
101    if sensor3 == 0:
102        enviar_sorting('I2',True)
103    if sensor3 == 1:
104        enviar_sorting('I2',False)
105
```

Nota: Se observa el código de programación para establecer la condición de recepción de datos a subir en la nube de 3 variables.

Cabe mencionar que la misma lógica de condición planteada fue utilizado para cada una de las señales a monitorizar en línea.

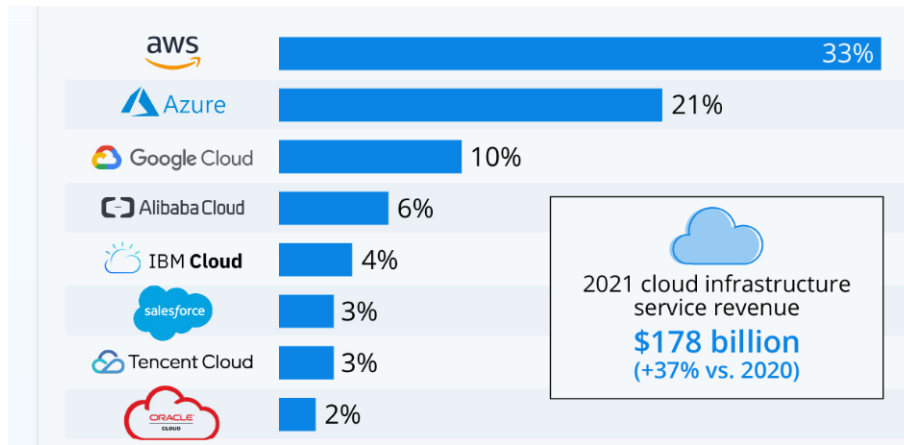
6.2. Configuración de la base de datos en la nube para el acceso de las aplicaciones

6.2.1. Selección de la plataforma basada en cloud Computing

Para el presente trabajo de titulación existieron tres posibles alternativas para la implementación de una base de datos basada en la nube, las cuales existen: aws, azure y Firebase, en el planteamiento de estas alternativas se tomó como referencia la tendencia actual del mercado donde destacan las 3 antes mencionadas tómesese como referencia la figura 12.

Figura 12

Plataformas de nube



Nota: Se aprecia la tendencia actual de plataformas basadas en la nube. (Richter, 2022)

La tendencia de plataformas basadas en la nube apunta a AWS como la plataforma con mayor demanda; sin embargo, para este trabajo de titulación se implementó la plataforma de Google denominada Firebase, esto debido a que presenta mayor prestación al momento de realizar el montaje de prototipos mediante el plan gratuito por consumo limitado que ofrece.

Firebase proporciona una interfaz sumamente amigable al momento de la creación de proyectos, cuenta con una documentación muy basta para la implementación en diferentes plataformas de desarrollo tales como Android o API web.

AWS y Azure también disponen de sus respectivas plataformas con prestaciones similares a Firebase; sin embargo, la documentación y el proceso de vinculación de estas plataformas requiere de mayor complejidad, además no disponen de todas las librerías que vinculen con todas las aplicaciones que se implementó en este proyecto. Finalmente, cabe recalcar que el plan gratuito de AWS solo está disponible por 8 meses, mientras que Firebase dispone de un plan ilimitado a partir de las tarifas limitadas de descarga y escritura que ofrece.

6.2.2. Configuración de firebase

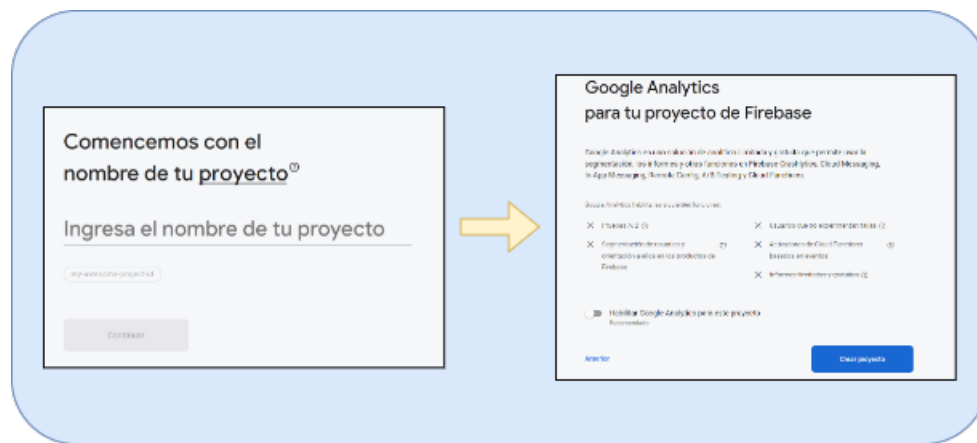
El sistema de monitorización planteado se basó en la implementación de protocolos de comunicación HTTPS para el proceso de envío y descarga de datos actualizados a la base de datos. Estos protocolos son principalmente empleados por el sistema de monitorización con

realidad aumentada y también por la aplicación web. Firebase dispone de un servicio denominado Firebase Realtime, este servicio, a diferencia de las solicitudes HTTPS, implementa la sincronización de información para detectar cambios en la base de datos. Estos cambios automáticamente se actualizarán tanto en la aplicación web como en la aplicación de realidad aumentada. Por lo que los primeros pasos que se realizó dentro de la configuración de la nube comienza con la creación del proyecto y selección de los servicios como Realtime Database para la base de datos en tiempo real. (Yahiaoui, s.f.)

La interfaz de creación de proyectos que proporciona Firebase permite configurar el proyecto siguiendo los pasos e indicaciones que sugiere, tomando como referencia la figura 13. Después de la creación del proyecto se vinculó las aplicaciones a Firebase, tanto como apps móviles, web o de escritorio. Finalmente, se generó las credenciales que se deben incluir en cada una de las diferentes aplicaciones.

Figura 13

Creación de proyecto Firebase

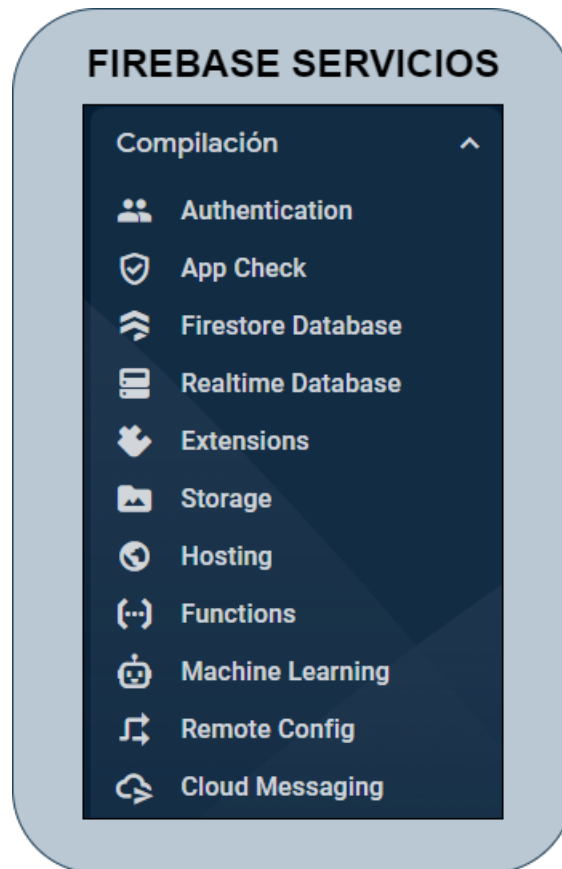


Nota: *Se visualiza el proceso de creación del proyecto en firebase.*

Firebase dispone de diferentes servicios que se emplean en cada una de las aplicaciones que dispongan, tal y como se observa en la figura 22, los principales servicios que se implementaron son los de Realtime Database para el envío de datos en tiempo real desde el módulo MPS hasta la nube como también para la descarga de las diferentes apps del sistema. Finalmente, se empleó el servicio de hosting para desplegar la aplicación web en internet con la finalidad de acceder a los datos desde cualquier dispositivo con conexión a internet.

Figura 14

Servicios de Firebase



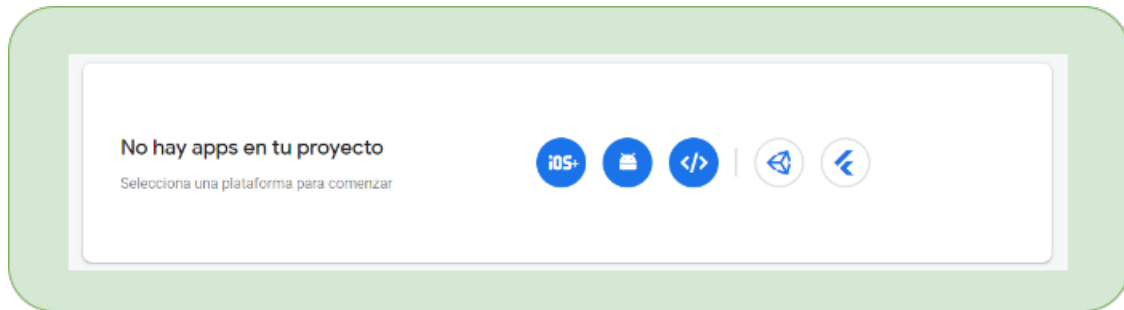
Nota: *Servicios disponibles en la plataforma firebase.*

6.2.3. Configuración de la aplicación Web

La aplicación web permite visualizar datos en línea mediante una interfaz gráfica que proporciona información acerca de los datos existentes en la nube “Firebase Realtime”; sin embargo, para que acceda a la base de datos se vinculó la aplicación web a Firebase. Para ello, proporciona la opción de agregar diferentes aplicaciones escritas en varios lenguajes de programación como JavascRipt, Go, kotlin, c o Python a Firebase.

Figura 15

Vinculación de aplicaciones a firebase



Nota: *Aplicaciones , disponibles para la vinculacion con firebase.*

Después de agregar la aplicación web se generan una serie de certificaciones que se incluyó en la configuración del proyecto, tomando como referencia la figura 16, estas configuraciones permitió acceder a los diferentes servicios que dispone Firebase. Para agregar las credenciales a la aplicación web, se copió en un archivo de configuración escrito en JavaScript, cabe recalcar que antes de agregar el archivo de configuraciones proporcionado por Firebase primero se requiere tener instalado el paquete con las librerías necesarias para que la aplicación pueda interactuar con los servicios de Firebase. Para la instalación de las librerías necesarias que se requieren, además se implementó el gestor de paquetes de Node.js, La versión de Firebase instalada es la 9.8.1.

Figura 16

Credenciales de conexión a Firebase.



The screenshot shows a code editor with three tabs: 'npm', 'CDN', and 'Configuración'. The 'npm' tab is active. Below the tabs, there is a text block stating: 'If you're already using [npm](#) and a module bundler such as [webpack](#) or [Rollup](#), you can run the following command to install the latest SDK:'. Below this is a code block with the command: '\$ npm install firebase'. Below the command, there is another text block: 'Then, initialize Firebase and begin using the SDKs for the products you'd like to use.'. Below this is a code block with the following JavaScript code:

```
// Import the functions you need from the SDKs you need
import { initializeApp } from "firebase/app";
// TODO: Add SDKs for Firebase products that you want to use
// https://firebase.google.com/docs/web/setup#available-libraries

// Your web app's Firebase configuration
const firebaseConfig = {
  apiKey: "AIzaSyDtVF3eaUV6dqCPXw1qhQPF8VZFeB1xdOA",
  authDomain: "thesis-8cf59.firebaseio.com",
  databaseURL: "https://thesis-8cf59-default-rtdb.firebaseio.com",
  projectId: "thesis-8cf59",
  storageBucket: "thesis-8cf59.appspot.com",
  messagingSenderId: "85349863650",
  appId: "1:85349863650:web:e2a902ec857bc8c3130ebb"
};

// Initialize Firebase
const app = initializeApp(firebaseConfig);
```

Nota: *Se observa las credenciales en formato JSON para agregar al proyecto.*

Después de agregar la aplicación web, se agregó la app realizada en Unity, este proceso es similar al antes implementado; sin embargo, se debe considerar algunos aspectos como el registro de la aplicación web mediante la asignación de un dominio con el cual será identificado como se aprecia en la figura 17, después de registrar el dominio se requiere instalar el SDK de Firebase disponible para Unity, para ello, Firebase proporciona un link de descarga donde se obtiene el SDK para instalarlo en Unity.

Además de la integración del SDK de Firebase, se empleó los archivos con las configuraciones en formato JSON, estas configuraciones permitieron acceder a los diferentes servicios que proporciona. En este caso, para realizar el proceso de conexión con la nube se tuvo que implementar Realtime Firebase para la actualización de las etiquetas con información actualizada y obtenida de los sensores y actuadores presentes en los módulos.

Figura 17

Registro de la aplicación web.



Nota: *Se visualiza el proceso de registro de la aplicación web*

6.2.4. Configuración de Firebase Realtime

Firebase Realtime es un servicio que permite crear una base de datos propia, para leer y escribir información con eventos en tiempo real, esta base de datos permite estructurarla desde el panel de administración de Firebase para crear manualmente los campos con los que va a trabajar, para ello se dividió la base de datos en 3 ramificaciones principales las cuales se designaron como Sorting, Handling e información, como se observa en la figura 18.

Figura 18

Estructura de la base de datos.

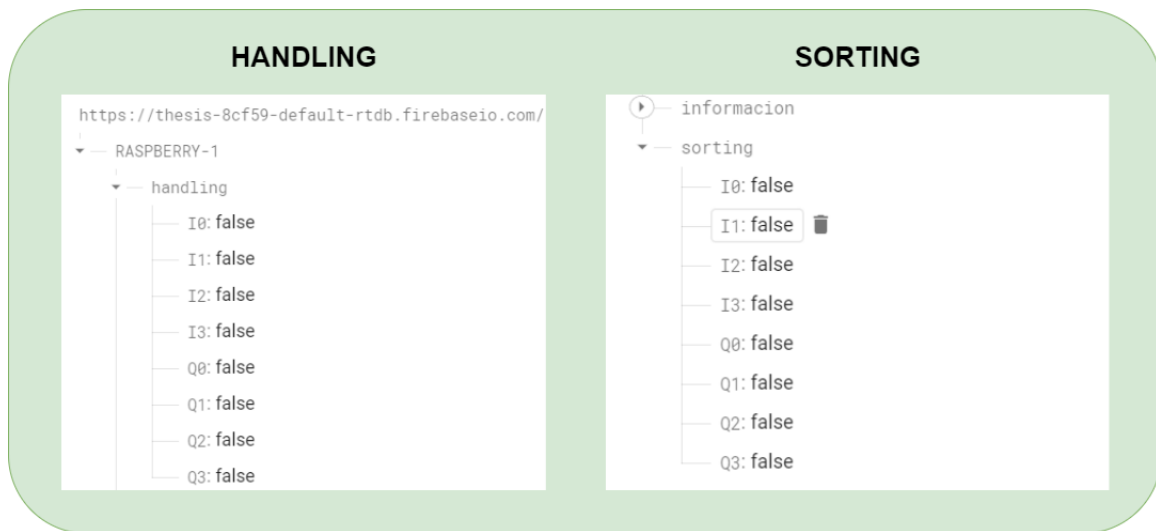


Nota: *Ramificación de la base de datos.*

En las ramificaciones Sorting y Handling se encuentra declarado las variables pertenecientes a este módulo, para ello se realizó una medición de las señales presentes en el módulo y en función de la cantidad existente se estructuró la base de datos, como resultado la figura 19 hace referencia a la cantidad de variables declaradas para cada uno de los módulos.

Figura 19

Variables de los módulos en Realtime Firebase.

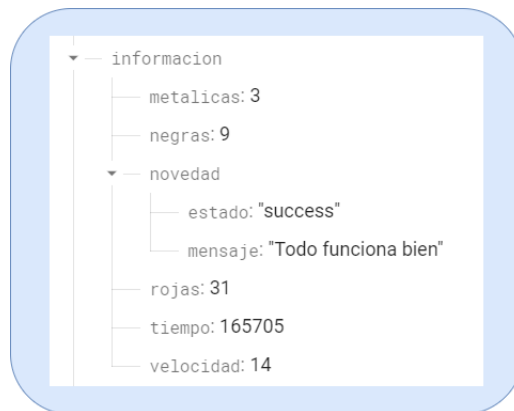


Nota: Se observan de las variables consideradas para cada uno de los módulos

En la sección de información, se planteó colocar todos los datos relacionados con la producción, como la cantidad de tapas clasificadas según el índice de reflectividad de los colores: rojo, negro o metálico, como se aprecia en la figura 20. Posteriormente, se actualizaron los datos en la aplicación web y se añadió un mensaje de alerta con un campo de tipo texto que brinda información acerca del estado de funcionamiento de la planta.

Figura 20

Variables de información de los módulos.

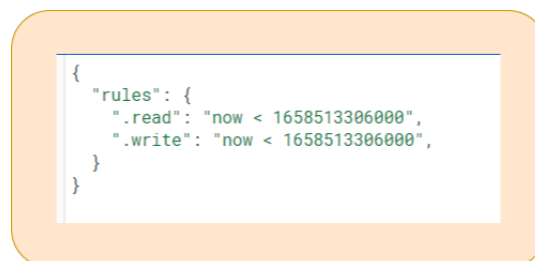


Nota: *Datos de interes de ambos modulos.*

Adicionalmente, se configuró las reglas de seguridad para la escritura de la base de datos, tal que el proceso se realizó editando las fechas límites de escritura y de lectura, estas fechas están dentro de un archivo “JSON” y reciben el valor de la fecha límite en formato EPOCH, como se observa en la figura 21.

Figura 21

Reglas de seguridad en Firebase Realtime.



Nota: *Fechas de habilitacion de escritura y lectura de la base de datos.*

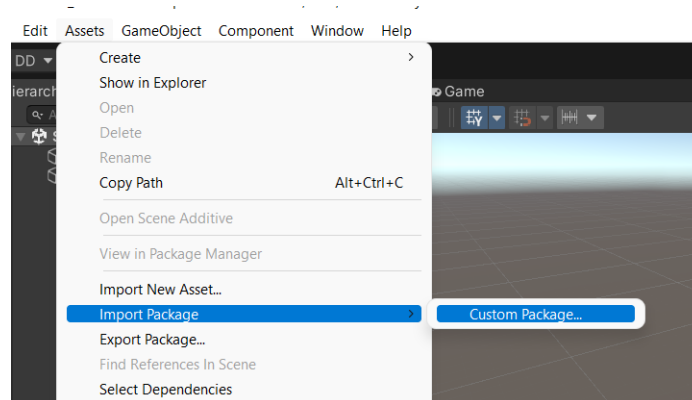
6.2.5. Incorporación del SDK de Firebase a Unity

El proceso de incorporación de Firebase en el software de Unity empieza con la descarga del SDK e importación del mismo, para ello dentro de la aplicación se escogió la opción de assets y se importa un paquete personalizado, este paquete proporciona Firebase para la implementación de los servicios que ofrece. El SDK fue descargado de la plataforma oficial

para la fecha de desarrollo de este trabajo de titulación, la versión más reciente de este paquete es la 9.0.2, estable para versiones de Unity 2021.3.0f1 , y posteriores.

Figura 22

Incorporación del paquete de firebase a unity

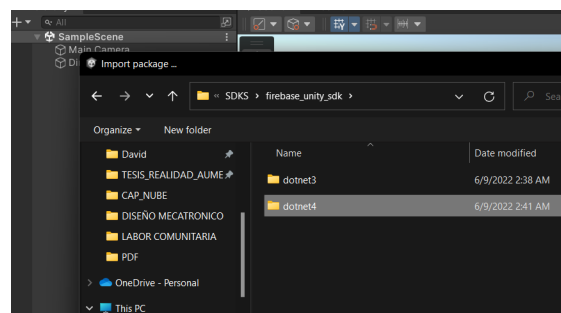


Nota: Se observa la incorporación del paquete de firebase al proyecto en unity.

El SDK de Firebase ofrece dos opciones .NET compatibles con las versiones de Unity, siguiendo la documentación oficial que Unity 5.x y las versiones anteriores usan .NET Framework 3.x, de modo que se debe importar el paquete dotnet3. por otro lado, Unity 2017.x y las versiones posteriores permiten usar .NET Framework 4.x. por lo tanto, en este caso se importó la carpeta dotnet4 como se aprecia en la figura 23.

Figura 23

Selección de Dotnet al proyecto.



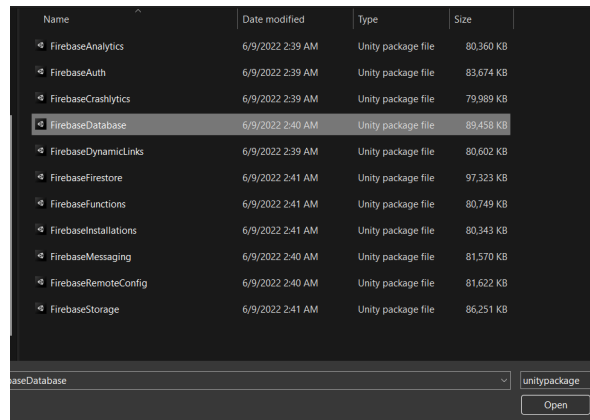
Nota: Se observa las versiones de dotnet, disponibles para implementar.

En la carpeta dotnet4 se encuentran diferentes paquetes de Unity que hacen referencia a cada uno de los servicios que se implementa dentro de la aplicación, tal y como se aprecia en

la figura 24, el servicio que se seleccionó fue “Firebase Database” este servicio ira actualizando los valores de las variables que se visualizaran en realidad aumentada.

Figura 24

Incorporación de la librería de Firebase realtime



Nota: *Se observa el proceso incorporación de la librería de firebase realtime*

Adicionalmente, se agregó las credenciales de Firebase para que la aplicación en Unity acceda a los servicios de la nube, estas credenciales al igual que en el caso de la aplicación web están en formato JSON y son proporcionadas al momento de vincular una aplicación a Firebase, para agregarlas se importó el archivo “google-services.json” como un asset personalizado. Es importante mencionar que si este archivo no está en la carpeta del proyecto, no podrá validar las peticiones que se haga a la nube.

Para el proceso de obtención de información proveniente de Firebase se creó una corrutina que permita realizar una petición a la base de datos, sin afectar el bucle principal que está ejecutando la aplicación de realidad aumentada, esta petición se basó en las solicitudes HTTP, GET, POST o UPDATE, en este caso se realizó una petición get para obtener los datos y actualizar los valores dentro de la realidad aumentada, en la figura 25 se evidencia el fragmento de código encargado de realizar todo esto antes mencionado.

Figura 25

Función encargada de tomar datos de Firebase

```
FirebaseDatabase.DefaultInstance.GetReference("RASPBERRY-1").Child("informacion").GetValueAsync().ContinueWithMainThread(task => {
    if (task.IsFaulted)
    {
        Debug.Log("no funciona read");
    }
    else if (task.IsCompleted)
    {
        DataSnapshot snapshot = task.Result;

        // HACEMOS REFERENCIA AL VALOR DE LA VARIABLE QUE QUEREMOS LEER EN LA BASE DE DATOS
        val5 = snapshot.Child("novedad").Child("mensaje").Value.ToString();

        // ASIGNAMOS EL VALOR OBTENIDO DE EL DATO NOVEDAD EN FIREBASE
        novedad.text = val5;
    }
});
```

Nota: *Se observa el código encargado de actualizar la base de datos.*

6.3. Diseño de una aplicación móvil y página web para el monitoreo y visualización de datos en línea a través de Realidad Aumentada

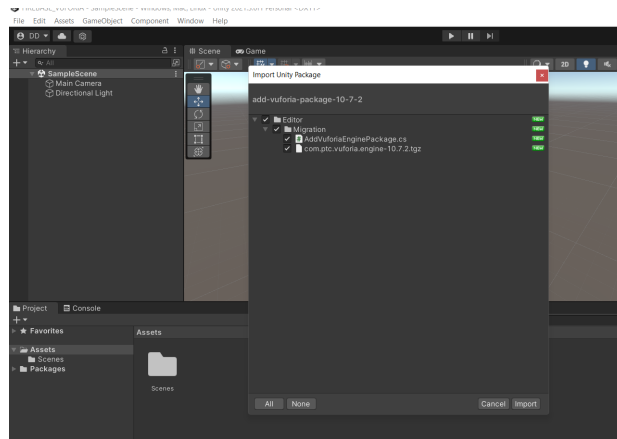
En el proceso de generación de realidad aumentada se analizaron 3 posibles alternativas para la implementación, cada plataforma ofrece diferentes opciones de rastreo de imágenes y a su vez implementan tecnologías relacionadas con visión artificial, se destacó a Vuforia como plataforma base para la creación de esta aplicación.

Vuforia permite el reconocimiento basado en modelos, donde mediante una imagen 2D podemos posicionar el elemento de realidad aumentada, este tipo de reconocimiento está catalogado dentro de Vuforia como Vumarks, cabe recalcar que plataformas como AR Core y AR toolkit también disponen de este tipo de tecnología, sin embargo, el proceso de implementación de las imágenes rastreables en Vuforia es mucho más sencillo de implementar en comparación a las otras plataformas, también es factible mencionar que en el caso de AR toolkit las aplicaciones solo estarían disponibles para iOS, por otro lado, con AR Core tendríamos desarrollo netamente optimizado para android, es por este principal motivo que se implementó vuforia debido a la variedad de plataformas con las que trabaja.

Se realizó una aplicación móvil utilizando el software Unity con la finalidad de implementar realidad aumentada durante el proceso de monitorización de las variables, para ello, se descargó el SDK de Vuforia Engine de la página oficial y se agregó la librería, como se observa en la figura 26, la versión del SDK de Vuforia implementado es la 10.7.2 y fue descargada de la página oficial.

Figura 26

Importacion del SDK de vuforia.

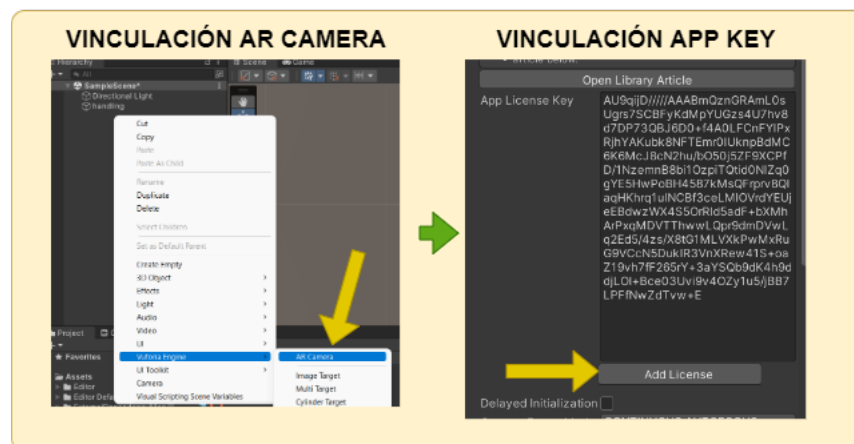


Nota: *Instalacion del SDK de vuforia, a Unity*

Después de agregar el SDK de Vuforia se vinculó la cámara de realidad aumentada, esta cámara permitió implementar el motor de realidad aumentada de Vuforia para el mapeo de objetos 3D, cabe mencionar que se vinculó la AR camera con la clave de licencia que proporciona la página oficial de Vuforia tal y como se observa en la figura 27.

Figura 27

Vinculación AR camera



Nota: *Vinculacion de ar camera con la llave proporcionada por vuforia.*

Después de crear la AR camera, se creó los image targets encargados de permitir el rastreo de la ubicación de los objetos 3D para su generación dentro del campo de visión de la cámara.

En la creación de los image targets se generó códigos QR que sirven como objetivo para desplegar la realidad aumentada, para ello se incorporó el objeto image target y se asoció a un código QR que será responsable de indicar la información necesaria del proceso, cada uno de estos image targets deben estar vinculados a la Ar Camera y tener las siguientes configuraciones:

- **Posición:** La posición del image target permitió tener una idea acerca de que tan alejado o cerca están las etiquetas con información acerca del proceso.
- **Default Observer Event Handler:** Este objeto ofrece varias opciones de seguimiento de imágenes de acuerdo a varios eventos por los cuales pasa el objetivo que está siendo rastreado, uno de ellos es la opción tracked la cual indica que la realidad aumentada va a estar desplegándose siempre y cuando se esté enfocando el imageTarget con la cámara, también ofrece funciones cuando se deja de enfocar el image target para llamar a correr otras rutinas dentro del programa.
- **Imagen:** Es muy importante que al momento de escoger la imagen que va a ser rastreada por la aplicación, esta permita tener varias características que sean fácilmente rastreables por el motor de Vuforia, por este motivo se implementó el uso de códigos QR, como se puede observar en la figura 28.

Figura 28

Configuración Image Target



Nota: *Configuración de los Image Target.*

6.3.1. Diseño de etiquetas con información 3D

La aplicación de realidad aumentada fue diseñada con el propósito de desplegar información visual en 3D cuando captura un objetivo que fue declarado anteriormente en los image target; sin embargo, hay que tener en cuenta que al momento de desplegar la información se consideró la posición espacial de las etiquetas dentro del campo de visión de la cámara, para ello, se procedió mediante la estimación de distancias de los objetos físicos junto con la de los virtuales, esto con la finalidad de ajustar las etiquetas en la posición más cercana al elemento del cual se quiere saber la información. Observar la figura 29 para identificar los image target.

Figura 29

Etiquetas de Información



Nota: *Se visualizan las etiquetas para desplegar información acerca de la planta.*

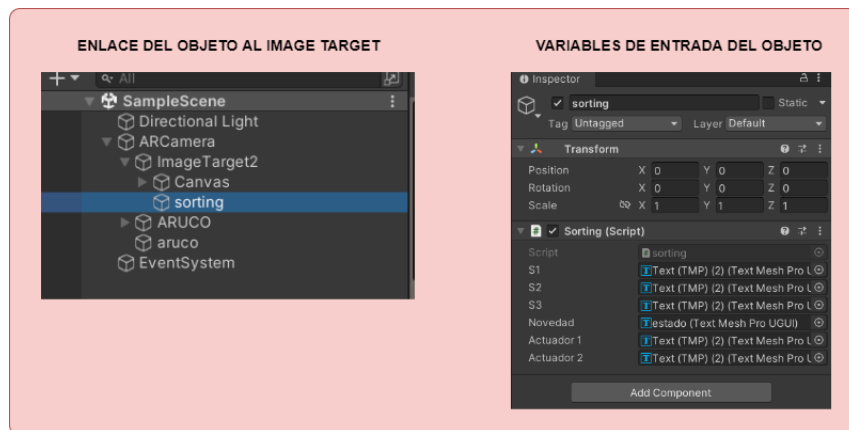
En el desarrollo de las etiquetas se utilizó el paquete de librerías UI que ofrece Unity, para los textos se implementó TextMeshPro y para el contenedor de las etiquetas se empleó un marco canvas como se aprecia en la figura 29, adicionalmente se colocó estratégicamente para que estén sobrepuestos sobre el actuador o sensor del cual se requiere visualizar la información.

6.3.2. Actualización de las etiquetas con la información de Firebase

Durante el proceso de actualización de las variables en las etiquetas se creó objetos vinculados a cada uno de los imageTarget, estos objetos a su vez están vinculados a un script encargado de actualizar los datos de las etiquetas basándose en las variables de entrada a las cuales están relacionadas con los valores de la etiqueta, para ello, se unió el nombre del campo de la etiqueta con los valores de entrada de los objetos tomando como referencia la figura 30.

Figura 30

Objeto de los Image Target



Nota: Se observan los objetos creados para los Image Targets.

El script encargado de actualizar los datos se basó en la implementación de corrutinas que se ejecutan al momento de detectar el image target con la cámara, estas corrutinas hacen llamadas a Firebase para posteriormente obtener la información pertinente.

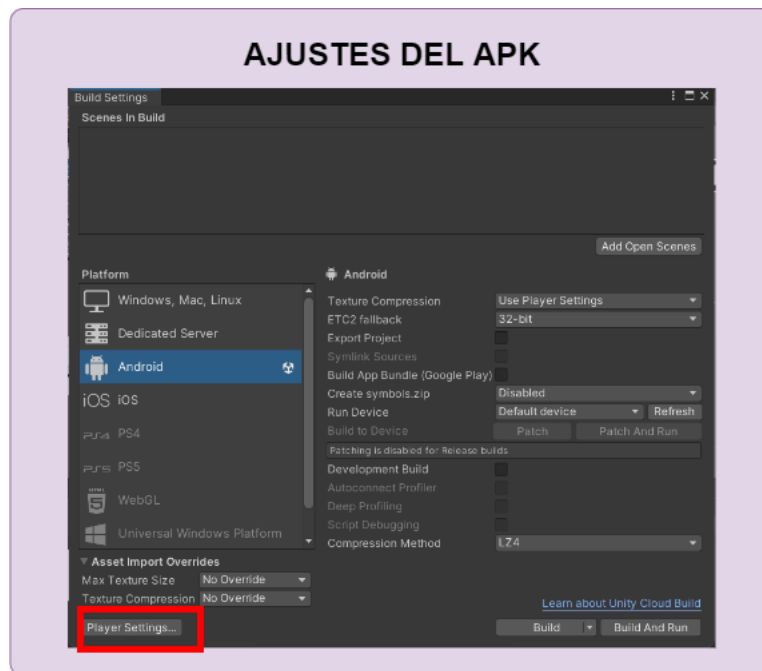
6.3.3. Elaboración del APK para dispositivos Android

En la elaboración del APK primero se configuró en la opción de ajustes de construcción los siguientes parámetros:

- **API Level:** Este parámetro indica las versiones para las cuales estará disponible la aplicación, para el presente caso de estudio se consideró como requisito mínimo API level 24.
- **Eliminación Vulkan:** Esta es una API multiplataforma de baja sobrecarga para gráficos 3D de alto rendimiento. Proporciona herramientas para crear gráficos de alta calidad en tiempo real en las aplicaciones; sin embargo, actualmente genera conflictos con el motor Vuforia por lo cual fue necesario renderizar.
- **Package Name:** El nombre del paquete de una aplicación de Android identifica de manera única su aplicación en el dispositivo, en Google Play Store y en las tiendas de Android de terceros compatibles este nombre coincide con el especificado en la configuración Firebase, tómesese como referencia la figura 31.

Figura 31

Ajustes de la configuración en Android



Nota: Ajustes de la aplicación para android.

6.3.4. Diseño de una aplicación web para la visualización de los datos mediante Firebase

La interfaz web proporciona datos acerca del funcionamiento del proceso, en la elaboración de la aplicación web se consideró todas las variables que están involucradas en ambos módulos, en total se utilizaron 15 variables de los módulos, 7 para el módulo Handling y 8 para el Sorting cabe mencionar que la aplicación web consta de 3 vistas principales que presentan información sutil acerca del caso de estudio.

Toda la aplicación web fue desarrollada con el framework React desde cero utilizando como base de datos no relacional a Firebase, entre los módulos implementados para el desarrollo de la aplicación están:

- **React Router:** react router es una librería que permite trabajar con URL para la navegación entre vistas dentro de la aplicación web y cuenta con hooks tales como useNavigate para cambiar de vistas a partir del manejo de eventos dentro de la app.
- **Material UI:** Es una librería que contiene diseños de componentes pre-programados

siguiendo el estilo visual Material, que es comúnmente implementado en aplicaciones Android, simplifica al proceso de diseño de la app mediante widgets programados con la funcionalidad deseada.

- **ReactStrap:** React Strap es la versión de bootstrap optimizada para su uso con React, cuenta con todos los recursos que proporciona bootstrap para agilizar la etapa de diseño.
- **Chart JS:** Chart JS facilita el proceso de elaboración de gráficos para la visualización de datos provenientes de Firebase, esta librería ofrece Widgets ya diseñados para su posterior utilización, a partir de datos se pasan mediante props, entre los gráficos implementados en este trabajo de titulación se tiene la gráfica de barras para la visualización de la cantidad de tapas que se han clasificado

6.3.5. Diseño de las Vistas de la aplicación Web

- **Vista de Inicio:** La vista de inicio indica varias fotografías acerca del laboratorio, el propósito de esta vista es dar una bienvenida a los usuarios mediante fotografías del laboratorio, está constituida por un widget carrusel proporcionado por ReactStrap que indica fotos del laboratorio, adicionalmente aparece un mensaje de bienvenida como se puede observar en la fig.
- **Vista de Información:** En la sección de información se presenta definiciones acerca de lo implementado en el proyecto, entre los temas explicados tenemos conceptos acerca de realidad aumentada, información relacionada al sistema de producción modular (MPS) y el sistema de adquisición desarrollado.
- **Vista de Dashboard:** El Dashboard se diseñó de tal manera que la información visual del funcionamiento de cada sensor y actuador, se consideró principalmente que las señales recibidas por el sistema de adquisición son señales booleanas, por lo que fue necesario diseñar un componente en React que permita visualizar de manera adecuada los valores de las variables.

6.4. Evaluación del desempeño de funcionamiento del sistema de monitoreo y visualización de las señales de los módulos del laboratorio de MPS

Para la evaluación del funcionamiento del sistema IoT implementado en el módulo Sorting and Handling del laboratorio de MPS, se realizó una prueba externa, el cual, se basa la

interacción de usuarios manipulando el módulo y visualizando la información de los sensores y actuadores a través de una aplicación web y realidad aumentada, para obtener criterios cuantitativos.

Se realizó una encuesta planteada en base a los beneficios que proporciona la implementación de un Sistema IoT para monitorización de información a 19 personas con el fin de obtener un análisis de la experiencia generada. Se evaluó aspectos puntuales como: la evidencia de la conexión de los módulos a internet, eficiencia de la velocidad de respuesta de la aplicación web y la realidad aumentada, información legible de las etiquetas y qué tipos de información son considerados a visualizar a través de las herramientas planteadas.

A continuación se presenta las preguntas que se planteó para la evaluación del funcionamiento:

- ¿Se logró evidenciar la conexión de los módulos a internet?
- ¿Que tan eficiente en términos de velocidad de respuesta considera que tiene la aplicación de realidad aumentada y web?
- ¿La información proporcionada por las etiquetas son legibles?
- ¿Qué otros tipos de información considera que se necesita visualizar en las aplicaciones realizadas?
- ¿Considera que el tiempo de actualización de los valores es moderado?

7. Resultados

7.1. Diseño de un sistema de adquisición de las señales de los módulos Sorting and Handling para el monitoreo y visualización de datos en línea

En la presentación de resultados de la adquisición de señales del módulo Sorting and Handling del laboratorio de MPS se presenta la tabla 3 de las señales obtenidas en forma de resumen con su conectividad a la tarjeta electrónica Raspberri pi 3 a través del esquema eléctrico diseñado, como se observa en la figura 32 y 33 de cada módulo respectivamente, comprobando que tuvo éxito la adquisición de las señales escogidas a ser monitorizadas.

Tabla 3*Señales adquiridas del módulo Sorting and Handling a monitorizar*

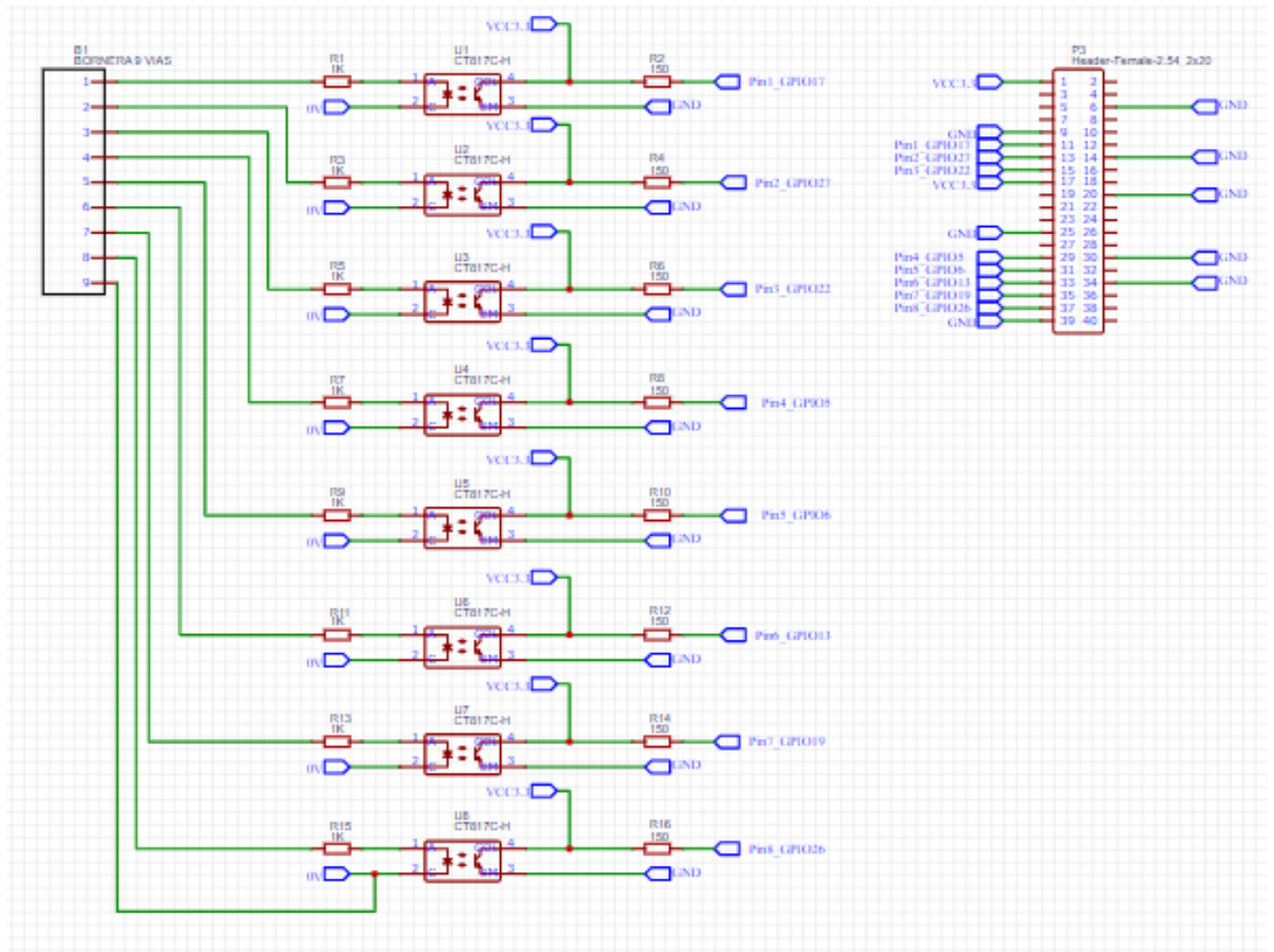
Señal	Dirección	Designación	Módulo	Tipo de Conectividad	Verificación de conexión a internet
I1	124.1	Sensor óptico de proximidad	Handling	Cable de bus de datos a optoacoplador	Correcta
I2	124.2	Sensor óptico de proximidad	Handling	Cable de bus de datos a optoacoplador	Correcta
I6	124.6	Sensor óptico de proximidad	Handling	Cable de bus de datos a optoacoplador	Correcta
Q0	124.0	Actuador neumatico	Handling	Cable de bus de datos a optoacoplador	Correcta
Q1	124.1	Actuador neumatico	Handling	Cable de bus de datos a optoacoplador	Correcta
Q2	124.2	Actuador neumatico	Handling	Cable de bus de datos a optoacoplador	Correcta
Q3	124.3	Actuador neumatico	Handling	Cable de bus de datos a optoacoplador	Correcta
I0	124.0	Sensor óptico de proximidad	Sorting	Cable de bus de datos a optoacoplador	Correcta
I1	124.1	Sensor inductivo de reflexión	Sorting	Cable de bus de datos a optoacoplador	Correcta
I2	124.2	Sensor óptico de proximidad	Sorting	Cable de bus de datos a optoacoplador	Correcta
I3	124.3	Sensor óptico de proximidad	Sorting	Cable de bus de datos a optoacoplador	Correcta
Q0	124.0	Actuador neumatico	Sorting	Cable de bus de datos a optoacoplador	Correcta
Q1	124.1	Actuador neumatico	Sorting	Cable de bus de datos a optoacoplador	Correcta
Q2	124.2	Actuador neumatico	Sorting	Cable de bus de datos a optoacoplador	Correcta
Q3	124.3	Actuador neumatico	Sorting	Cable de bus de datos a optoacoplador	Correcta

Nota: *Se visualiza el nombre con su respectiva dirección de cada una de las señales a monitorizar para la conectividad mediante la tarjeta electronica para ser procesada a internet.*

El diseño del circuito electrónico para que el cambio que tensión producida por el PLC a la tarjeta de adquisición o CPU cumplió con lo requerido del objetivo.

Figura 32

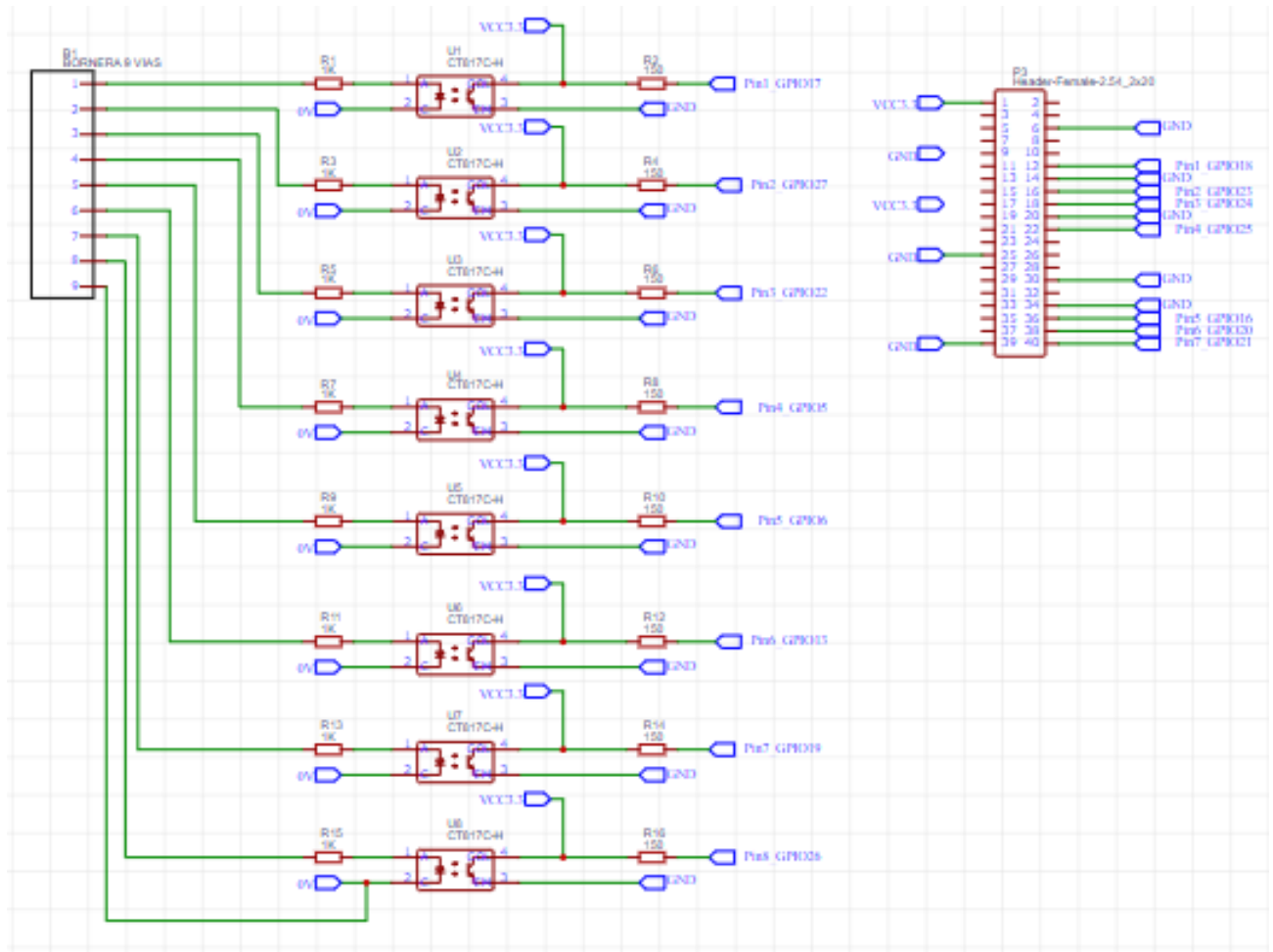
Esquema Electrónico para conexión del PLC a Raspberry pi 3 para el módulo Sorting



Nota: Se observa el esquema que se implemento para la adquisicion de señales del PLC al Raspberry Pi 3 del módulo Sorting.

Figura 33

Esquema Electrónico para conexión del PLC a Raspberry pi 3 para el módulo Handling



Nota: Se observa el esquema que se implemento para la adquisicion de señales del PLC al Raspberry Pi 3 del módulo Handling.

7.2. Configuración de la base de datos en la nube para el acceso de las aplicaciones

En este apartado se presenta las distintas imágenes que tuvo como resultado el momento en que se verificó el cambio de variables presentes en la línea de producción del MPS 500 y se van atenuando tanto en el PLC como en la plataforma de Firebase según el proceso que realizó cada módulo, en la figura 34 se observa el momento en que el módulo Handling detecta una variación de funcionamiento en su proceso y se visualizó de manera exitosa en la

base de datos. De igual manera se observa la alteración de los sensores y actuadores de lo módulo Sorting en la figura 35.

Figura 34

Variación de estado de señales del módulo Handling



Nota: Se observa el cambio de estado de las señales proporcionadas por el módulo Handling.

Figura 35

Variación de estado de señales del módulo Sorting



Nota: Se observa el cambio de estado de las señales proporcionadas por el módulo Sorting.

7.3. Diseño de una aplicación móvil y página web para el monitoreo y visualización de datos en línea a través de Realidad Aumentada

Los resultados obtenidos de la conexión de Firebase tanto en la página web y la aplicación móvil tuvieron una respuesta exitosa debido a que las señales fueron adquiridas y subidas a la nube de manera correcta, la plataforma de Firebase se conectó a la página web en el cual se muestra información sobre los autores del proyecto y sobre el tema del proyecto desarrollado, como se observa la figura 36 y un Dashboard referenciado en la figura 37, en el cual se visualizara toda la información en tiempo real de acuerdo con el funcionamiento de los módulos del MPS 500, creando así a lo que se define como un sistema IoT. Además de una interfaz dando la bienvenida a la página como se aprecia en la figura 38.

Figura 36

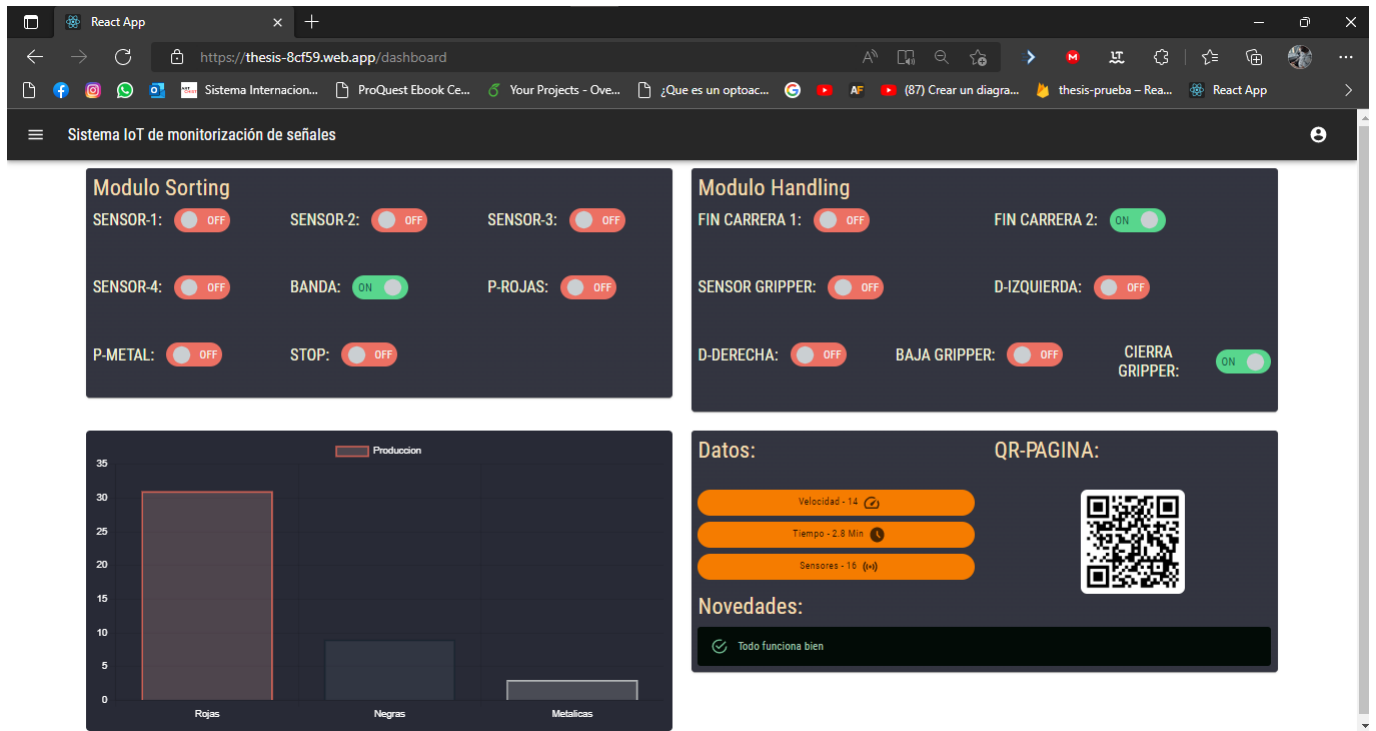
Información del proyecto visualizado en la página web



Nota: Se aprecia la interfaz de la información del proyecto realizado a través de la página web.

Figura 37

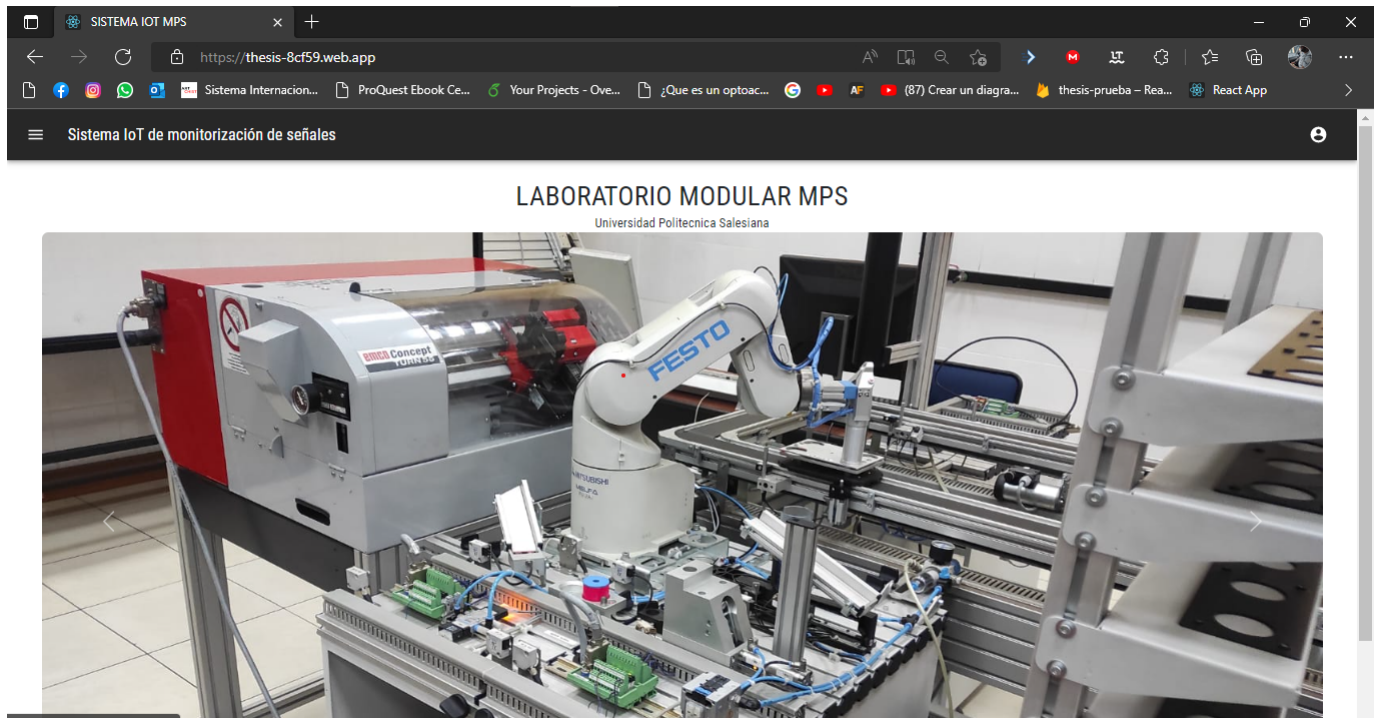
Información del Dashboard visualizado en la página web



Nota: Se aprecia la interfaz del Dashboard con toda la información a monitorizar.

Figura 38

Visualización del Home de la página web

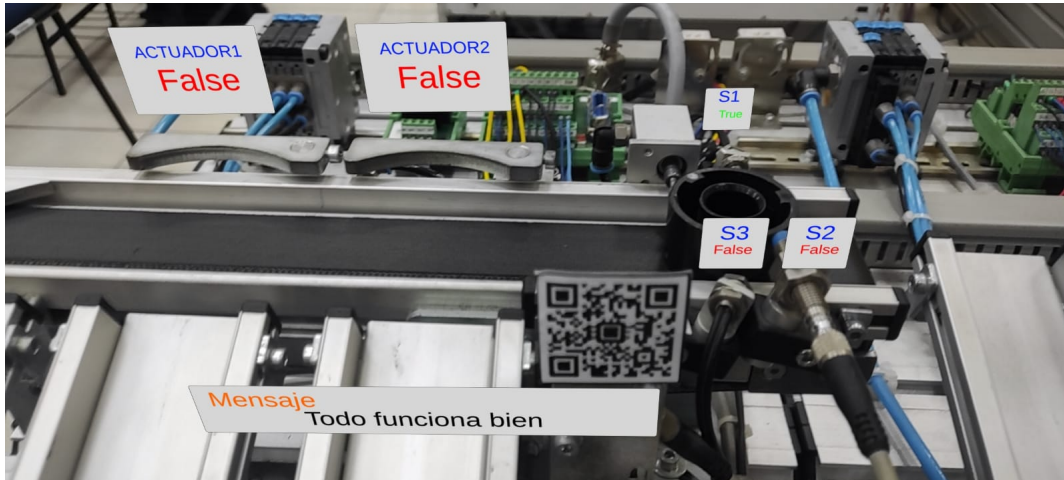


Nota: *Se observa la interfaz del Home de la página web.*

En cuanto a los resultados obtenidos de la aplicación móvil, mostrando las señales de los sensores y actuadores presentes en los módulos y de acuerdo a su funcionamiento a través de realidad aumentada, se observa en la figura 39 como el cambio de estado de los sensores varía dependiendo al tipo de tapa que está al inicio del proceso del módulo Sorting de manera exitosa. En la figura 40 se visualiza la activación de sensores y el actuador de clasificación al estar presente una tapa de color metálico, de igual manera en la figura 41 se observa cuando existe la presencia de una tapa de color rojo.

Figura 39

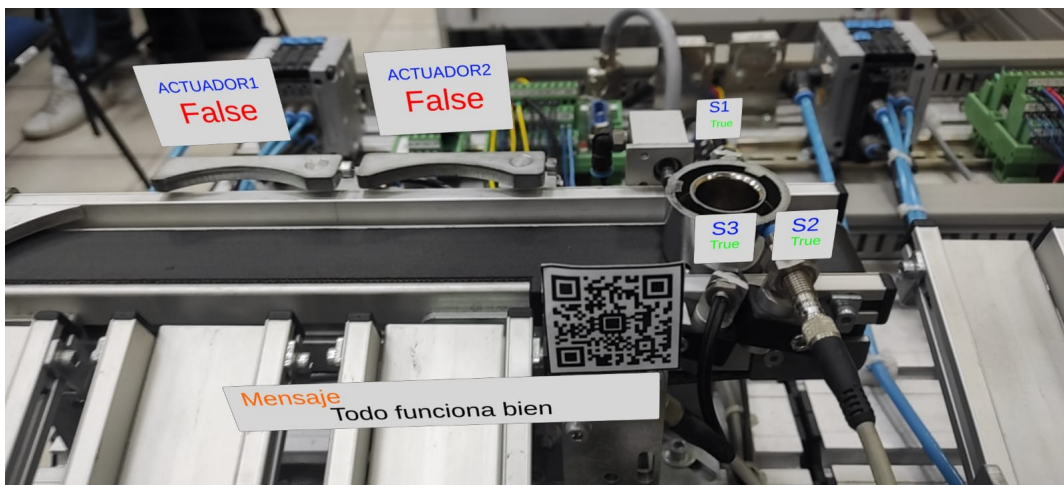
Variación de estado de sensores y actuadores del módulo Sorting con realidad aumentada una tapa de color negro



Nota: *Se visualiza el cambio de estado de las señales de sensores al presenciar una tapa de color negro.*

Figura 40

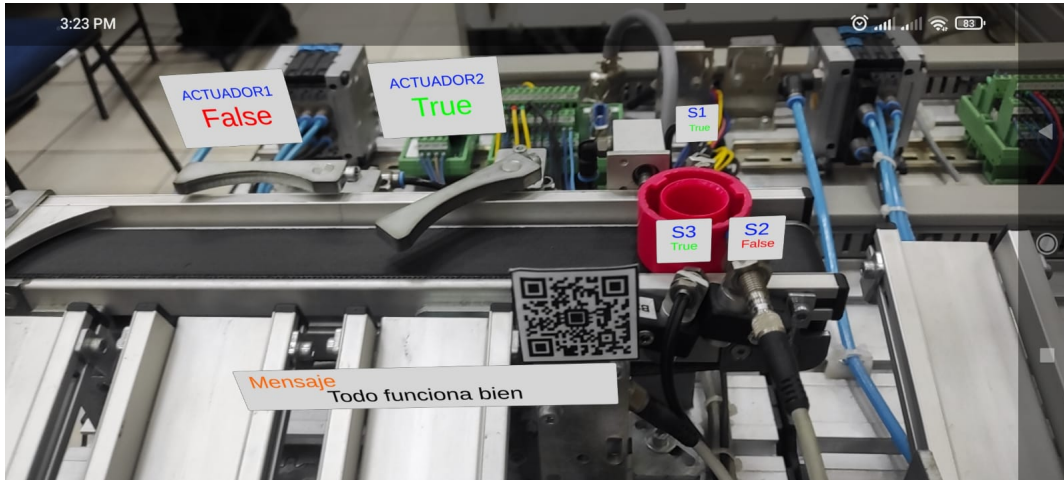
Variación de estado de sensores y actuadores del módulo Sorting con realidad aumentada una tapa de color metálico



Nota: *Se visualiza el cambio de estado de las señales de sensores al presenciar una tapa de color metálico.*

Figura 41

Variación de estado de sensores y actuadores del módulo Sorting con realidad aumentada una tapa de color rojo

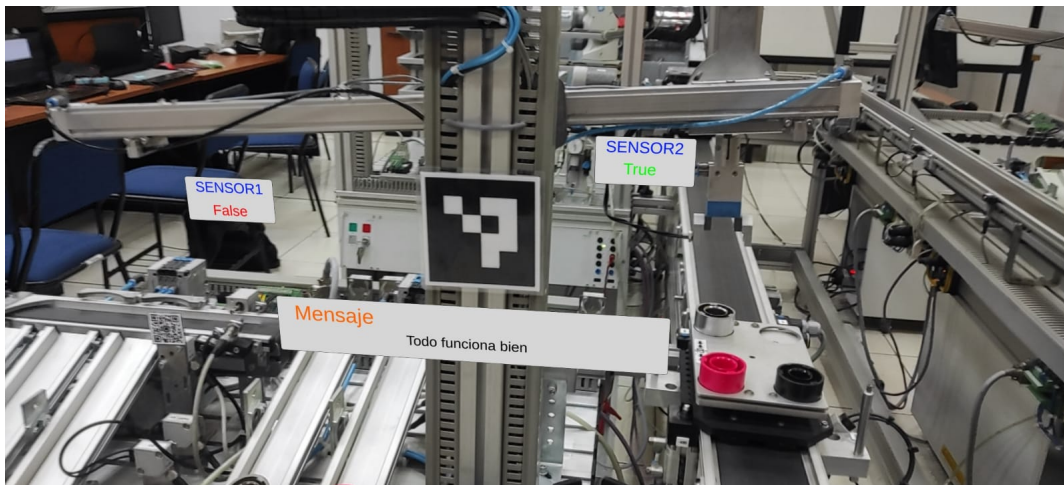


Nota: *Se observa el cambio de estado de las señales de sensores al presenciar una tapa de color rojo.*

De igual manera, tomando como referencia la figura 42 se observa como el posicionamiento de módulo Handling detecta la activación del Sensor fin carrera 1B1 con realidad aumentada y del mismo modo se observó la activación del Sensor fin carrera 1B3 como se aprecia en la figura 43.

Figura 42

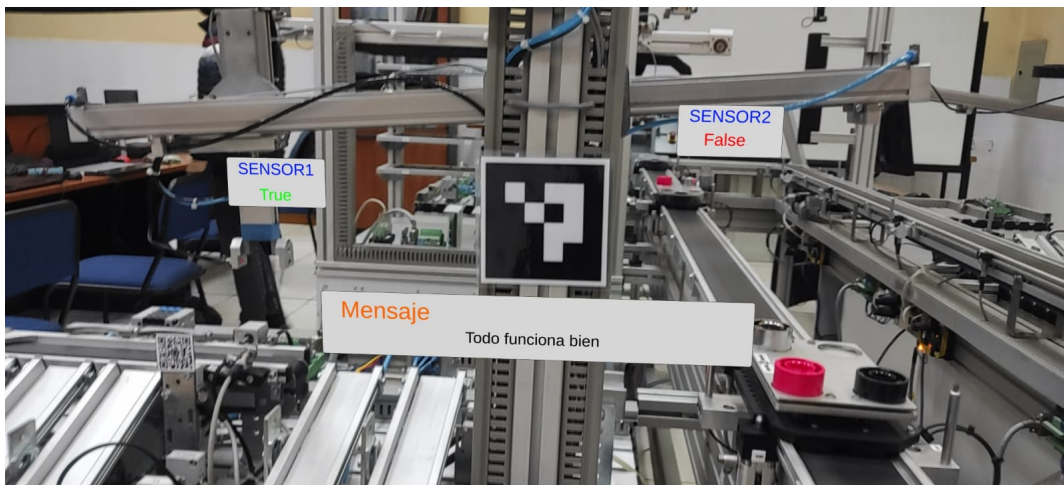
Variación de estado del sensor 1B1 con realidad aumentada



Nota: Se observa el cambio de estado de las señales proporcionadas por el módulo Handling con realidad aumentada.

Figura 43

Variación de estado del sensor 1B3 con realidad aumentada



Nota: Se observa el cambio de estado de las señales proporcionadas por el módulo Handling con realidad aumentada.

7.4. Evaluación del desempeño de funcionamiento del sistema de monitoreo y visualización de las señales de los módulos del laboratorio de MPS

Los resultados obtenidos de la evaluación del funcionamiento del sistema IoT implementado para monitoreo del módulo Sorting and Handling sirvieron para determinar si el sistema es o no funcional.

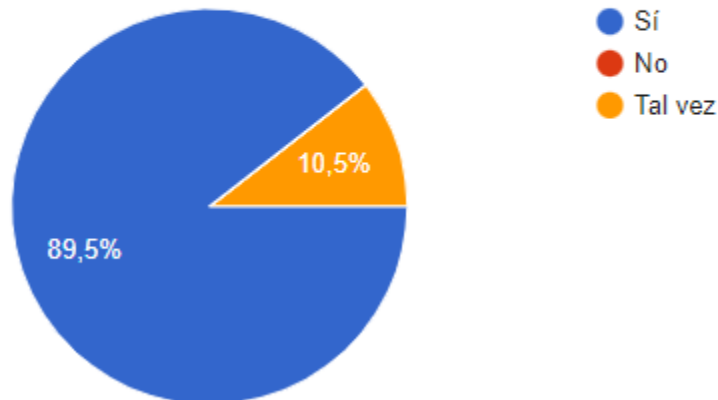
- En la pregunta 1: **¿Se logró evidenciar la conexión de los módulos a internet?** de las 19 personas evaluadas, el 89.5% (17 personas) indicaron que si se evidenció la conexión a internet, mientras que el 10.5% (2 personas) indican que no se logró evidenciar la conectividad de manera rápida o no pudieron observar adecuadamente debido a la ubicación del CPU; en cambio, existe un 0% de personas que identificaron un fallo en la conexión. Se observa en la figura 44 los resultados obtenidos de esta pregunta.

Figura 44

Resultado de evaluación de pregunta 1

¿Se logró evidenciar la conexión de los módulos a internet?

19 respuestas



Nota: Se visualiza una gráfica especificando los resultados en porcentaje de la primera pregunta planteada.

- En la pregunta 2: **¿Que tan eficiente en términos de velocidad de respuesta considera que tiene la aplicación de realidad aumentada y web?** de las 19 personas evaluadas, el 31.6% (6 personas) indica que la velocidad de respuesta del sistema IoT es excelente, el 42.1% (8

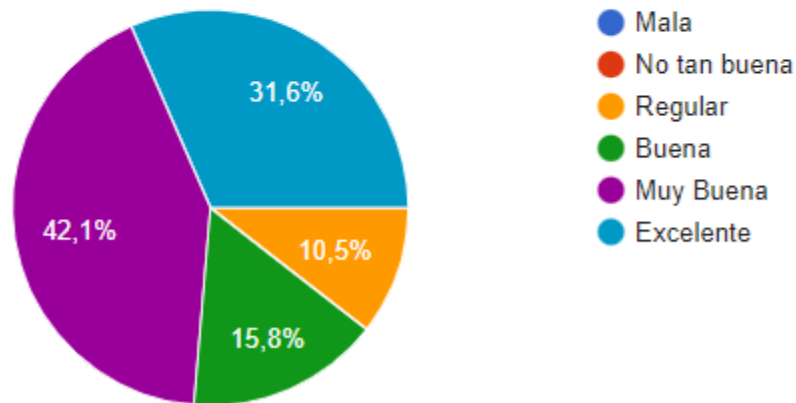
personas) que es muy bueno, el 15.8% (3 personas) es bueno y el 10.5% (2 personas) que es regular, mientras que existe un 0% de opciones de no tan bueno y malo la velocidad de respuesta del sistema implementado. Se observa en la figura 45 los resultados obtenidos de esta pregunta.

Figura 45

Resultado de evaluación de pregunta 2

¿Que tan eficiente en términos de velocidad de respuesta considera que tiene la aplicación de realidad aumentada y web?

19 respuestas



Nota: Se observa una gráfica especificando los resultados en porcentaje de la segunda pregunta planteada.

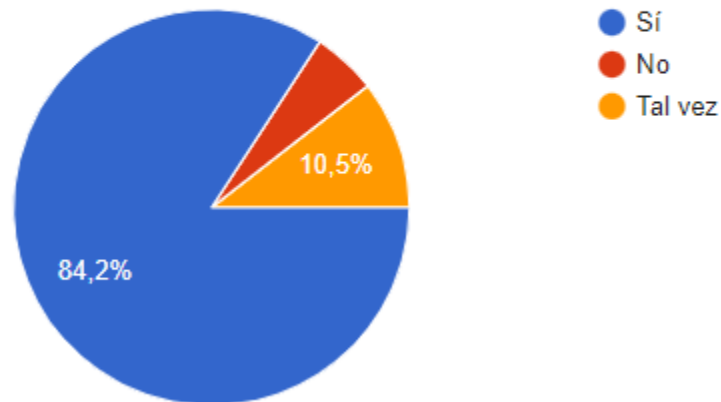
- En la pregunta 3: **¿La información proporcionada por las etiquetas son legibles?** de las 19 personas evaluadas, el 84.2% (16 personas) afirman que las etiquetas mostradas a través de realidad aumentada son entendibles y nítidas para la visualización del usuario, mientras que el 5.3% (1 persona) indica que no lo son; en cambio, existe un 10.5% (2 personas) que consideran que la visualización de las etiquetas es visible pero no en su mejor ni peor calidad. Se observa en la figura 46 los resultados obtenidos de esta pregunta.

Figura 46

Resultado de evaluación de pregunta 3

¿La información proporcionada por las etiquetas son legibles?

19 respuestas



Nota: Se observa una gráfica especificando los resultados en porcentaje de la tercera pregunta planteada.

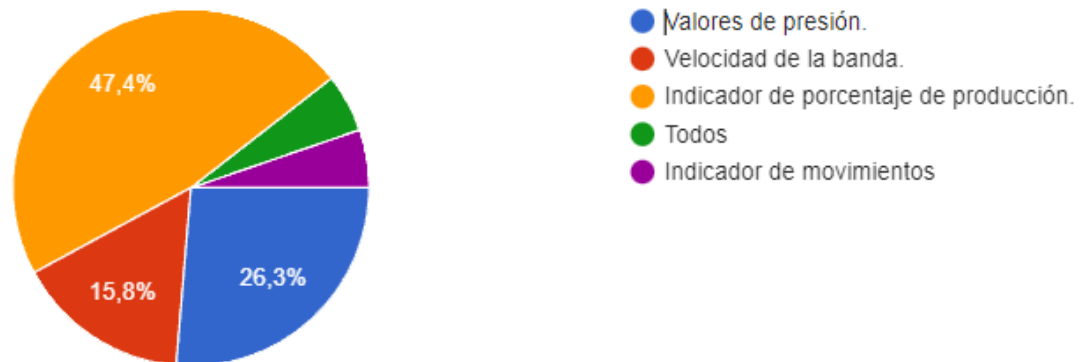
- En la pregunta 4: **¿Qué otros tipos de información considera que se necesita visualizar en las aplicaciones realizadas?** de las 19 personas evaluadas, el 47.4% (9 personas) consideran que debería ser visualizado un indicador de porcentaje de producción, el 26.3% (5 personas) indican que debería ser visualizado valores de presión, el 15.8% (3 personas) consideran que debe ser visualizado la velocidad de la banda, el 5.3% (1 persona) considera que debe existir un indicador de movimiento para cada acción y para finalizar existe un 5.3% (1 persona) que considera que deben ser visualizados todas las opciones mencionadas. Se observa en la figura 48 los resultados obtenidos de esta pregunta.

Figura 47

Resultado de evaluación de pregunta 4

¿Qué otros tipos de información considera que se necesita visualizar en las aplicaciones realizadas?

19 respuestas



Nota: Se observa una gráfica especificando los resultados en porcentaje de la cuarta pregunta planteada.

- En la pregunta 5: **¿Considera que el tiempo de actualización de los valores es moderado?** de las 19 personas evaluadas, el 89.5% (17 personas) indican que el tiempo de actualización de datos es el moderado, teniendo en cuenta el tipo de proceso que se realizó para la adquisición de datos, en cambio, el 10.5% consideran que el tiempo no exagera ni es demasiado lento. Se observa en la figura ?? los resultados obtenidos de esta pregunta.

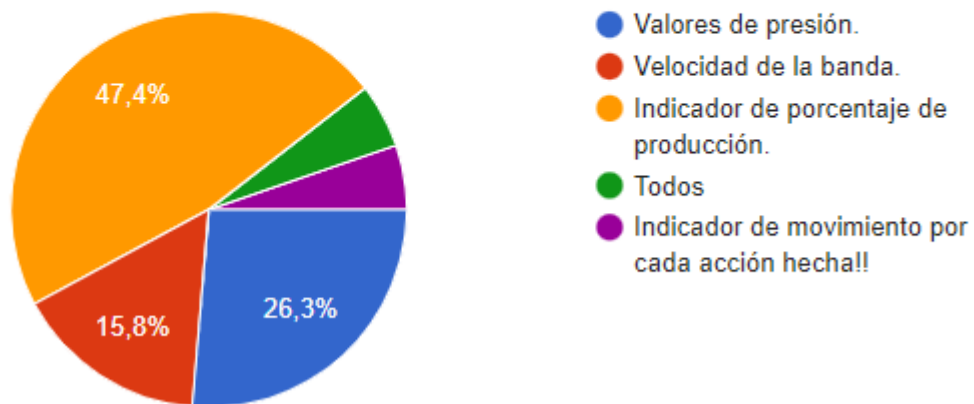
Figura 48

Resultado de evaluación de pregunta 5

¿Qué otros tipos de información considera que se necesita visualizar en las aplicaciones realizadas?



19 respuestas



Nota: Se observa una gráfica especificando los resultados en porcentaje de la quinta pregunta planteada.

7.5. Análisis de costos

El análisis de costos es el proceso de identificación de los recursos necesarios para llevar a cabo el proyecto. Determina la calidad y cantidad de recursos necesarios. Entre otros factores, se analizó el costo del proyecto en términos de dinero. Está enfocado en el análisis de los factores del proyecto que se encuentra en ejecución o ya se ejecutó.

7.5.1. Costo presupuestado

Los gastos presupuestarios se establecen desde un punto de vista contable, para concretar la relación de gastos. En primer lugar, se consideró necesario determinar los costos de los materiales y el equipo implementado que fue utilizado para el desarrollo del proyecto, tomando como referencia visualizar la tabla 4.

Tabla 4*Análisis de costo presupuestado de los materiales y equipos*

Equipos/Materiales	Cantidad (u)	Costo (\$)	Subtotal(\$)
Rollo de alambre de cobre #16 (50m)	1	28.00	28.00
Kit de herramientas	1	250.00	250.00
Tarjeta de adquisición de datos	1	198.99	198.99
Módulo de ABS con optoacopladores (4 integrados)	5	13.39	66.95
Licencia de plataforma de Firebase (plan gratuito)	1	0.00	0.00
Licencia de SDK de Vuforia (plan gratuito)	1	0.00	0.00
Licencia de Framework Reac (plan gratuito)	1	0.00	0.00
		Total (\$)	543.94

Nota: *Se puede observar los materiales y equipos utilizados para la implementación del sistema de monitoreo.*

Después de haber seleccionado todos los recursos necesarios con respecto a todos los equipos, materiales y herramientas necesarias para implementar el proyecto y se obtuvo un valor total de \$543.94.

7.5.2. Costo por actividades

Los costos por actividades están estipulados en función de la cantidad de tareas que se requieren realizar para el montaje del sistema de adquisición, para ello se tomó como referencia el costo unitario por hora en relación con el costo por hora de un ingeniero como de un tecnólogo experimentado.

una vez designados estos valores se planteó una tabla referente al precio de cada actividad a desarrollar, dentro de las actividades más demandantes esta designada el cableado de los sensores debido a que el proceso de realización de dicha actividad requiere de la conexión física e cada sensor a la tarjeta de adquisición.

Actividad	Encargado	Personas	Tiempo de duración (horas)	Costo de mano de obra
Estudio del proceso	Ingeniero	1	16	130
Selección de variables	Ingeniero	1	16	130
Selección de herramientas	Tecnólogo	2	24	259.88
Cableado de sensores	Ingeniero y tecnólogo	2	20	216
Cableado de actuadores	Ingeniero	1	20	162.50
Conexión con la nube	Ingeniero	1	8	65
Programación del sistema DAQ	Ingeniero y tecnólogo	2	20	216.56
Conexión de DAQ con sensores y actuadores	Ingeniero	1	32	260
Configuración de protocolos de comunicación	Ingeniero	1	12	97.50
Vinculación de las aplicaciones a la nube	Ingeniero y tecnólogo	2	8	67.70
Edición de reglas de seguridad de la nube.	Ingeniero	1	12	97.50
Diseño de la interfaz	Ingeniero	1	24	195
Conexión con la nube de la aplicación	Tecnólogo	1	12	32.44
Cuestionario de satisfacción	Oficial	1	8	21.25
Charla de capacitación	Ingeniero	1	16	130.00
			Total	2114.33

7.5.3. Costo por paquete

en el análisis del costo de cada paquete que hace relación al conjunto de actividades desarrolladas para la elaboración del presente trabajo de titulación se consideraron las presentadas en la siguiente tabla:

Paquete	Actividad	Subtotal	Total
Levantamiento de información.	Estudio del proceso.	130	292.44
	Selección de variables.	130	
	Selección de herramientas.	32.44	
Instalación de Sensores y	Cableado de Sensores.	216	378
	Cableado de Actuadores.	162	
Actuadores. Instalación del Sistema DAQ	Conexión la nube.	65	541
	Programación del Sistema DAQ.	216	
	Conexiones de DAQ con sensores y actuadores.	260	
Configuración con la nube	Configuración de protocolos de comunicación.	97	262
	Vinculación de las aplicaciones a la nube.	67	
	Edición de las reglas de seguridad de la nube.	97	
Desarrolló de la App WEB	Diseño de la Interfaz	195	227
	Conexión con la nube de la Aplicación.	32	
Pruebas del Sistema	Cuestionario de satisfacción.	21.25	21.25
Capacitación de los usuarios	Charla de Capacitación.	130	130

7.5.4. Costo total del proyecto

Realizado el análisis de los costos relacionados con la obtención del material, equipos y herramientas, además de los costos de cada actividad para la realización del proyecto, basándose en el costo de la mano de obra, se realizó una suma de los análisis para obtener el valor total de la implementación del proyecto. Donde, el costo total presupuestado es CTP, el costo presupuestado inicial es CP y el costo presupuestado por actividades es CPA. A continuación, se presenta el cálculo para la obtención del costo total:

$$\begin{aligned}
 \text{CTP} &= \text{CP} + \text{CPA} \\
 \text{CTP} &= 543.94 + 2114.33 \\
 \text{CTP} &= 2658.27
 \end{aligned}$$

Luego de realizar el respectivo cálculo, se determinó que el costo total del proyecto está valorado por una cantidad de: \$2694.37.

8. Conclusiones

- El estudio de los parámetros iniciales con los que cuenta cada PLC en los módulos Sorting y Handling permitió determinar la selección de datos de sensores y actuadores que mayor actividad y funcionalidad cumplen dentro de un proceso de producción para implementar un sistema IoT de monitoreo y visualización, concluyendo que la implementación del circuito electrónico planteado permitió el envío de datos del PLC de una forma eficiente a la nube. Adicionalmente, se debe considerar que se trabajó con señales digitales de 6 sensores ópticos reflexivos y 1 sensor inductivo de reflexión, además con la presencia de 4 señales digitales con activación de actuadores neumáticos.
- La aplicación de realidad aumentada, proporcionó información claramente detallada respecto al funcionamiento de la planta, lo cual llevo a un mejor entendimiento del proceso que se realizó en dicha estación, cabe mencionar que el SDK de Vuforia permite tener una mayor portabilidad para la implementación de la aplicación en diferentes versiones de Android por lo cual resulto factible la inclusión de este motor de realidad aumentada. Finalmente, la experiencia obtenida durante el proceso de evaluación evidencio que el retraso de actualización de los valores está vinculado al desempeño del ancho de banda del internet al cual está conectado el sistema de adquisición.
- La aplicación web permitió tener una corroboración de la información actualizada en la aplicación de realidad aumentada, ofreciendo un sistema dual para la visualización del proceso de funcionamiento del sistema de monitorización implementado, también se obtuvo un desempeño similar al de la aplicación con realidad aumentada, obteniendo un retraso de 1.5 segundos entre el funcionamiento de la planta en tiempo real y el tiempo de demora en la actualización de los datos. Finalmente, el proceso de despliegue de la aplicación web mediante Firebase agilizo el proceso de la aplicación para el análisis de desempeño, por lo cual Firebase es sumamente útil al momento de realizar el testeado de aplicaciones.
- Con respecto a la evaluación de desempeño del sistema IoT se obtuvieron resultados positivos en cuanto al funcionamiento y comportamiento del sistema. Tras evaluar cada pregunta del total de la población encuestada, se concluye que existe una excelente

conexión de los módulos a internet, un tiempo de respuesta eficiente, una interpretabilidad adecuada con las etiquetas visualizadas a través de realidad aumentada, el tiempo de actualización de los datos se encuentra un rango aceptable y que la información necesaria de visualizar es un indicador de porcentaje de producción. Gracias al análisis de evaluación se pudo destacar que aunque existe un retardo de tiempo en el procesamiento de información, el comportamiento del sistema es el adecuado para la población evaluada.

9. Recomendaciones

- Para el desarrollo de proyectos relacionados con sistemas IoT se recomienda adquirir todas las señales presentes en los PLC de las demás estaciones presentes en el MPS 500, debido a que en el presente trabajo solo se tomaron las más visibles e importantes, según el operador.
- Se recomienda realizar una investigación sobre más propuestas existentes para la comunicación de sistemas IoT con componentes físicos, tomando en cuenta la aparición de nuevas tecnologías en cuanto a sistemas embebidos y tarjetas electrónicas o CPUs para la adquisición de datos digitales.
- Durante el proceso de adquisición de señales se sugiere que dentro del programa encargado de la lectura de señales del Raspberry Pi 3, se trabaje con corrutinas con la finalidad de hacer más eficiente el proceso de lectura y escritura de la base de datos, cabe mencionar que al trabajar con corrutinas permite hacer un mayor uso del procesador de nuestra placa.
- Se recomienda investigar más métodos de comunicación en el cual permita desarrollar un sistema IoT con un menor tiempo de comunicación de respuesta, permitiendo así un correcto y exhaustivo análisis para la implementación en líneas de producción industriales presentes hoy en día en la Industria 4.0.

Referencias

- Andrés, M. B. (2020). *Internet de las cosas*. Reus.
- Bejerano, P. G. (2014). El origen de la realidad aumentada.
- Borreiro, F., y Paguay, E. (2010). *Estudio para la optimización del sistema de control de la unidad de bombeo power oil 3 del campor auca central de petroproducción utilizando plc's*. Escuela Superior Politécnica del Chimborazo.
- Cirani, S., Ferrari, G., Picone, M., y Veltri, L. (2018). *Internet of things architectures, protocols and standards*. Wiley Sons, Incorporated, John.
- Daneri, P. (2009). *Plc - automatización y control industrial*. Editorial Hispano Americana HASA.
- FestoDidactic. (2021). Estación de clasificación mps d: combinación de sensores ópticos y detectores inductivos - estaciones mps - módulos de la fábrica para la enseñanza - automatización de fábricas e industria 4.0 - productos. Disponible en: <https://www.festo-didactic.com/es-es/productos/automatizacion-de-fabricas-e-industria-4.0/modulos-de-la-fabrica-para-la-ensenanza/estaciones-mps/vista-general-de-las-estaciones-del-sistema-de-produccion-modular.htm?fbid=ZXMuZXMuNTQ3LjE0LjE4LjYwNi40NzY1>.
- Furht, B. (2011). *Handbook of Augmented Reality* (1.^a ed.). New York: Springer New York, NY. Descargado 2022-06-09, de <https://link.springer.com/book/9783030678210>
- Glover, J. (2018). *Unity 2018 augmented reality projects* (P. P. Ltd., Ed.). Packt Publishing Ltd.
- Hersent, O., Boswarthick, D., y Elloumi, O. (2011). *Internet of things key applications and protocols*. Wiley Sons, Incorporated, John.
- Jara, H. D. T. (2020). *Desarrollo de una aplicación móvil basada en realidad aumentada para la caracterización y visualización del patrón de radiación en 3d de antenas predefinidas*. (Tesis de Master no publicada). Universidad Politécnica Salesiana.
- Linowes, J. (2021). *Augmented Reality with Unity AR Foundation* (August 2021 ed., Vol. 1) (n.º 1130821). Livery Place 35 Livery Street Birmingham: Packt Publishing Ltd.
- Pangilinan, E., Lukas, S., y Vasanth, M. (s.f.). *Creating Augmented and Virtual Realities*. Descargado 2022-06-09, de <https://learning.oreilly.com/library/view/creating-augmented-and/9781492044185/>
- Richter, F. (2022, febrero). *Amazon leads 180 – billioncloudmarket*. Statista. Descargado de doi: AmazonWebServices
- Rose, K. (2015). La internet de las cosas — una breve reseña. *Internet Society*.

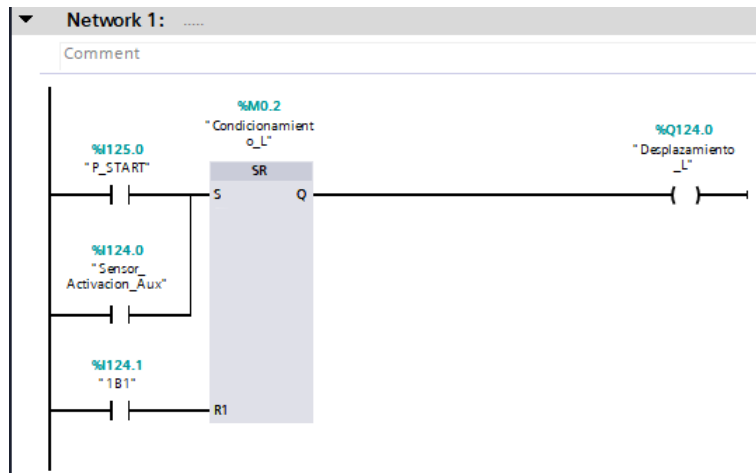
- Shanan, M. C. R. (2015). *Fundamentals of azure* (D. Musgrave, Ed.). Microsoft Press.
- SIEMENS. (2006a). Communication with simatic - system manual [Manual de software informático]. Germany, Nurnberg.
- SIEMENS. (2006b). Handling station [Manual de software informático]. Germany, Nurnberg.
- SIEMENS. (2006c). Sorting station [Manual de software informático]. Germany, Nurnberg.
- Torres, J. M. U. (2009). Configuración y puesta en marcha de una red mpi.
- Velux. (2020). Visualize your room in a whole new light with the velux app.
- Weigmann, J., y Kilian, G. (2004). *Decentralization with profibus dp/dpv1*. Wiley-VCH.
- Weis, O. (s.f.). *Qué es rs485 - guía de la comunicación rs485 [2021]*. Descargado de <https://www.eltima.com/es/article/rs485-communication-guide/#:~:text=RS485esunestándarde,nodosseconectenentresí>.
- Wittig, A., y Wittig, M. (s.f.). *Amazon web services in action*. Manning Publications.
- Yahiaoui, H. (s.f.). *Firestore cookbook: Over 70 recipes to help you create real-time web and mobile applications with firestore*. Packt Publishing.

ANEXOS

Anexo A: Programación de secuencia lógica de los PLC's mediante el software TIA PORTAL

Figura 49

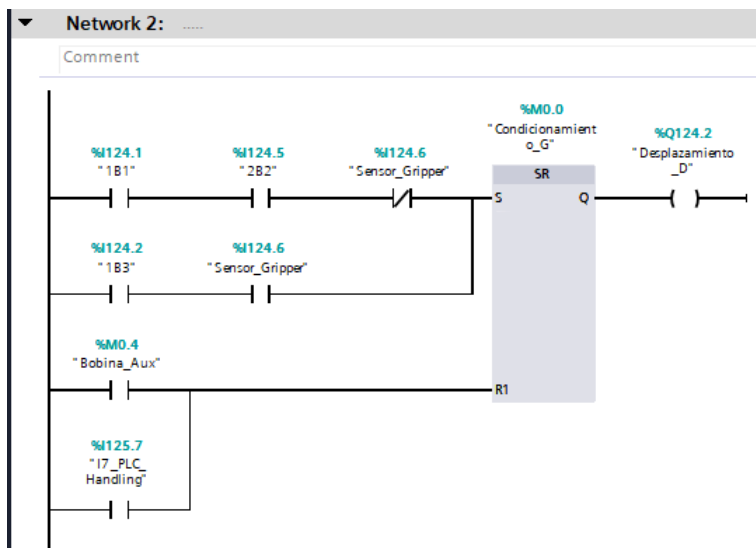
Programación de secuencia lógica del módulo Handling Network 1



Nota: Se aprecia la programación realizada para el módulo Handling.

Figura 50

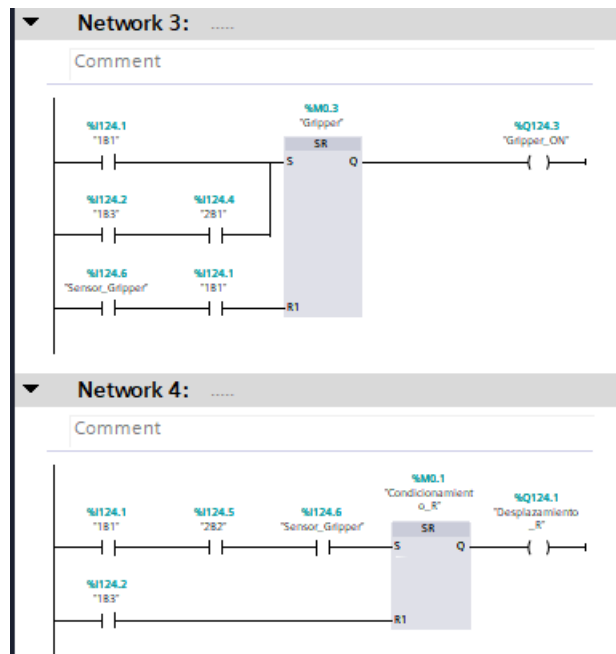
Programación de secuencia lógica del módulo Handling Network 2



Nota: Se aprecia la programación realizada para el módulo Handling.

Figura 51

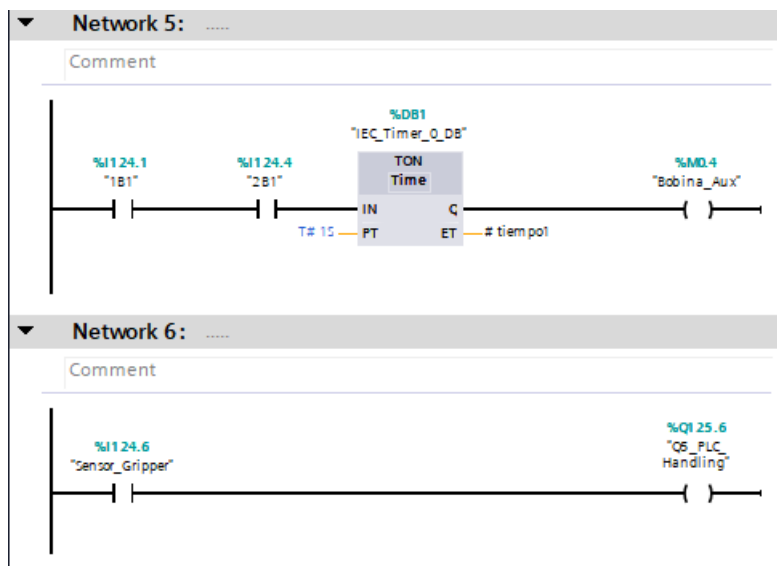
Programación de secuencia lógica del módulo Handling Network 3 y 4



Nota: Se aprecia la programación realizada para el módulo Handling.

Figura 52

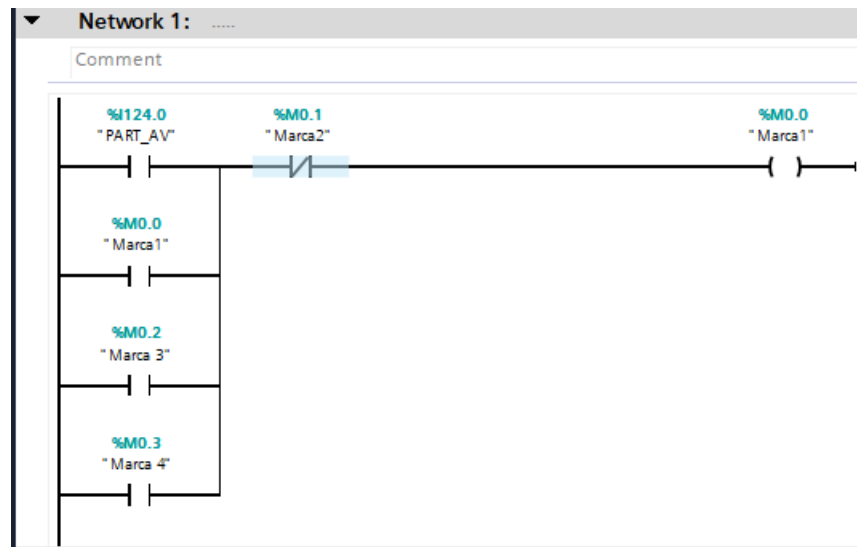
Programación de secuencia lógica del módulo Handling Network 5 y 6



Nota: Se aprecia la programación realizada para el módulo Handling.

Figura 53

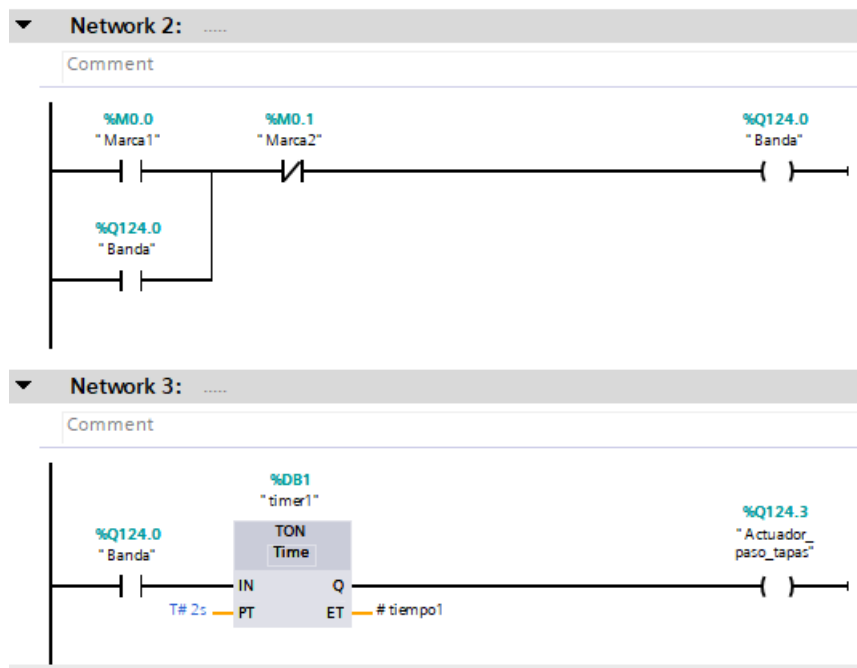
Programación de secuencia lógica del módulo Sorting Network 1



Nota: Se aprecia la programación realizada para el módulo Sorting.

Figura 54

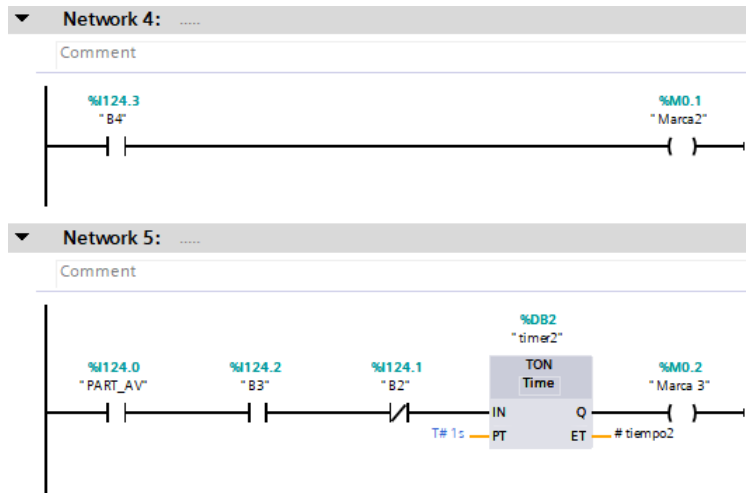
Programación de secuencia lógica del módulo Sorting Network 2 y 3



Nota: Se aprecia la programación realizada para el módulo Sorting.

Figura 55

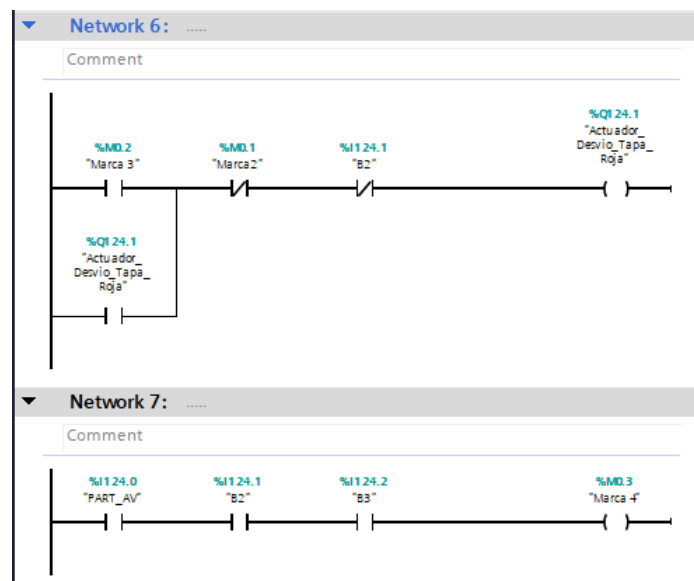
Programación de secuencia lógica del módulo Sorting Network 4 y 5



Nota: Se aprecia la programación realizada para el módulo Sorting.

Figura 56

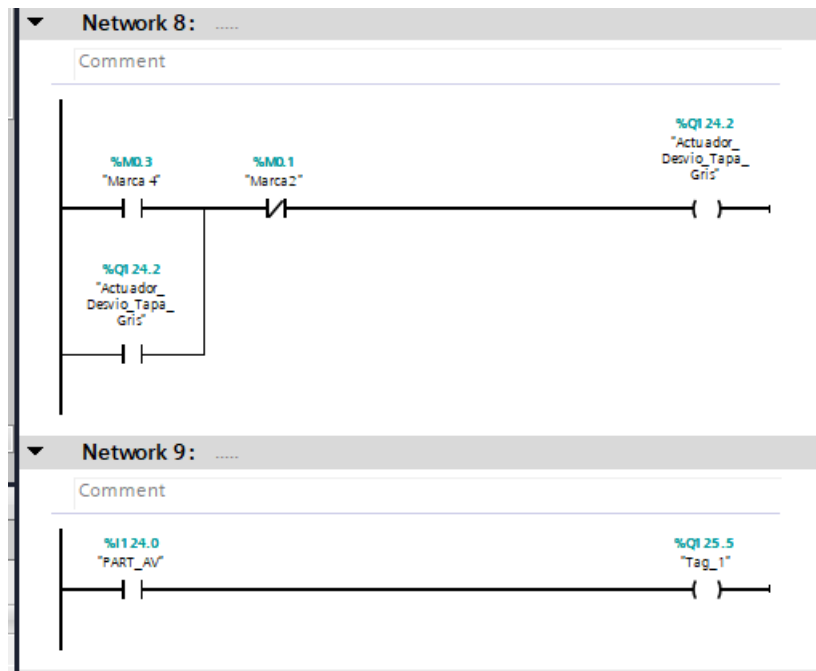
Programación de secuencia lógica del módulo Sorting Network 6 y 7



Nota: Se aprecia la programación realizada para el módulo Sorting.

Figura 57

Programación de secuencia lógica del módulo Sorting Network 8 y 9



Nota: Se aprecia la programación realizada para el módulo Sorting.