

UNIVERSIDAD PRIVADA ANTENOR ORREGO

ESCUELA DE POSGRADO



TESIS PARA OPTAR EL GRADO DE MAESTRO EN INGENIERÍA DE SISTEMAS CON MENCIÓN EN SISTEMAS DE INFORMACIÓN

Módulo de conectividad IoT – BI para monitoreo de producción de
sistema robótico agrobot upao en Trujillo, 2024

Área de Investigación:

Sistemas inteligentes

Autor:

Mendoza Chomba, Jorge Esteban

Jurado Evaluador:

Presidente: Azabache Fernández, Filiberto Melchor

Secretario: De La Cruz Rodríguez, Oscar Miguel

Vocal: Alvarado Rodríguez, Luis Enrique

Asesor:

Cerna Sánchez, Eduardo Elmer

Código Orcid: <https://orcid.org/0000-0001-5778-2259>

TRUJILLO – PERÚ

2024

Fecha de sustentación: 2024/03/18

MÓDULO DE CONECTIVIDAD IOT – BI PARA MONITOREO DE PRODUCCIÓN DE SISTEMA ROBÓTICO AGROBOT UPAO EN TRUJILLO, 2024

ORIGINALITY REPORT

1 %	1 %	1 %	0 %
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	hdl.handle.net Internet Source	1 %
----------	--	------------

Exclude quotes On
Exclude bibliography On

Exclude matches < 1%


Ms. Ing. Eduardo Cerna Sánchez
Asesor

R.D. 0073-2024-D-EPG-UPAO

DECLARACION DE ORIGINALIDAD

Yo, Eduardo Elmer Cerna Sánchez, docente del Programa de Estudio de Pregrado de la Universidad Privada Antenor Orrego, asesor de la tesis titulada “MÓDULO DE CONECTIVIDAD IOT – BI PARA MONITOREO DE PRODUCCIÓN DE SISTEMA ROBÓTICO AGROBOT UPAO EN TRUJILLO, 2024”, del autor Jorge Esteban Mendoza.

- El mencionado documento tiene un índice de puntuación de similitud del 1%. Así lo consigna el reporte de similitud emitido por el software Turnitin el día de 22 de Marzo del 2024.
- He revisado con detalle dicho reporte de la tesis “MÓDULO DE CONECTIVIDAD IOT – BI PARA MONITOREO DE PRODUCCIÓN DE SISTEMA ROBÓTICO AGROBOT UPAO EN TRUJILLO, 2024” y no se advierte indicios de plagio.
- Las citas a otros autores y sus respectivas referencias cumplen con las normas establecidas por la Universidad.

Ciudad y fecha: Trujillo 22 de marzo del 2024



MS. Ing. EDUARDO CERNA SÁNCHEZ

DNI: 18216463

ORCID: <https://orcid.org/0000-0001-5778-2259>



Br. MENDOZA CHOMBA, JORGE ESTEBAN

DNI: 74822575

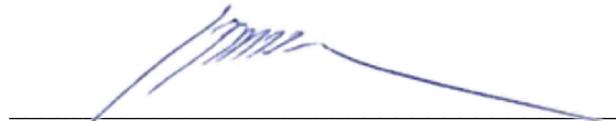
ORCID: <https://orcid.org/0000-0001-7316-9841>

ACREDITACIÓN

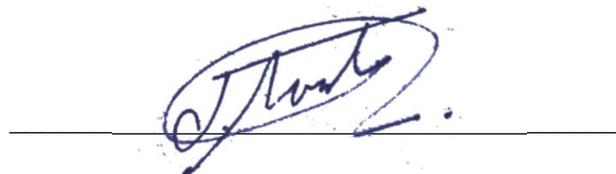
El Ms. Ing. Eduardo Elmer, CERNA SÁNCHEZ, que suscribe, asesor de la Tesis con Título “**MÓDULO DE CONECTIVIDAD IOT – BI PARA MONITOREO DE PRODUCCIÓN DE SISTEMA ROBÓTICO AGROBOT UPAO EN TRUJILLO, 2024.**”, desarrollado por el Br. Jorge Esteban, MENDOZA CHOMBA, en Ingeniería de Sistemas: Mención en sistemas de información, acredita haber realizado las observaciones y recomendaciones pertinentes, encontrándose expedita para su revisión por parte de los señores miembros del Jurado Evaluador.

Trujillo, 2024

El asesor:



Ms. Ing. Eduardo Elmer, CERNA SÁNCHEZ.



Br. Jorge Esteban, MENDOZA CHOMBA

PRESENTACIÓN

Señores Miembros del Jurado:

En cumplimiento con lo requerido por el reglamento de Grados y Títulos de la Escuela de Postgrado de la Universidad Privada Antenor Orrego, con el fin de obtener el grado de Maestro en Ingeniería de Sistemas con mención en Sistemas de Información, pongo a vuestra disposición la presente tesis con título: **MÓDULO DE CONECTIVIDAD IOT – BI PARA MONITOREO DE PRODUCCIÓN DE SISTEMA ROBÓTICO AGROBOT UPAO EN TRUJILLO, 2024**

Gracias.

Trujillo, 24 de enero de 2024

Br. Jorge Esteban MENDOZA CHOMBA

Dedicado:

*A mis padres, Nelson y Gladys, cuyo amor y sabiduría han sido mi faro para mí,
guiándome con paciencia y ternura hacia este logro.*

*A mi Tía Liliana, cuyos consejos y apoyo incondicional han sido un refugio de paz y
sabiduría en los momentos más desafiantes.*

*A mi mamá Carmen, que desde un lugar más allá del tiempo y el espacio, continúa
siendo mi inspiración y fuerza. Su legado vive en cada página de este trabajo.*

*Y a mis abuelos, cuya presencia silenciosa y constante amor han sido las raíces que
me mantienen firme y centrado en la vida y en este viaje académico.*

*Este trabajo es un reflejo de lo que todos ustedes han sembrado en mí. Con todo mi
amor y gratitud.*

Br. Jorge Esteban Mendoza Chomba.

AGRADECIMIENTO

Expreso el más profundo agradecimiento:

Al Dr. Ing. Sixto Prado, investigador del LABINM, al que expreso mi profunda gratitud por su colaboración e incondicional apoyo para el desarrollo del presente trabajo de investigación permitiéndome pertenecer al grupo de investigación, donde he logrado ampliar y mejorar mis capacidades profesionales, aportando el resultado del presente estudio al proyecto PE501079170-2022-PROCIENCIA de título “Desarrollo e Implementación de una Máquina Automatizada Multifuncional para la Siembra de Semillas de Diferentes Tamaños en bandejas multicelda y Repique de Plantines según calidad de Crecimiento en Viveros de la Región La Libertad-Perú”, que se desarrolló en el laboratorio de investigación multidisciplinaria (Labinm) de la Universidad Privada Antenor Orrego.

Br. Jorge Esteban Mendoza Chomba.



RESUMEN

“MÓDULO DE CONECTIVIDAD IOT – BI PARA MONITOREO DE PRODUCCIÓN DE SISTEMA ROBÓTICO AGROBOT UPAO EN TRUJILLO, 2024.”

Br. Jorge Esteban MENDOZA CHOMBA

Este trabajo de investigación analiza el impacto de la implementación de un módulo de comunicación IoT-BI en el sistema robótico Agrobot UPAO PE501079170-2022-PROCIENCIA, específicamente en el macroproceso de monitoreo de producción en la ciudad de Trujillo durante el año 2024. A través de la automatización de tres microprocesos críticos que lo conforman, como son la generación y difusión de reportes de producción y el informe diario de producción en pro de optimizar los tiempos de gestión de información relacionada y mejorar la eficiencia de la gestión de datos para la toma de decisiones.

Utilizando una metodología cuantitativa preexperimental, se compararon los tiempos de duración de estos procesos antes y después de la implementación del sistema, aplicando pruebas T para evaluar la significancia del impacto observado. Los resultados revelaron una reducción significativa en los tiempos de todos los microprocesos y el macroproceso global, evidenciando la eficacia del módulo IoT-BI en la optimización del monitoreo de la producción. Este hallazgo es coherente con estudios previos en el campo de la automatización agrícola, como los realizados por (Loyola Diaz, 2022), (Mendoza Vásquez, 2021), y (Arcentales Toledo & Slacedo Mena, 2023), que también han documentado mejoras significativas en la eficiencia operativa mediante la implementación de tecnologías similares. Además, se alinea con la tendencia creciente hacia la integración de sistemas automatizados en la agricultura 4.0, destacada por (Shaukat et al., 2020), (Ahmad et al., 2020), y (Muñoz Choque, 2021), quienes resaltan la importancia de los automatismos informáticos y la tecnología IoT-BI en la mejora de los procesos de gestión de información y productividad.

Las implicaciones de este estudio sugieren que la adopción de tecnologías IoT-BI puede ser una estrategia clave para la mejora continua en la agricultura moderna. Sin embargo, se reconoce la necesidad de implementar esquemas de seguridad robustos, expandir la muestra de sistemas robóticos para futuras investigaciones, y explorar la integración de inteligencia artificial para la automatización de análisis y redacción de informes, asegurando así la escalabilidad y la sostenibilidad a largo plazo de estas soluciones tecnológicas.

Palabras clave: *IoT-BI, sistema robótico agrícola, automatización, monitoreo de producción, agricultura 4.0.*

ABSTRACT

“IOT – BI CONNECTIVITY MODULE FOR PRODUCTION MONITORING OF THE AGROBOT UPAO ROBOTIC SYSTEM IN TRUJILLO, 2024.

Bs. Jorge Esteban MENDOZA CHOMBA

This research work examines the impact of implementing an IoT-BI communication module in the Agrobot UPAO PE501079170-2022-PROCIENCIA robotic system, specifically on the macro-process of production monitoring in the city of Trujillo during the year 2024. Through the automation of three critical micro-processes that comprise it, such as the generation and dissemination of production reports and the daily production report, the goal is to optimize the management times of related information and improve the efficiency of data management for decision-making.

Using a pre-experimental quantitative methodology, the duration times of these processes before and after the system implementation were compared, applying T-tests to assess the significance of the observed impact. The results revealed a significant reduction in the times of all micro-processes and the overall macro-process, demonstrating the effectiveness of the IoT-BI module in optimizing production monitoring. This finding is consistent with previous studies in the field of agricultural automation, such as those conducted by (Loyola Diaz, 2022), (Mendoza Vásquez, 2021), and (Arcentales Toledo & Slacedo Mena, 2023), which have also documented significant improvements in operational efficiency through the implementation of similar technologies. Furthermore, it aligns with the growing trend towards the integration of automated systems in agriculture 4.0, highlighted by (Shaukat et al., 2020), (Ahmad et al., 2020), and (Muñoz Choque, 2021), who emphasize the importance of computer automatons and IoT-BI technology in improving information management processes and productivity.

The implications of this study suggest that the adoption of IoT-BI technologies can be a key strategy for continuous improvement in modern agriculture. However, the need to implement robust security schemes, expand the sample of robotic systems for future research, and explore the integration of artificial intelligence for the automation of analysis and report writing is recognized, thus ensuring the scalability and long-term sustainability of these technological solutions.

Keywords: *IoT-BI, agricultural robotic system, automation, production monitoring, agriculture 4.0.*

ÍNDICE

CAPÍTULO I: INTRODUCCIÓN	16
1.1. EL PROBLEMA	16
1.1.1. Contexto donde se presenta el problema	17
1.1.2. Características problemáticas.....	17
1.1.3. Análisis de características problemáticas.....	18
1.2. ENUNCIADO DEL PROBLEMA	19
1.3. ALCANCE.....	19
1.4. VIABILIDAD	19
1.5. JUSTIFICACIÓN	20
1.5.1. Conveniencia.....	20
1.5.2. Relevancia Social.....	20
1.5.3. Justificación Teórica	20
1.5.4. Justificación Práctica	21
1.6. HIPÓTESIS	21
1.6.1. HIPÓTESIS ESPECÍFICAS.....	21
1.6.2. VARIABLES	21
1.6.3. MATRIZ DE OPERACIONALIZACIÓN DE VARIABLES	22
1.7. OBJETIVOS.....	23
1.7.1. OBJETIVO GENERAL.....	23
1.7.2. OBJETIVOS ESPECÍFICOS	23
CAPITULO II MARCO TEÓRICO.....	25
2.1. ANTECEDENTES.....	25
2.2. BASES TEÓRICAS	28
2.2.1. Bussiness Intelligence (BI)	28
2.2.2. Cloud Computing.....	29
2.2.3. Aplication Programing Interface (API)	29

2.2.4.	Base de datos.....	30
2.2.5.	Cliente Web	30
2.2.6.	Sistema Robótico Agrícola	30
CAPÍTULO III: METODOLOGÍA.....		33
3.1.	TIPO DE INVESTIGACIÓN	33
3.2.	DISEÑO	33
3.2.1.	Etapa Pre-test	33
3.2.2.	Implementación.....	34
3.2.3.	Etapa Post-test	34
3.3.	POBLACIÓN	35
3.4.	MUESTRA.....	35
3.5.	MÉTODOS DE RECOLECCIÓN DE DATOS.....	35
3.5.1.	INSTRUMENTOS.....	35
3.6.	ASPECTOS ÉTICOS	36
CAPÍTULO IV: RESULTADOS.....		38
4.1.	Identificación de indicadores dentro de realidad problemática.....	38
4.2.	Aplicación de instrumento en etapa pre-test.....	42
4.2.1.	Medición de tiempos de generación de reporte de operación.....	42
4.2.2.	Medición de tiempos de difusión de reporte de operación	44
4.2.3.	Medición de tiempos de generación-difusión de informe diario de operación	45
4.2.4.	Resumen de tiempos totales.....	47
4.3.	Desarrollo del módulo de comunicación IoT-BI	48
4.3.1.	Elaboración de lista de requerimientos	48
4.3.2.	Implementación de requerimientos en bloques funcionales	49
4.3.3.	Pruebas de operatividad	64
4.3.4.	Corrección de errores y despliegue de módulo funcional.....	64
4.4.	Aplicación de instrumento en etapa post-test.....	66

4.4.1.	Medición de tiempos de generación de reporte de operación.....	66
4.4.2.	Medición de tiempos de difusión de reporte de operación	68
4.4.3.	Medición de tiempos de generación-difusión de informe diario de operación	69
4.4.4.	Resumen de tiempos totales.....	72
4.5.	Procesamiento y análisis de datos.....	74
4.5.1.	Procesamiento de datos.....	74
4.5.2.	Análisis de normalidad de datos	75
4.5.3.	Prueba de hipótesis	77
4.5.3.1.	Objetivo General.....	78
4.5.3.1.1.	Comprobación de Hipótesis.....	78
4.5.3.2.	Objetivo Específico 1.....	79
4.5.3.2.1.	Comprobación de Hipótesis Específica 1	79
4.5.3.3.	Objetivo Específico 2.....	80
4.5.3.3.1.	Comprobación de Hipótesis Específica 2	80
4.5.3.4.	Objetivo Específico 3.....	82
4.5.3.4.1.	Comprobación de Hipótesis Específica 3	82
	CAPÍTULO V: DISCUSIÓN	85
5.1.	DISCUSIÓN	85
	CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES.....	88
6.1.	CONCLUSIONES.....	88
6.2.	RECOMENDACIONES	90
7.	REFERENCIAS BIBLIOGRÁFICAS.....	92
8.	ANEXOS	94

ÍNDICE DE FIGURAS

Figura 1 <i>Instrumento guía de observación de tiempos de elaboración de reportes.</i>	36
Figura 2 <i>Vista isométrica frontal de modelo 3D de estructura de PE501079170-2022-PROCIENCIA.</i>	38
Figura 3 <i>Vista lateral de modelo 3D de estructura de PE501079170-2022-PROCIENCIA.</i>	39
Figura 4 <i>Vista isométrica posterior de modelo 3D de estructura de PE501079170-2022-PROCIENCIA.</i>	39
Figura 5 <i>Estructura física de PE501079170-2022-PROCIENCIA.</i>	40
Figura 6 <i>Terminal de Linux con el script principal en ejecución.</i>	41
Figura 7 <i>Resultados del procesamiento mostrados en la terminal.</i>	41
Figura 8 <i>Diagrama de bloques de script generador de reporte.</i>	49
Figura 9 <i>Implementación de script generador de reporte en Python.</i>	50
Figura 10 <i>Parámetros de la cuenta de servicio para acceso a archivos compartidos.</i>	51
Figura 11 <i>Diagrama de flujo de script lector de direcciones de correo.</i>	52
Figura 12 <i>Implementación de script lector de direcciones de correo en Python.</i>	52
Figura 13 <i>Parámetros de dirección de correo del script gestor de correos.</i>	53
Figura 14 <i>Diagrama de flujo de script gestor de correos.</i>	53
Figura 15 <i>Implementación de script gestor de correos en Python líneas 7-27.</i>	54
Figura 16 <i>Implementación de script gestor de correos en Python líneas 24 a 49.</i>	55
Figura 17 <i>Base datos implementada en Firebase.</i>	56
Figura 18 <i>Diagrama de flujo de script de gestión de base de datos.</i>	56
Figura 19 <i>Implementación de script de gestión de base de datos en Python.</i>	57
Figura 20. <i>Enlace de cliente Web con base de datos.</i>	58
Figura 21 <i>Diagrama de flujo de script generador de informe diario de producción.</i>	59
Figura 22 <i>Implementación de script generador de informe diario de producción en Python líneas 6 a 38.</i>	60
Figura 23 <i>Implementación de script generador de informe diario de producción en Python líneas 39 a 56.</i>	61
Figura 24 <i>Cabecera de informe diario de producción generado por script.</i>	61
Figura 25 <i>Gráfica recuperada de cliente web insertada en informe.</i>	62
Figura 26 <i>Implementación de script generador de informe diario de producción en Python líneas 54 a 75.</i>	63

Figura 27 <i>Implementación de script de extracción de datos de cliente web en Python líneas 3 a 41.....</i>	65
Figura 28 <i>Implementación de script de extracción de datos de cliente web en Python líneas 41 a 55.....</i>	66
Figura 29 <i>Gráfica de barras de tiempos por subproceso pre y post-test.....</i>	74

ÍNDICE DE TABLAS

Tabla 1	<i>Matriz de operacionalización de variables</i>	22
Tabla 2	<i>Tiempos por subproceso de la generación de reporte de producción pre-test.</i>	43
Tabla 3	<i>Tiempos por subproceso de la difusión de reporte de producción pre-test.</i>	44
Tabla 4	<i>Tiempos por subproceso de la generación- difusión de informe diario de producción pre-test.</i>	46
Tabla 5	<i>Resumen de tiempos de subprocesos de monitoreo de producción de PE501079170-2022-PROCIENCIA pre-test.</i>	47
Tabla 6	<i>Correspondencia de tareas con capacidades requeridas del módulo IoT-BI.</i>	48
Tabla 7	<i>Correspondencia de tareas y scripts de automatización en generación de reporte de producción.</i>	67
Tabla 8	<i>Tiempos por subproceso de la generación de reporte de producción post-test.</i>	67
Tabla 9	<i>Correspondencia de tareas y scripts de automatización en difusión de reporte de producción.</i>	68
Tabla 10	<i>Tiempos por subproceso de la difusión de reporte de producción post-test.</i>	69
Tabla 11	<i>Correspondencia de tareas y scripts de automatización en generación-difusión de informe diario de producción.</i>	70
Tabla 12	<i>Tiempos por subproceso de la generación- difusión de informe diario de producción post-test.</i>	71
Tabla 13	<i>Resumen de tiempos de subprocesos de monitoreo de producción de PE501079170-2022-PROCIENCIA Post-test.</i>	73
Tabla 14	<i>Promedio de tiempos por subproceso pre y post-test.</i>	74
Tabla 15	<i>Resultados de prueba de normalidad Shapiro-Wilk a valores de tiempo de los indicadores pre y post-test.</i>	76
Tabla 16	<i>Resultado de prueba T de los valores de tiempo total del proceso de monitoreo de PE501079170-2022-PROCIENCIA</i>	78
Tabla 17	<i>Resultado de prueba T de los valores de tiempo del indicador "Generación de reporte de operación" PE501079170-2022-PROCIENCIA</i>	79
Tabla 18	<i>Resultado de prueba T aplicada a los valores de tiempo del indicador "Difusión de reporte de operación" PE501079170-2022-PROCIENCIA</i>	81
Tabla 19	<i>Resultado de prueba T de los valores de tiempo del indicador "Generación-difusión del informe diario de operación" PE501079170-2022-PROCIENCIA</i>	82

CAPÍTULO I

INTRODUCCIÓN

CAPÍTULO I: INTRODUCCIÓN

1.1. EL PROBLEMA

Alrededor del año 2011 con el surgimiento del paradigma Industria 4.0 y, en concordancia con (Basco et al., 2018), su posterior consolidación en 2013, dio inicio a un proceso de evolución en los flujos de información que conforman los procesos industriales, generando un estrechamiento del vínculo entre estos y la inteligencia de negocio; la cual previamente había dependido, de manera exclusiva, de indicadores dependientes en su totalidad del factor humano; siendo así que procesos como el levantamiento de data a través de instrumentos, registro, transmisión, difusión y procesamiento de ésta eran manejados por completo por humanos y muchas veces de manera manual; lo cual, daría lugar, inevitablemente, a errores o cuellos de botella generados por las actividades adicionales que debe realizar cada empleado y a su vez, estas se podrían afectar de factores subjetivos como la motivación, de acuerdo con (Matamoros Zapata, 2010). Los problemas mencionados encontrarían solución definitiva a través de la implementación del paradigma Industria 4.0 dentro de los procesos, ya que esto permite generar los indicadores de forma automática e inmediata valiéndose de lecturas de sensores enviadas en tiempo real a un servicio en la nube que permita visualizar de manera amigable el estado de los procesos para optimizar la inteligencia de negocio.

En este orden de ideas, es coherente inferir que uno de los procesos sobre los que más se han implementado sistemas de industria 4.0, ha sido la línea de producción, que de acuerdo con (De La Cruz Mariños, 2022), específicamente la producción agrícola conforma uno de los ejes fundamentales de la industria; dando lugar a la generación de indicadores que llevarán a la toma de decisiones con respecto de acciones tan importantes como inversión o expansión, formando en general un reflejo del estado de la compañía.

Teniendo en cuenta lo mencionado anteriormente y orientándolo a la realidad del área geográfica comprendida por este estudio (Trujillo, La Libertad), la cual se encuentra en medio de un surgimiento de proyectos de robótica aplicada a las necesidades del sector agropecuario local como son los proyectos de (De La Cruz Mariños, 2022) y (García Honores & Moncada Calmet, 2022); innovaciones que en años anteriores eran exclusivas de la gran empresa que pudiera permitirse destinar recursos a la

investigación de estos temas, ahora están al alcance de productores locales gracias al esfuerzo e inversión de entidades educativas privadas.

Con lo cual, si se tiene en cuenta lo sostenido por (García Honores & Moncada Calmet, 2022), la implementación de un sistema robótico tiene un aporte significativo en la producción agropecuaria de La Libertad; y como todo proceso en crecimiento es susceptible de implementar paradigmas de Industria 4.0; es por esto que el presente estudio busca analizar el impacto de implementación de un módulo IoT – BI sobre el sistema robótico Agrobot, desarrollado en la Universidad Privada Antenor Orrego, con respecto del proceso de monitoreo de producción.

1.1.1. Contexto donde se presenta el problema

El problema objeto de este estudio está sujeto al monitoreo de la producción del sistema robótico “Maquina Automatizada Multifuncional para la Siembra de Semillas de Diferentes Tamaños en bandejas multicelda y Repique de Plantines según calidad de Crecimiento en Viveros de la Región La Libertad-Perú” proyecto PE501079170-2022-PROCIENCIA con seudónimo Agrobot desarrollado en el LABINM de la Universidad Privada Antenor Orrego.

1.1.2. Características problemáticas

La realidad problemática estudiada presenta las siguientes características:

- ✓ Dado el algoritmo que gobierna el funcionamiento del sistema robótico de producción agrícola Agrobot, requiere a un operador para que constantemente extraiga la data generada en la terminal para generar el reporte de producción.
- ✓ Dado el algoritmo que gobierna el funcionamiento del sistema robótico de producción agrícola Agrobot, requiere a un operador para que difunda el reporte de producción a los interesados a través de correo electrónico y/o servicios de mensajería.
- ✓ Dado el algoritmo que gobierna el funcionamiento del sistema robótico de producción agrícola Agrobot, requiere a un operador que genere el informe diario de producción y lo remita al área correspondiente.

1.1.3. Análisis de características problemáticas

- ✓ En la etapa pre-test del presente estudio, el algoritmo que gobierna el funcionamiento del sistema robótico de producción agrícola Agrobot está codificado en Python y compuesto por funciones de envío y recepción de datos a través de Modbus, así como también, visión artificial. Este sistema interactuaba con el usuario a través de mensajes en la terminal, por lo cual, para lograr obtener los datos, era necesario contar con un dispositivo de video, periféricos como teclado, ratón, etc. Y un operador capacitado que se encargue de extraer la data y darle formato para que sea legible; esta metodología de trabajo al ser tan común en la última década, ha sido ampliamente estudiada y modificada en pro de optimizarse a través de la implementación de sistemas que permitan anular la necesidad de periféricos y un operador, reemplazándolos con sistemas capaces de extraer, dar formato y mostrar la data en dispositivos en red de manera autónoma y dado que el sistema Agrobot no es solo un sistema informático alojado en un ordenador, sino más bien cuenta con estructura física especializada, su comunicación a través de la red cae dentro del paradigma IoT.
- ✓ En la etapa pre-test del presente estudio, dada la naturaleza de la interacción con el usuario que presentaba el algoritmo de Agrobot y el procedimiento manual necesario para extraer la data generada por el sistema, se requirió de forma adicional que el operador se encargue del proceso de difusión del reporte de producción; este reporte constaba únicamente de las magnitudes obtenidas de las hojas de los plantines analizados por el sistema y el resultado de este análisis en escala del 1 al 3 en donde 1 es calidad buena, 2 es calidad media y 3, calidad baja. El reporte debía ser difundido al líder del equipo de trabajo, investigadores que usan la data y demás interesados. Con lo cual, implicaba no solo que debe enviarse el archivo manualmente, sino también que la difusión de esta data depende por completo del operador y la disponibilidad de éste. El proceso de difusión está basado en el envío del reporte a personas específicas a través de correo electrónico, con lo cual, es viable optimizar este proceso a través de la implementación de un módulo de comunicaciones IoT

- ✓ En la etapa pre-test del presente estudio, la manera en la que se maneja la generación del reporte y la difusión de éste, es la misma en la que se realiza la generación y difusión del informe diario de producción; este informe, además de incluir la data cruda, incluye un resumen de los totales por bandeja y se considera la generación y difusión de este documento como un mismo proceso, dada su naturaleza, ya que este existe solo y únicamente para retroalimentar la lógica de negocio y proporcionar datos suficientes para la toma de decisiones, su generación y difusión se han considerado un único proceso que al tratarse de indicadores que aportan al proceso de toma de decisiones caen en el paradigma de Business Intelligence (BI).

1.2. ENUNCIADO DEL PROBLEMA

¿Cómo influye la implementación de un módulo de conectividad IoT-BI en el monitoreo de la producción del sistema robótico Agrobot UPAO en la ciudad de Trujillo - 2024?

1.3. ALCANCE

El presente trabajo de investigación se encuentra delimitado por el acceso a sistemas robóticos agrícolas de selección de semillas en el Laboratorio de Investigación Multidisciplinaria de la Universidad Privada Antenor Orrego dado que se requiere integrar scripts en el entorno de trabajo del algoritmo principal, acción que no podría realizarse en un equipo en operación dentro de planta dado que implicaría pérdida de horas/máquina y considerar adicionalmente que este tipo de sistemas robóticos son escasos aun a nivel nacional.

1.4. VIABILIDAD

El presente trabajo de investigación es viable dada la disponibilidad del sistema robótico “Maquina Automatizada Multifuncional para la Siembra de Semillas de Diferentes Tamaños en bandejas multicelda y Repique de Plantines según calidad de Crecimiento en Viveros de la Región La Libertad-Perú” con código de proyecto PE501079170-2022-PROCIENCIA en el LABINM de la Universidad Privada Antenor Orrego, que permite que se desplieguen scripts funcionales en el entorno de producción del algoritmo principal, así como también es viable en tiempo ya que este proyecto ha sido desarrollado en el periodo de pruebas de implementación del sistema.

1.5. JUSTIFICACIÓN

La presente investigación se justifica en la realidad actual considerando la tendencia de implementar tecnologías de industria 4.0 en los procesos de producción con la finalidad de optimizar su costo/beneficio, ya que de acuerdo con (Ahmad et al., 2020), existe una marcada tendencia positiva en los indicadores de productividad en los procesos que implementan este paradigma; a su vez y en concordancia con (Arcentales Toledo & Slacedo Mena, 2023), la integración de inteligencia de negocio en procesos de producción, está estrechamente vinculada con el beneficio monetario producido, con lo cual, la implementación de un módulo de comunicación IoT-BI, permitirá remover el error humano del proceso de monitoreo, logrando de esta manera, generar data más confiable y a su vez, confirmar si las tendencias encontradas en estudios de implementaciones de estos sistemas en procesos de producción tradicional, se repiten en un proceso de producción que implementa sistemas robóticos.

1.5.1. Conveniencia

Desde la perspectiva del LABINM (Laboratorio de Investigación Multidisciplinaria), donde se ha desarrollado Agrobot, es conveniente evaluar la influencia de la implementación de un módulo de comunicación IoT-BI para monitoreo de la producción del sistema robótico, ya que esto les permite saber si éste se alinea con las tendencias conocidas de los procesos productivos tradicionales en lo que a integración de industria 4.0 refiere.

Desde la perspectiva de los usuarios, es conveniente conocer el impacto de la implementación de un módulo de comunicación IoT-BI para monitoreo de la producción del sistema robótico ya que, al ser un módulo independiente, permite hacer una evaluación de costo/beneficio, permitiendo a potenciales usuarios tener mayor libertad de escoger las capacidades que requieren en un sistema robótico de selección y repique de plantines de alcachofa.

1.5.2. Relevancia Social

El presente estudio alcanza relevancia social dado el espacio geográfico en el que se desarrolla, en el cual se destacan las actividades agrícolas como uno de los principales mecanismos de producción de la región.

1.5.3. Justificación Teórica

El presente estudio se justifica en la teoría dado su grado de especificidad, ya que toma como unidad de análisis un sistema robótico de selección y repique de plantines; dado que este tipo de sistemas son relativamente recientes, el conocer si se alinean o no a las tendencias de los sistemas tradicionales en lo que a integración de paradigmas de industria 4.0 refiere, es de gran relevancia.

1.5.4. Justificación Práctica

El presente estudio se justifica en la práctica dadas las tecnologías usadas para el desarrollo del módulo de comunicación IoT-BI, objeto de este estudio, ya que permitirán que el sistema sea modular en funcionalidad y además, al tratarse de tecnologías de código abierto, puede recibir aportes de desarrolladores externos o comunidades interesadas.

1.6. HIPÓTESIS

La implementación de un módulo de comunicación IoT-BI influye significativamente en el monitoreo de la producción del sistema robótico Agrobot UPAO en la ciudad de Trujillo - 2024.

1.6.1. HIPÓTESIS ESPECÍFICAS

- ✓ La implementación de un módulo de comunicación IoT-BI con base datos NoSQL en la nube influye significativamente en el monitoreo de la producción del sistema robótico Agrobot UPAO en la ciudad de Trujillo – 2024.
- ✓ La implementación de un módulo de comunicación IoT-BI con comunicación a una interfaz de usuario web frontend con gráficas BI influye significativamente en el monitoreo de la producción del sistema robótico Agrobot UPAO en la ciudad de Trujillo – 2024.
- ✓ La implementación de un módulo de comunicación IoT-BI que puede generar y enviar por correo los informes diarios de producción influye significativamente en el monitoreo de la producción del sistema robótico Agrobot UPAO en la ciudad de Trujillo – 2024.

1.6.2. VARIABLES

Variable I: Conectividad IoT-BI del sistema robótico Agrobot UPAO.

Variable II: Monitoreo de producción en sistema robótico Agrobot UPAO.

1.6.3. MATRIZ DE OPERACIONALIZACIÓN DE VARIABLES

Tabla 1

Matriz de operacionalización de variables

Variables	Definición Conceptual	Dimensiones	Indicadores	Tipo	Técnica	Instrumento
Variable I: <i>-Conectividad IoT-BI del sistema robótico Agrobot UPAO</i>	Sistema informático alojado en la nube accesible a través de navegador desde cualquier dispositivo orientado a la gestión de data de procesos con la finalidad de generar indicadores de inteligencia de negocios	Comunicación con base de datos NoSQL almacenada en la nube	Horas/Máquina dedicadas a la generación del reporte de producción	Cuantitativo	Observación	Guía de observación
		Comunicación interfaz de usuario Frontend BI	Horas/Máquina dedicadas a la difusión del reporte de producción	Cuantitativo	Observación	Guía de observación
		Generación y difusión de informe diario de producción	Horas/Máquina dedicadas a la generación y difusión de informe diario de producción	Cuantitativo	Observación	Guía de observación
Variable II: <i>- Monitoreo de producción en sistema robótico Agrobot UPAO.</i>	Proceso mediante el cual se genera y procesa información con de la producción de bandejas de plantines categorizadas con la finalidad de conocer las cantidades y características de estas para tener control y tomar decisiones sobre esta área y las áreas relacionadas	Tiempo necesario para la generación del reporte de producción	Horas/hombre dedicadas a la generación del reporte de producción	Cuantitativo	Observación	Guía de observación
		Tiempo necesario para la difusión del reporte de producción	Horas/hombre dedicadas a la difusión del reporte de producción	Cuantitativo	Observación	Guía de observación
		Tiempo necesario para generación y difusión de informe diario de producción	Horas/hombre dedicadas a la generación y difusión de informe diario de producción	Cuantitativo	Observación	Guía de observación

1.7. OBJETIVOS

1.7.1. OBJETIVO GENERAL

Determinar cómo influye la implementación de un módulo de conectividad IoT-BI en el monitoreo de la producción del sistema robótico Agrobot UPAO en la ciudad de Trujillo – 2024.

1.7.2. OBJETIVOS ESPECÍFICOS

- ✓ Determinar cómo influye la implementación de un módulo de comunicación IoT-BI con base de datos NoSQL en la nube en el monitoreo de la producción del sistema robótico Agrobot UPAO en la ciudad de Trujillo – 2024.
- ✓ Determinar cómo influye la implementación de un módulo de comunicación IoT-BI con comunicación a una interfaz de usuario web frontend con gráficas BI, en el monitoreo de la producción del sistema robótico Agrobot UPAO en la ciudad de Trujillo – 2024.
- ✓ Determinar cómo influye la implementación de un módulo de comunicación IoT-BI que puede generar y enviar por correo los informes diarios de producción, en el monitoreo de la producción del sistema robótico Agrobot UPAO en la ciudad de Trujillo – 2024.

CAPÍTULO II

MARCO TEÓRICO

CAPITULO II MARCO TEÓRICO

2.1. ANTECEDENTES.

(Loyola Diaz, 2022) en su tesis para obtener el grado de Maestro “Solución Business Intelligence para Mejorar la Toma de Decisiones del Área de Certificación de Cultivos Orgánicos de la Cooperativa Agraria CEPROVASC – 2022” analiza cuantitativamente y de manera preexperimental, la influencia de la implementación de una solución BI en una cooperativa agraria, específicamente la cooperativa CEPROVASC confirmando que a través de esta es posible reducir los tiempos de gestión de información hasta en un 99.25%, concluyendo principalmente que:

- ✓ La integración de una solución de inteligencia de negocios a los procesos del área de certificación de cultivos orgánicos de CEPROVASC, mejoró la toma de decisiones y redujo el tiempo promedio requerido para la preparación de informes, gestión de información y análisis de ésta; así mismo, aumentó la eficacia y la satisfacción del personal.
- ✓ A través del uso de una solución BI, se logró reducir el tiempo de generación de reportes llegando a obtener tiempos entre los 6 y 30 segundos, representando una mejora de hasta el 99.24% con respecto a la etapa previa a la implementación de la solución.
- ✓ Tras la implementación de la solución BI, se obtuvo una mejora del 96.8% en tiempo total del proceso, comprendiendo este, la generación de reportes, gestión de información y generación de informes.

(Mendoza Vásquez, 2021) en su tesis para acceder al grado de maestro en ingeniería de sistemas “Implementación de herramientas Python en el proceso de producción de cultivos agrícolas del fundo San Juan de Buenavista” pone en funcionamiento una aplicación basada en modelos predictivos usando Python con la finalidad de generar indicadores de inteligencia de negocio reemplazando un operador en la tarea de gestión de data de este proceso y generación de informes predictivos, logrando de esta manera generar informes con mayor precisión y de manera inmediata, concluyendo principalmente que:

- ✓ A través de la implementación de la aplicación de modelos predictivos para la generación de indicadores BI, se logró centralizar los datos correspondientes al proceso, ya que se usó el paradigma ETL.
- ✓ Es posible la implementación de una herramienta generadora de indicadores BI a través de Python
- ✓ La implementación de esta herramienta permitió superar el límite humano de 80% de precisión en la generación de un informe predictivo basado en la data obtenida en el proceso.

(Arcentales Toledo & Slacedo Mena, 2023) En su tesis “Implementación De Un Aplicativo Web Para El Control De Producción Y La Gestión De Ventas De Plantas E Injertos De Cacao, Aplicando Inteligencia De Negocios (Bi)” implementan una solución de inteligencia de negocio al proceso de producción, vinculado a su vez, al proceso de venta, generando en consecuencia data usable para otros procesos de la empresa, como lo son Marketing, inventarios, facturación o seguridad; concluyendo principalmente que:

- ✓ Es posible implementar una aplicación de inteligencia de negocio sobre un proceso productivo de cacao, usando Python y PostgreSQL para controlar la producción y a la vez generar data usable para el proceso de ventas.
- ✓ A través de fichas de observación, es posible retroalimentar el proceso de implementación de un aplicativo BI para conocer su influencia en los procesos.
- ✓ Un aplicativo Web-BI implementado sobre el proceso de producción permite alcanzar las expectativas y requerimientos necesarios para optimizar la lógica de negocio.

(Shaukat et al., 2020) en su investigación “The Impact of Artificial intelligence and Robotics on the Future Employment Opportunities”, analizan la actual tendencia de incluir robots en los procesos de producción y otros que hasta hace poco se habían mantenido al margen, como pueden ser salud, transporte o gestión del suministro y cómo es que se da su interacción con la inteligencia de negocios influyendo, de esta manera, en los empleos dentro de los campos laborales que implementan estos sistemas, encontrando que si bien, la implementación de robótica en los diferentes campos lleva

ya algunos años evolucionando, su integración con la inteligencia de negocios generaría un nuevo paradigma que impactaría de forma significativa en los empleos, ya que un sistema robótico capaz de generar y recibir información es en la práctica, un elemento autogestionado, convirtiéndose de esta manera en un sistema robótico autónomo, permitiéndole ejecutar tareas que hasta el momento eran exclusivas de operadores experimentados, concluyendo principalmente que:

- ✓ Actualmente los sistemas robóticos autónomos son capaces de desarrollarse en entornos abiertos de maneras en las que no les sería posible a sistemas robóticos tradicionales que dependen de un operador o diseñador humano.

(Ahmad et al., 2020), en su artículo “Towards Sustainable Textile and Apparel Industry: Exploring the Role of Business Intelligence Systems in the Era of Industry 4.0”, enfocan su estudio en la integración del paradigma Business Intelligence en los procesos de industria 4.0 y su influencia sobre estos, específicamente en la industria de textiles y vestimenta, analizando datos de 12 de las empresas más grandes a nivel global, encontrando que la integración de BI a sistemas de industria 4.0 genera una optimización de los procesos de producción y además, que las empresas de venta y distribución de vestimenta presentaron aun mayor optimización ya que ya contaban con una implementación avanzada de BI cuando adoptaron los paradigmas de industria 4.0, concluyendo principalmente que:

- ✓ Las mejoras más significativas en las industrias textiles y de vestimenta se han generado cuando se han combinado la industria 4.0, el BI, el IoT y otras tecnologías recientes como IA, machine learning, impresión 3D, etc.
- ✓ Las mejoras en la cadena de producción resultado de la sinergia de Industria 4.0, BI e IoT, generan indicadores útiles para procesos circundantes como ventas, inventarios, gestión de personal o marketing.
- ✓ Las mejoras más significativas se observaron en empresas que usaron esta sinergia de tecnologías y que, además, se encontraban en entornos en los que interactuaban con más empresas que usaban la misma tecnología en países en donde el cliente también estaba adecuado a usarla, mientras que en países en los que el cliente presentaba resistencia a usar la tecnología, se observó una mejora menos sustancial.

(Muñoz Choque, 2021) en su artículo “Estudio de Tiempos y su Relación Con La Productividad” analiza la interdependencia entre la operatividad de procesos y los tiempos de operación en una empresa cementera, comparando el desempeño de los operarios con la eficiencia de las máquinas a través de la observación basada en cronometraje, encontrando que la reducción de tiempos tiene un impacto significativo en la productividad a diferencia de las condiciones de trabajo; concluyendo principalmente que:

- ✓ Existe una relación inversa entre los tiempos de operación y la productividad y descartando la influencia de las condiciones de trabajo.

2.2. BASES TEÓRICAS

2.2.1. Business Intelligence (BI)

Para definir el paradigma “Inteligencia de Negocio” o BI, por sus siglas en inglés; es necesario hacerlo en el contexto del campo en el que se esté aplicando. En este orden de ideas se puede definir mediante sus aspectos principales como son la obtención, acopio, procesamiento y entrega de información disponible y relevante tanto de forma interna como externa; de la misma manera se puede enunciar como su función principal el aporte de indicadores al proceso de toma de decisiones estratégicas en un proyecto o corporación (Boonsiritomachai et al., 2016).

Teniendo en cuenta lo mencionado anteriormente, en el contexto de la interacción de sistemas con la nube, BI hace referencia al trabajo de gestionar y analizar la data almacenada en la nube de uno o más procesos computacionales con el fin de obtener indicadores y organizarlos en pro de optimizar la toma de decisiones (Muntean, 2015).

De acuerdo con (English & Hoffmann, 2018) aunque se desarrolle dentro de sistemas informáticos complejos en la nube, la finalidad del BI continúa orientada a su objetivo tradicional, que es facilitar al proyecto o empresa el análisis de la data que se genera en sus procesos, ahora bien, al estar desplegada en sistemas en la nube, posee una característica significativamente positiva, que es el acceso a información en tiempo real y potencia de cómputo suficiente para poder contrastar la nueva data con el historial de manera inmediata y generar

indicadores predictivos que pueden ser compartidos de forma instantánea y así retroalimentar el proceso de forma eficiente.

Adicionando las ideas de (Barón Ramírez et al., 2021) a las previamente mencionadas, es necesario tomar en cuenta a la inteligencia de negocio como un macroproceso en sinergia con los procesos tradicionales por lo que la información obtenida de procesos de logística, ventas, inventario, producción, marketing, etc. conformará el input ingresado al sistema en la nube y dependiendo del análisis que requiera, la capacidad de difundirlo y el nivel de seguridad necesario, determinará el costo que este pueda representar para la organización.

2.2.2. Cloud Computing

Según el NIST, Instituto Nacional de Estándares y Tecnología, se puede definir Cloud Computing como un modelo que proporciona, bajo demanda, recursos en la red a través de la habilitación de poder computacional configurable y compartido que puede ser desplegado y provisto de forma rápida con una gestión, esfuerzo o gestión con el proveedor mínimos; de igual forma; es necesario definir Cloud Computing como un conjunto extenso de recursos virtuales accesibles y dinámicamente reconfigurables con la finalidad de soportar una carga de datos y/o solicitudes que varían en magnitud, permitiendo de esta manera, la optimización de recursos. (Singh Gurjar & Singh Rathore, 2013).

En este orden de ideas y de acuerdo con el centro de formación de Amazon Web Services, se puede definir Cloud Computing como un término que abarca la totalidad del proceso de obtener recursos computacionales a través de internet para obtener potencia de trabajo y recursos a requerimiento. (Amazon Web Services, 2024).

2.2.3. Application Programming Interface (API)

Se define como un mecanismo que permite que dos componentes de programa puedan establecer comunicación a través de la red implementando un conjunto de definiciones y protocolos.

2.2.4. Base de datos

De acuerdo con (Pisco Gómez et al., 2017), surge en 1980 a raíz de la fuerte demanda de las empresas de contar con sistemas de información organizada y administrable; de esta manera, se puede definir como un conjunto estructurado y ordenado de datos variables en el tiempo organizado independientemente de las aplicaciones que los generan o que los consuman.

2.2.4.1. Modelo Relacional

Un modelo de base datos es el conjunto de herramientas conceptuales para la descripción de los datos y la relación entre ellos, actualmente el modelo relacional es el de uso más difundido dada su facilidad para representar el ordenamiento de la información almacenada y la ventaja que representa la incorporación de potentes mecanismos de consulta. Un modelo tiene como estructura fundamental a la relación, en este caso, la estructura relacional es una tabla bidimensional conformada por filas (tuplas) y columnas (atributos) (Vele Zhingri, 2015).

2.2.4.2. Modelo No Relacional

Más reciente que el modelo relacional, fue pensado para sistemas de crecimiento horizontal y que requieran dejar de depender la interdependencia de los datos para pasar a almacenarlos de manera agrupada por elementos comunes para que, de esta manera, la acción de consumir un dato o almacenarlo no requiera la interacción con múltiples bloques (Valverde et al., 2019).

2.2.5. Cliente Web

El cliente web es un programa a través del cual el usuario interactúa con los recursos de una aplicación alojada en el servidor a través de HTTP (Luján Mora, 2001).

2.2.6. Sistema Robótico Agrícola

Se puede definir como una máquina o conjunto coordinado de máquinas que integran sistemas electrónicos, mecánicos e informáticos para lograr automatismos que se puedan desplegar sobre las actividades de producción agrícola (De La Cruz Mariños, 2022).

De manera puntual y aplicando el concepto al sistema sobre el que se está desplegando la solución que implementa este estudio, se debe acotar que se trata de sistemas robóticos industriales de manipulación de múltiples grados de libertad, multifuncionales y reprogramables (Herrera German, 2022).

CAPÍTULO III

METODOLOGÍA

CAPÍTULO III: METODOLOGÍA

3.1. TIPO DE INVESTIGACIÓN

Esta investigación presenta una estructura preexperimental, lo cual condiciona una etapa pre-test en la que se aplique los instrumentos correspondientes con el fin de medir la realidad del problema previo a la implementación; seguida de una etapa post-test en la que se realice nuevamente la medición y de esta forma, a través de contraste entre ambas pruebas, se pueda evaluar de manera cuantitativa la influencia de la implementación sobre el problema.

De acuerdo con su:

Finalidad: Aplicada

Tipo: Explicativa

Paradigma: Cuantitativo

Contrastación: Preexperimental con pre-test y post-test

3.2. DISEÑO

3.2.1. Etapa Pre-test

- ✓ **Medición de Horas/Hombre en la generación de reporte de operación**
Realizada a través del registro del tiempo consumido en el proceso de la generación del reporte de operación.

- ✓ **Medición de Horas/Hombre en la difusión de reporte de operación**
Realizada a través del registro del tiempo consumido en el proceso de la difusión del reporte de operación.

- ✓ **Medición de Horas/Hombre en la generación-difusión de informe diario operación**
Realizada a través del registro del tiempo consumido en el proceso de la generación-difusión del informe diario de operación.

3.2.2. Implementación

✓ **Diseño inicial de Módulo de comunicación IoT BI**

Basado en la revisión de la documentación actual de proyecto PE501079170-2022-PROCIENCIA Agrobot, y en el algoritmo implementado; se realizó el listado de las características del módulo de comunicación IoT-BI.

✓ **Pruebas de Funcionalidad**

Las características listadas en el diseño inicial implementadas en Python se desplegaron en el entorno de pruebas del sistema robótico PE501079170-2022-PROCIENCIA Agrobot.

✓ **Corrección de errores**

En base a los errores que se presentaron en la etapa de pruebas, se implementaron las correcciones necesarias y repitieron las pruebas hasta obtener funcionalidad completa.

✓ **Despliegue de Módulo de comunicación IoT BI**

Habiendo completado la corrección de errores, el módulo de comunicación IoT-BI se desplegó en el entorno de producción del sistema robótico PE501079170-2022-PROCIENCIA Agrobot.

3.2.3. Estaba Post-test

✓ **Medición de Horas/Máquina en la generación de reporte de operación**

Realizada a través de la función “time.perf_counter()” de Python, se registró la hora de inicio y la hora de finalización de la generación del reporte de operación.

✓ **Medición de Horas/Hombre en la difusión de reporte de operación**

Realizada a través de la función “time.perf_counter()” de Python, se registró la hora de inicio y la hora de finalización de la generación del reporte de operación.

✓ **Medición de Horas/Máquina en la generación-difusión de informe diario operación**

Realizada a través del registro de la hora de inicio y la hora de finalización de la generación-difusión del informe diario de operación.

3.3. POBLACIÓN

La población de este estudio está conformada por sistemas robóticos de producción agrícola que integren visión artificial en una Plataforma Python-Enabled con conectividad a internet en Trujillo, 2024.

3.4. MUESTRA

Teniendo en consideración el requerimiento indispensable que representa el acceso al software y hardware del sistema robótico a analizar, la selección de la muestra se ha dado por conveniencia, conformando la totalidad de la muestra, el sistema robótico de producción agrícola Agrobot UPAO en Trujillo 2024; desarrollado por el LABINM de la Universidad Privada Antenor Orrego.

3.5. MÉTODOS DE RECOLECCIÓN DE DATOS

3.5.1. INSTRUMENTOS

La recolección de los datos necesarios para la contrastación cuantitativa de los tiempos de generación y difusión de reportes de operación, así como, el tiempo de generación-difusión del informe diario de productividad, está basada en la recolección de datos realizada en el estudio de (Loyola Diaz, 2022), que en su tesis para la obtención del grado de maestro en Maestro en Ingeniería de Sistemas con Mención en Tecnologías de la Información, requiere un análisis cuantitativo a través de la contrastación de los tiempos de generación de reportes por operarios vs los reportes autogenerados por el sistema que conforma su solución; en este orden de ideas, se hace uso de una guía de observación de tiempos en la que registra los datos del investigador, variable/proceso y los tiempos consumidos en la elaboración de reportes que se muestra en la Figura 1:

Figura 1

Instrumento guía de observación de tiempos de elaboración de reportes.

Guía de Observación N° 01 – Tiempo Promedio en Elaboración de Reportes		
Guía de observación de medición del tiempo en la elaboración de los reportes		
Investigador	Jhon Alexander Loyola Díaz	
Proceso / Variable	Toma de Decisiones	
PRETEST / POSTTEST		
N° de Obs.	Fecha	Tiempo consumido en la elaboración del Reporte
1		
2		
3		
4		
5		
6		
7		
8		
9		
10		

Nota. obtenido de (Loyola Diaz, 2022)

3.6. ASPECTOS ÉTICOS

Las Pruebas se han realizado en colaboración con el Laboratorio de Investigación Multidisciplinaria (LABINM) de la universidad Privada Antenor Orrego bajo el contexto de aporte al proceso de desarrollo y optimización del sistema robótico PE501079170-2022-PROCIENCIA.

El software usado para el desarrollo del módulo de comunicación IoT BI en el sistema robótico PE501079170-2022-PROCIENCIA es libre mientras que los recursos en la nube se encuentran sujetos al acuerdo de licencia de uso gratuito de los proveedores tanto de host de la base de datos, como del cliente web.

CAPÍTULO IV

RESULTADOS

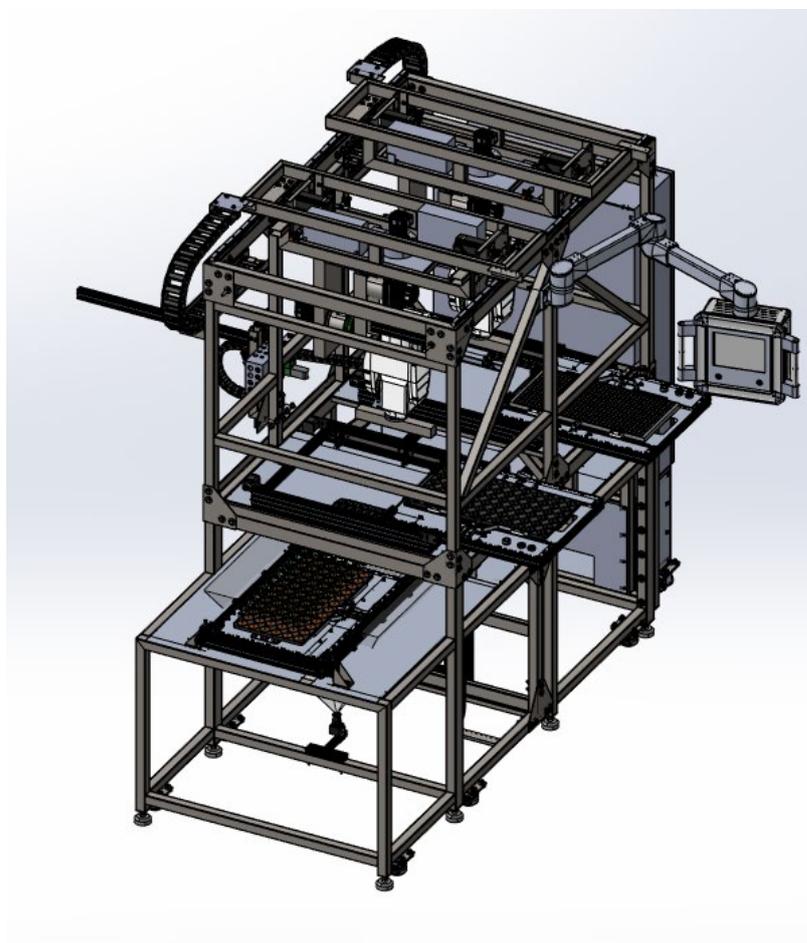
CAPÍTULO IV: RESULTADOS

4.1. Identificación de indicadores dentro de realidad problemática

El proyecto PE501079170-2022-PROCIENCIA desarrollado en el Laboratorio de Investigación Multidisciplinaria de la Universidad Privada Antenor Orrego es un sistema robótico multifuncional para la siembra de semillas de diferentes tamaños en bandejas multicelda y repique de plantines según calidad de crecimiento, físicamente, su estructura es la mostrada en la Figura 2.

Figura 2

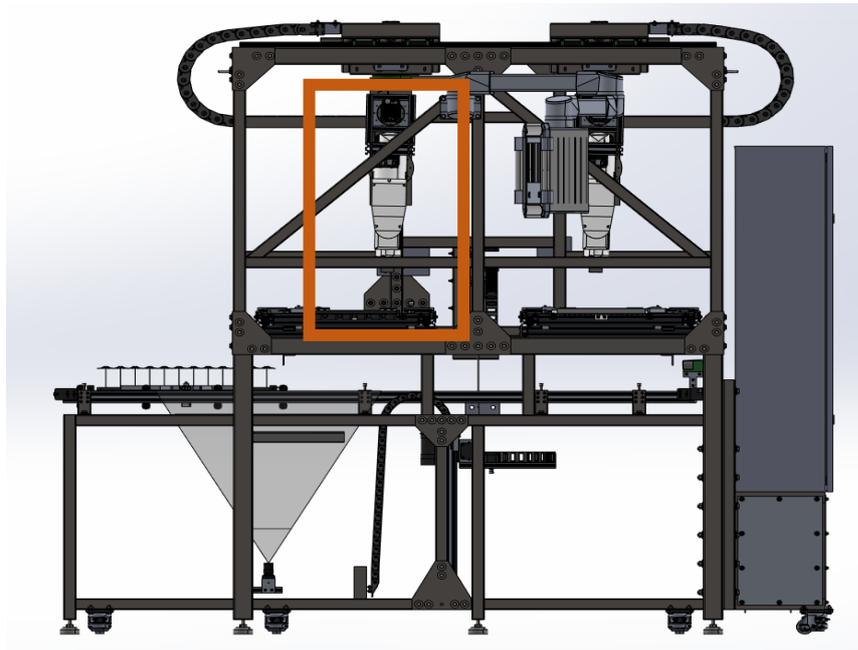
Vista isométrica frontal de modelo 3D de estructura de PE501079170-2022-PROCIENCIA.



El cual hace uso de un brazo robótico para la manipulación del contenido de las bandejas, éste está ubicado en la parte superior de la estructura como se muestra señalado en la Figura 3.

Figura 3

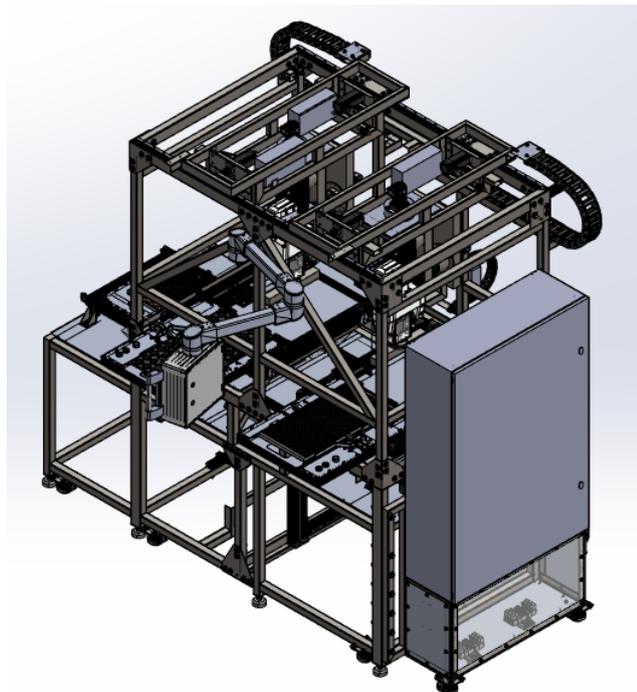
Vista lateral de modelo 3D de estructura de PE501079170-2022-PROCIENCIA.



Sus movimientos están gobernados por sistemas electrónicos de potencia ubicados en el gabinete lateral señalado en la Figura 4.

Figura 4

Vista isométrica posterior de modelo 3D de estructura de PE501079170-2022-PROCIENCIA.



El sistema electrónico de potencia está compuesto principalmente por un PLC y siete drivers de motor de pasos de alto torque, uno para cada grado de libertad del brazo manipulador y los correspondientes componentes de potencia, como se muestra en las Figuras 5a y 5b.

Figura 5

Estructura física de PE501079170-2022-PROCIENCIA.



(a)

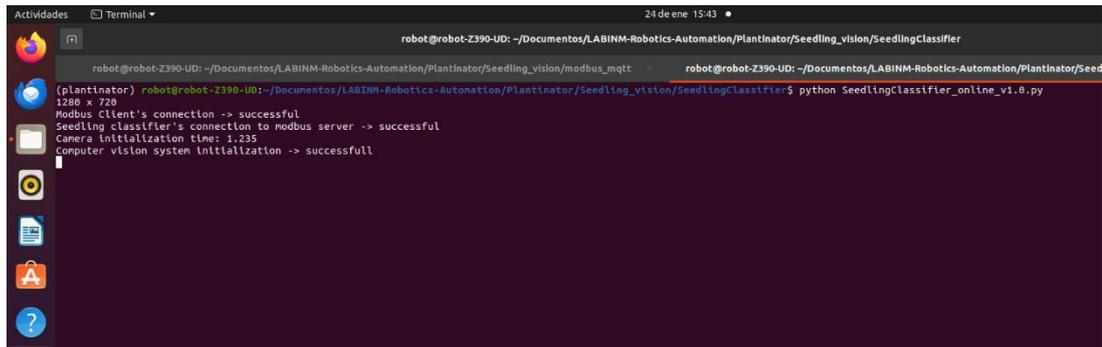


(b)

Si bien, es el controlador PLC el encargado de generar los pulsos necesarios para que los drivers PaP gestionen los pasos exactos para posicionar el brazo, estos datos de posición para el brazo y demás partes móviles, son generados a través de cálculos en el computador, el cual recibe los datos de la cámara y aplica el algoritmo de visión artificial para determinar en dónde debe posicionarse el brazo a través de algoritmo principal.

Este algoritmo, principalmente de visión artificial con funciones adicionales como comunicación a través de Modbus, está implementado en Python y desplegado en un sistema operativo Linux; con lo cual, la ejecución de éste se da a través de la terminal, como se muestra en la Figura 6.

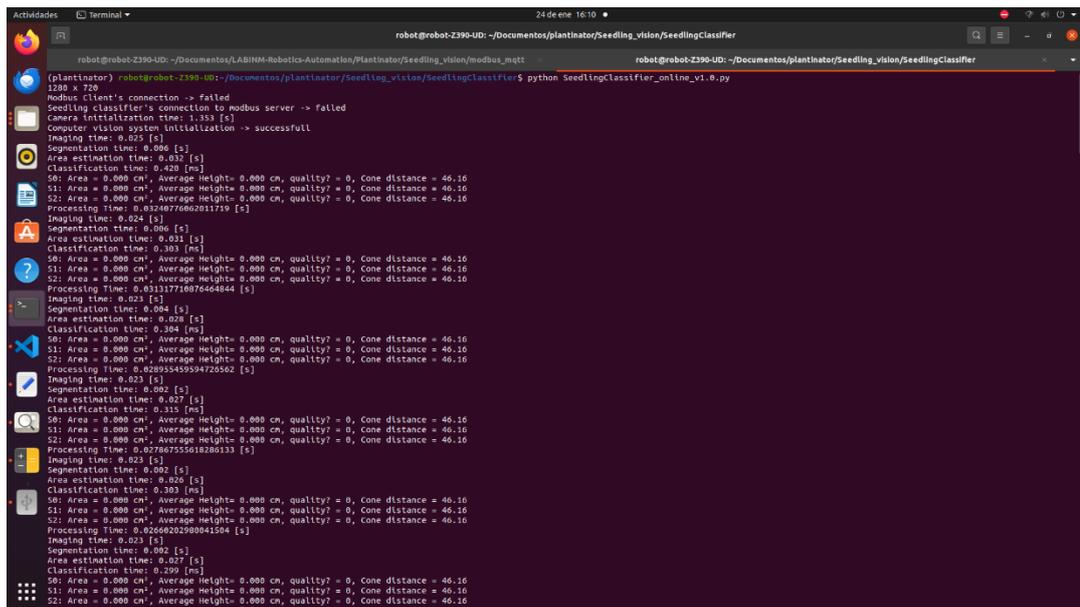
Figura 6
Terminal de Linux con el script principal en ejecución.



```
robot@robot-Z390-UD: ~/Documentos/LABINM-Robotics-Automation/Plantinator/Seedling_vision/SeedlingClassifier
robot@robot-Z390-UD: ~/Documentos/LABINM-Robotics-Automation/Plantinator/Seedling_vision/modbus_mqtt
robot@robot-Z390-UD: ~/Documentos/LABINM-Robotics-Automation/Plantinator/Seedling_vision/SeedlingClassifier$ python SeedlingClassifier_online_v1.0.py
1280 x 720
Modbus client's connection -> successful
Seedling classifier's connection to modbus server -> successful
Camera initialization time: 1.235
Computer vision system initialization -> successful
```

Los resultados del procesamiento son enviados a través de Modbus al controlador PLC para que ejecute las acciones mecánicas correspondientes y los datos del procesamiento, así como los tiempos de ejecución, son mostrados como se aprecia en la Figura 7.

Figura 7
Resultados del procesamiento mostrados en la terminal.



```
robot@robot-Z390-UD: ~/Documentos/LABINM-Robotics-Automation/Plantinator/Seedling_vision/SeedlingClassifier
robot@robot-Z390-UD: ~/Documentos/LABINM-Robotics-Automation/Plantinator/Seedling_vision/modbus_mqtt
robot@robot-Z390-UD: ~/Documentos/LABINM-Robotics-Automation/Plantinator/Seedling_vision/SeedlingClassifier$ python SeedlingClassifier_online_v1.0.py
1280 x 720
Modbus client's connection -> failed
Seedling classifier's connection to modbus server -> failed
Camera initialization time: 1.353 [s]
Computer vision system initialization -> successful
Timing time: 0.025 [s]
Segmentation time: 0.006 [s]
Area estimation time: 0.032 [s]
Classification time: 0.420 [ms]
50: Area = 0.000 cm², Average Height= 0.000 cm, quality? = 0, Cone distance = 40.10
51: Area = 0.000 cm², Average Height= 0.000 cm, quality? = 0, Cone distance = 40.10
52: Area = 0.000 cm², Average Height= 0.000 cm, quality? = 0, Cone distance = 40.10
Processing Time: 0.03240776062011719 [s]
Timing time: 0.024 [s]
Segmentation time: 0.006 [s]
Area estimation time: 0.032 [s]
Classification time: 0.303 [ms]
50: Area = 0.000 cm², Average Height= 0.000 cm, quality? = 0, Cone distance = 40.10
51: Area = 0.000 cm², Average Height= 0.000 cm, quality? = 0, Cone distance = 40.10
52: Area = 0.000 cm², Average Height= 0.000 cm, quality? = 0, Cone distance = 40.10
Processing Time: 0.031317710876464844 [s]
Timing time: 0.023 [s]
Segmentation time: 0.004 [s]
Area estimation time: 0.028 [s]
Classification time: 0.304 [ms]
50: Area = 0.000 cm², Average Height= 0.000 cm, quality? = 0, Cone distance = 40.10
51: Area = 0.000 cm², Average Height= 0.000 cm, quality? = 0, Cone distance = 40.10
52: Area = 0.000 cm², Average Height= 0.000 cm, quality? = 0, Cone distance = 40.10
Processing Time: 0.02895455994725502 [s]
Timing time: 0.023 [s]
Segmentation time: 0.002 [s]
Area estimation time: 0.027 [s]
Classification time: 0.315 [ms]
50: Area = 0.000 cm², Average Height= 0.000 cm, quality? = 0, Cone distance = 40.10
51: Area = 0.000 cm², Average Height= 0.000 cm, quality? = 0, Cone distance = 40.10
52: Area = 0.000 cm², Average Height= 0.000 cm, quality? = 0, Cone distance = 40.10
Processing Time: 0.027867555618286133 [s]
Timing time: 0.023 [s]
Segmentation time: 0.002 [s]
Area estimation time: 0.026 [s]
Classification time: 0.303 [ms]
50: Area = 0.000 cm², Average Height= 0.000 cm, quality? = 0, Cone distance = 40.10
51: Area = 0.000 cm², Average Height= 0.000 cm, quality? = 0, Cone distance = 40.10
52: Area = 0.000 cm², Average Height= 0.000 cm, quality? = 0, Cone distance = 40.10
Processing Time: 0.02660202980041504 [s]
Timing time: 0.023 [s]
Segmentation time: 0.002 [s]
Area estimation time: 0.027 [s]
Classification time: 0.299 [ms]
50: Area = 0.000 cm², Average Height= 0.000 cm, quality? = 0, Cone distance = 40.10
51: Area = 0.000 cm², Average Height= 0.000 cm, quality? = 0, Cone distance = 40.10
52: Area = 0.000 cm², Average Height= 0.000 cm, quality? = 0, Cone distance = 40.10
Processing Time: 0.02660202980041504 [s]
```

Dado que los datos generados por el programa son visibles únicamente en la terminal, la generación del informe requiere que el operador los extraiga hacia una hoja de cálculo en donde sea posible organizarlos.

Posteriormente debe revisar la lista de personas que deben recibir esta información, en este caso, los desarrolladores, investigadores que están realizando estudios y análisis sobre PE501079170-2022-PROCIENCIA y el investigador líder, a los cuales,

se les hacen llegar los reportes correspondientes a los lotes de bandejas que son analizados.

Al terminar la producción del día, se realiza un informe que contiene un resumen de la producción diaria, así como un resumen de las incidencias que puedan haberse dado durante el día; este informe va dirigido al investigador líder.

4.2. Aplicación de instrumento en etapa pre-test

A los procedimientos previamente descritos, se les aplicó la guía de observación de tiempos para registrar la duración de cada tarea que los conforma.

4.2.1. Medición de tiempos de generación de reporte de operación

De acuerdo con el proceso analizado, la generación de reporte está compuesta por los procedimientos:

- ✓ Ejecución de algoritmo de reconocimiento
- ✓ Extracción de data cruda de la terminal
- ✓ Procesamiento de datos en hoja de cálculo

De tal manera que, aplicando la guía de observación de tiempos a 8 lotes de 6 bandejas cada uno, se obtienen los tiempos comprendidos en la Tabla 2

Tabla 2*Tiempos por subproceso de la generación de reporte de producción pre-test.*

LOTE	TIEMPO POR SUBPROCESO	B1	B2	B3	B4	B5	B6	SUBTOTAL
L1	tiempo algoritmo (S)	0.04860	0.05015	0.05003	0.05082	0.05168	0.05331	0.30459
	tiempo extracción (S)	22.7873	26.5606	22.8673	27.5448	27.3507	21.4768	148.58746
	tiempo procesamiento (S)	245.347	251.767	262.472	259.801	258.315	251.679	1529.38002
L2	tiempo algoritmo (S)	0.05144	0.04953	0.05239	0.05233	0.05479	0.04841	0.30890
	tiempo extracción (S)	25.1809	26.1501	24.1653	23.3027	27.5413	28.9304	155.27058
	tiempo procesamiento (S)	255.613	245.423	243.025	257.868	240.655	259.574	1502.15666
L3	tiempo algoritmo (S)	0.05078	0.05350	0.04832	0.05112	0.04928	0.04873	0.30175
	tiempo extracción (S)	24.3513	21.3933	26.4091	21.9714	28.9091	24.8255	147.85975
	tiempo procesamiento (S)	258.885	246.314	244.232	250.748	249.469	245.2	1494.84757
L4	tiempo algoritmo (S)	0.05311	0.05349	0.05463	0.05588	0.05479	0.05501	0.32691
	tiempo extracción (S)	29.2793	29.2738	21.6824	21.571	25.6459	25.1433	152.59575
	tiempo procesamiento (S)	248.437	264.455	248.894	263.88	243.644	259.818	1529.12784
L5	tiempo algoritmo (S)	0.05184	0.05070	0.05157	0.05488	0.05134	0.05362	0.31395
	tiempo extracción (S)	21.4678	27.1634	26.8749	28.8602	28.4722	26.0693	158.90782
	tiempo procesamiento (S)	244.208	262.806	255.061	265.407	253.402	265.924	1546.80799
L6	tiempo algoritmo (S)	0.05341	0.05296	0.05512	0.05331	0.04990	0.05106	0.31575
	tiempo extracción (S)	24.2856	24.602	22.3205	25.4704	27.4198	24.5867	148.68484
	tiempo procesamiento (S)	248.453	249.739	264.64	259.211	245.506	265.248	1532.79682
L7	tiempo algoritmo (S)	0.05083	0.04939	0.04819	0.05032	0.05143	0.05057	0.30072
	tiempo extracción (S)	23.8512	28.8737	27.9516	25.4932	21.6519	24.1105	151.93199
	tiempo procesamiento (S)	244.541	266.181	264.936	240.609	261.212	264.225	1541.70449
L8	tiempo algoritmo (S)	0.04933	0.05530	0.05452	0.05138	0.04836	0.05350	0.31240
	tiempo extracción (S)	22.4177	22.1122	27.5933	22.913	27.8319	21.5977	144.46578
	tiempo procesamiento (S)	242.927	243.183	249.762	257.064	261.269	249.83	1504.03474
TOTAL								13391.64508

De acuerdo con los tiempos detallados en la Tabla 2, el menor tiempo lo consume la ejecución del algoritmo de procesamiento de imágenes, seguido de la extracción de datos de la terminal y por último, siendo el más tardado, el tiempo de procesamiento que requieren estos datos en la hoja de cálculo para

separarse y cambiar de formato para, de esta forma, generar una tabla ordenada que conforme el reporte; obteniéndose, en un total en los 8 lotes observados, una duración total de 13391.64508 segundos, lo que equivale aproximadamente a 223.2 minutos o 3.7 horas.

4.2.2. Medición de tiempos de difusión de reporte de operación

El subproceso de difusión de reporte de operación consiste en las siguientes tareas:

- ✓ Verificación por parte del operador de la lista de direcciones de correos de los interesados
- ✓ Redacción y envío de correo electrónico con el reporte de producción

De tal manera que, aplicando la guía de observación de tiempos a 8 lotes de 6 bandejas cada uno, se obtienen los tiempos comprendidos en la Tabla 3.

Tabla 3
Tiempos por subproceso de la difusión de reporte de producción pre-test.

LOTE	TIEMPO POR SUBPROCESO	B1	B2	B3	B4	B5	B6	SUBTOTAL
L1	tiempo de verificación de lista de correos	34.87	33.76	29.77	35.65	35.98	30.62	200.65
	tiempo de redacción y envío de correo	20.51	19.77	19.95	22.14	23.27	21.73	127.38
L2	tiempo de verificación de lista de correos	32.01	34.27	30.52	33.04	31.01	31.61	192.46
	tiempo de redacción y envío de correo	23.79	22.50	21.03	19.21	22.83	21.03	130.39
L3	tiempo de verificación de lista de correos	31.35	30.31	35.53	30.16	30.80	33.57	191.72
	tiempo de redacción y envío de correo	21.11	20.66	24.45	19.56	19.27	20.58	125.64
L4	tiempo de verificación de lista de correos	34.92	35.10	34.59	30.79	30.59	33.76	199.74
	tiempo de redacción y envío de correo	25.35	21.41	19.73	25.06	25.01	21.60	138.16
L5	tiempo de verificación de lista de correos	31.74	35.99	31.06	34.41	31.17	31.36	195.73
	tiempo de redacción y envío de correo	23.44	24.02	19.92	24.66	19.33	21.98	133.34
L6	tiempo de verificación de lista de correos	34.82	33.51	33.50	30.03	34.27	33.52	199.65
	tiempo de redacción y envío de correo	19.59	19.70	19.76	24.24	22.60	22.71	128.60
L7	tiempo de verificación de lista de correos	30.32	30.65	33.13	29.94	34.84	30.40	189.28
	tiempo de redacción y envío de correo	19.70	24.94	19.27	24.57	20.94	19.70	129.13
L8	tiempo de verificación de lista de correos	34.40	31.86	30.81	32.19	36.38	33.42	199.06
	tiempo de redacción y envío de correo	21.80	24.04	20.93	19.86	25.05	24.88	136.56
TOTAL								2617.483

De acuerdo con la data mostrada en la Tabla 3, la duración de los dos subprocesos no difiere significativamente, no obstante, la actividad de mayor

duración es verificar la lista de correos dado que esta se encuentra alojada en una hoja de cálculo compartida en drive para que pueda ser actualizada por el líder del proyecto, mientras que la redacción y envío del correo se trata de colocar los correos de destino verificados, adjuntar el archivo y, si es necesario, mencionar alguna incidencia.

4.2.3. Medición de tiempos de generación-difusión de informe diario de operación

El subproceso de generación-difusión de informe diario de operación consiste en las siguientes tareas:

- ✓ Generación de tablas resumen
- ✓ Generación de gráficas de tablas resumen
- ✓ Redacción de informe con incidencias
- ✓ Verificación por parte del operador de la lista de direcciones de correos de los interesados
- ✓ Redacción y envío de correo electrónico con el informe diario de producción

De tal manera que, aplicando la guía de observación de tiempos al proceso de generación-difusión de informe de producción, realizado una vez por día, se obtienen los tiempos comprendidos en la Tabla 4.

Tabla 4

Tiempos por subproceso de la generación- difusión de informe diario de producción pre-test.

<i>LOTE</i>	<i>TIEMPOS POR ACTIVIDAD</i>	<i>DURACIÓN (S)</i>
<i>L1</i>	Generación de tablas resumen (S)	201.89
	Generación de gráficas de tablas resumen (S)	68.3
	Redacción de informe con incidencias (S)	838.77
	Verificación de lista de correos de los interesados (S)	34.18
	Redacción y envío de correo con el informe diario (S)	24.84
<i>L2</i>	Generación de tablas resumen con gráficas (S)	177.64
	Generación de tablas resumen (S)	76.68
	Redacción de informe con incidencias (S)	827.51
	Verificación de lista de correos de los interesados (S)	35.57
	Redacción y envío de correo con el informe diario (S)	21.01
<i>L3</i>	Generación de tablas resumen con gráficas (S)	168.23
	Generación de tablas resumen (S)	60.91
	Redacción de informe con incidencias (S)	783.05
	Verificación de lista de correos de los interesados (S)	36.2
	Redacción y envío de correo con el informe diario (S)	23.39
<i>L4</i>	Generación de tablas resumen con gráficas (S)	200.56
	Generación de tablas resumen (S)	70.76
	Redacción de informe con incidencias (S)	662.18
	Verificación de lista de correos de los interesados (S)	32.49
	Redacción y envío de correo con el informe diario (S)	22.22
<i>L5</i>	Generación de tablas resumen con gráficas (S)	204.48
	Generación de tablas resumen (S)	69.73
	Redacción de informe con incidencias (S)	640.06
	Verificación de lista de correos de los interesados (S)	35.35
	Redacción y envío de correo con el informe diario (S)	22.51
<i>L6</i>	Generación de tablas resumen con gráficas (S)	164.29
	Generación de tablas resumen (S)	87.99
	Redacción de informe con incidencias (S)	711.69
	Verificación de lista de correos de los interesados (S)	32.87
	Redacción y envío de correo con el informe diario (S)	19.94
<i>L7</i>	Generación de tablas resumen con gráficas (S)	165.15
	Generación de tablas resumen (S)	65.44
	Redacción de informe con incidencias (S)	563.55
	Verificación de lista de correos de los interesados (S)	32.56
	Redacción y envío de correo con el informe diario (S)	22.4
<i>L8</i>	Generación de tablas resumen con gráficas (S)	188
	Generación de tablas resumen (S)	65.01
	Redacción de informe con incidencias (S)	783.79
	Verificación de lista de correos de los interesados (S)	30.45
	Redacción y envío de correo con el informe diario (S)	19.36
TOTAL		8291.00

De acuerdo con los datos mostrados en la Tabla 4, las tareas de mayor duración son la generación de tablas resumen y la redacción del informe ya que éste implica redacción, formato y demás procedimientos propios de la generación de un documento administrativo.

4.2.4. Resumen de tiempos totales

En concordancia con la data mostrada en las Tablas 2, 3 y 4 se puede extraer los siguientes tiempos en la Tabla 5 con la finalidad de resumir y dar una vista general de los tiempos consumidos por actividad en el proceso de monitoreo de producción del sistema robótico PE501079170-2022-PROCIENCIA.

Tabla 5
Resumen de tiempos de subprocesos de monitoreo de producción de PE501079170-2022-PROCIENCIA pre-test.

<i>LOTE</i>	<i>TIEMPO POR ACTIVIDAD</i>	<i>DURACIÓN (S)</i>	<i>DURACIÓN TOTAL POR LOTE (S)</i>
<i>L1</i>	Generación de reporte de operación (S)	1678.27	3174.28
	Difusión de reporte de operación (S)	328.03	
	Generación-difusión de informe diario de operación (S)	1167.98	
<i>L2</i>	Generación de reporte de operación (S)	1657.74	3119.00
	Difusión de reporte de operación (S)	322.86	
	Generación-difusión de informe diario de operación (S)	1138.41	
<i>L3</i>	Generación de reporte de operación (S)	1643.01	3032.14
	Difusión de reporte de operación (S)	317.35	
	Generación-difusión de informe diario de operación (S)	1071.78	
<i>L4</i>	Generación de reporte de operación (S)	1682.05	3008.16
	Difusión de reporte de operación (S)	337.90	
	Generación-difusión de informe diario de operación (S)	988.21	
<i>L5</i>	Generación de reporte de operación (S)	1706.03	3007.23
	Difusión de reporte de operación (S)	329.07	
	Generación-difusión de informe diario de operación (S)	972.13	
<i>L6</i>	Generación de reporte de operación (S)	1681.80	3026.83
	Difusión de reporte de operación (S)	328.25	
	Generación-difusión de informe diario de operación (S)	1016.78	
<i>L7</i>	Generación de reporte de operación (S)	1693.94	2861.44
	Difusión de reporte de operación (S)	318.40	
	Generación-difusión de informe diario de operación (S)	849.10	
<i>L8</i>	Generación de reporte de operación (S)	1648.81	3071.04
	Difusión de reporte de operación (S)	335.62	
	Generación-difusión de informe diario de operación (S)	1086.61	
TOTAL			24300.13

De acuerdo con la Tabla 5 la tarea que más tiempo consume es la generación del reporte de operación, seguida de cerca por la generación-difusión de informe diario de operación y por último y con una duración menor, la difusión de reporte de operación

4.3. Desarrollo del módulo de comunicación IoT-BI

4.3.1. Elaboración de lista de requerimientos

En base a las tareas y subprocesos tratados en el punto 4.2, se establecen las funcionalidades requeridas al módulo de comunicación IoT-BI y de estas, las capacidades que debe tener en sus diferentes instancias tanto en el entorno de despliegue, como en la nube y el cliente web; esta información se muestra en la Tabla 6.

Tabla 6
Correspondencia de tareas con capacidades requeridas del módulo IoT-BI.

<i>TAREA</i>	<i>CAPACIDAD DEL MÓDULO IoT-BI</i>
1 Ejecución de algoritmo de reconocimiento	-
2 Extracción de data cruda de la terminal	Exportar la información en un formato de archivo de lista legible y enviar totales a la base de datos
3 Procesamiento de datos en hoja de cálculo	Ordenar los datos en formato de tabla
4 Verificación por parte del operador de la lista de direcciones de correos de los interesados	Leer de la lista de direcciones de correo de interesados del archivo compartido en drive
5 Redacción y envío de correo electrónico con el reporte de producción	Redactar y enviar de correo electrónico adjuntando reporte de producción
6 Generación de tablas resumen	Extraer información de los reportes generados durante el día de la base de datos para generar tabla resumen
7 Generación de gráficas de tablas resumen	Generar gráficos con la información de la base de datos
8 Redacción de informe con incidencias	Leer el listado de incidencias del archivo compartido en drive e incluirlas en el informe diario, de ser necesario
9 Verificación por parte del operador de la lista de direcciones de correos de los interesados	Leer de la lista de direcciones de correo de interesados del archivo compartido en drive
10 Redacción y envío de correo electrónico con el informe diario de producción	Redactar y enviar de correo electrónico adjuntando informe diario de producción

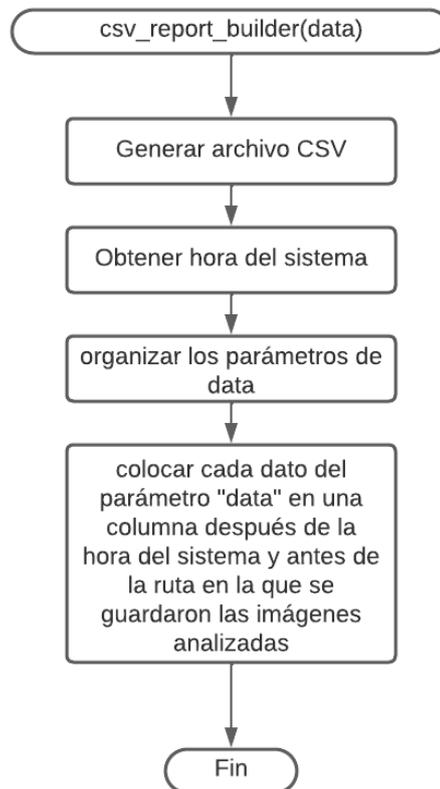
Tal y como se aprecia en la Tabla 6, no hay una capacidad para el módulo de comunicación IoT-BI para la actividad de procesamiento de imágenes, dado que, por su naturaleza, la actividad del módulo de comunicación empieza después de que este subproceso ha terminado y usa la data generada por éste para la generación y difusión de reportes e informes.

4.3.2. Implementación de requerimientos en bloques funcionales

- a. **Extracción de data cruda de la terminal:** esta tarea se debe realizar automáticamente a través de la generación de un archivo de tipo lista con el fin de implementar los procesos de comunicación con la base de datos sobre este archivo y no sobre el script principal, en pro de no variar el tiempo de procesamiento y poder establecer la periodicidad del envío de datos de manera independiente.

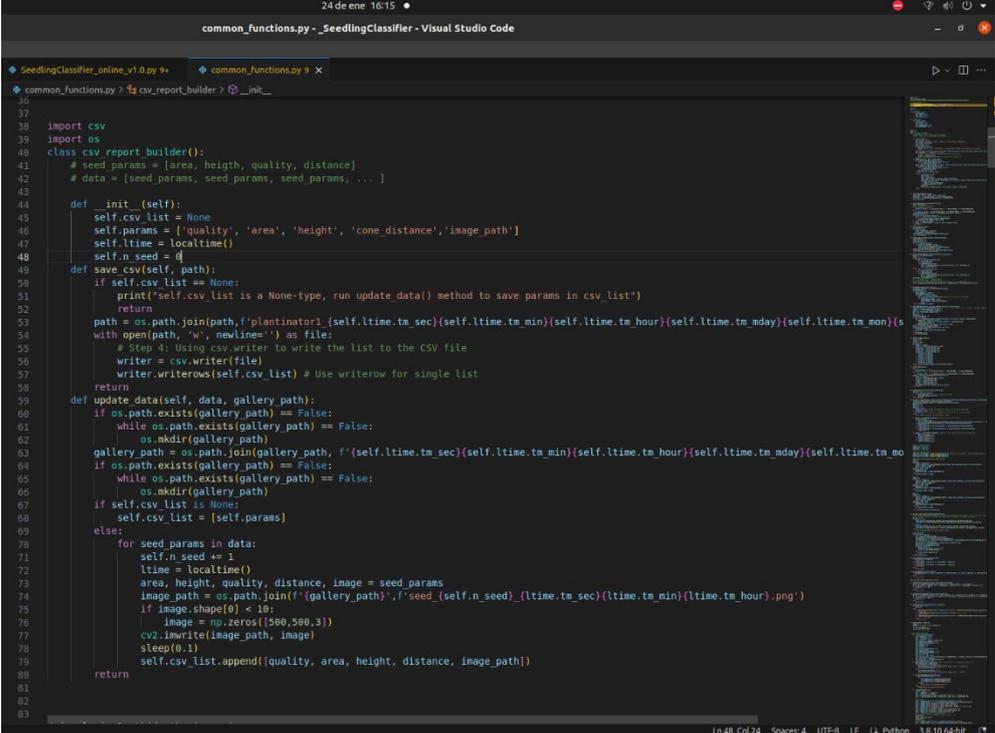
Figura 8

Diagrama de bloques de script generador de reporte.



En la Figura 8 se muestra el diagrama de flujo de la función realizada para esta tarea, la cual será la única que se desplegará incluyéndose en el script principal.

Figura 9
Implementación de script generador de reporte en Python.



```
24 de ene 16:15
common_functions.py - SeedingClassifier - Visual Studio Code

SeedingClassifier_online_v1.0.py x
common_functions.py x
common_functions.py > csv_report_builder > _init_

36
37
38 import csv
39 import os
40 class csv_report_builder():
41     # seed_params = [area, height, quality, distance]
42     # data = [seed_params, seed_params, seed_params, ... ]
43
44     def __init__(self):
45         self.csv_list = None
46         self.params = ['quality', 'area', 'height', 'cone_distance', 'image_path']
47         self.ltime = localtime()
48         self.n_seed = 0
49     def save_csv(self, path):
50         if self.csv_list == None:
51             print("self.csv_list is a None-type, run update_data() method to save params in csv_list")
52             return
53         path = os.path.join(path, f'plantinator1_{self.ltime.tm_sec}{self.ltime.tm_min}{self.ltime.tm_hour}{self.ltime.tm_mday}{self.ltime.tm_mon}{self.ltime.tm_year}.csv')
54         with open(path, 'w', newline='') as file:
55             # Step 4: Using csv.writer to write the list to the CSV file
56             writer = csv.writer(file)
57             writer.writerow(self.csv_list) # Use writerow for single list
58         return
59     def update_data(self, data, gallery_path):
60         if os.path.exists(gallery_path) == False:
61             while os.path.exists(gallery_path) == False:
62                 os.mkdir(gallery_path)
63             gallery_path = os.path.join(gallery_path, f'{self.ltime.tm_sec}{self.ltime.tm_min}{self.ltime.tm_hour}{self.ltime.tm_mday}{self.ltime.tm_mon}{self.ltime.tm_year}.')
64         if os.path.exists(gallery_path) == False:
65             while os.path.exists(gallery_path) == False:
66                 os.mkdir(gallery_path)
67         if self.csv_list is None:
68             self.csv_list = [self.params]
69         else:
70             for seed_params in data:
71                 self.n_seed += 1
72                 ltime = localtime()
73                 area, height, quality, distance, image = seed_params
74                 image_path = os.path.join(f'{gallery_path}', f'seed_{self.n_seed}_{ltime.tm_sec}_{ltime.tm_min}_{ltime.tm_hour}.png')
75                 if image.shape[0] < 10:
76                     image = np.zeros((500, 500, 3))
77                 cv2.imwrite(image_path, image)
78                 sleep(0.1)
79                 self.csv_list.append([quality, area, height, distance, image_path])
80         return
81
82
83
```

De acuerdo con la Figura 9, en la línea 45 a la 48 se muestra cómo es que el parámetro data está compuesto por la hora, un número de orden, correspondiente al número de semilla de la bandeja y los parámetros evaluados en esta como área, altura, ruta en la que se guarda la imagen evaluada y la calidad de la semilla.

b. Ordenamiento de datos en formato de tabla:

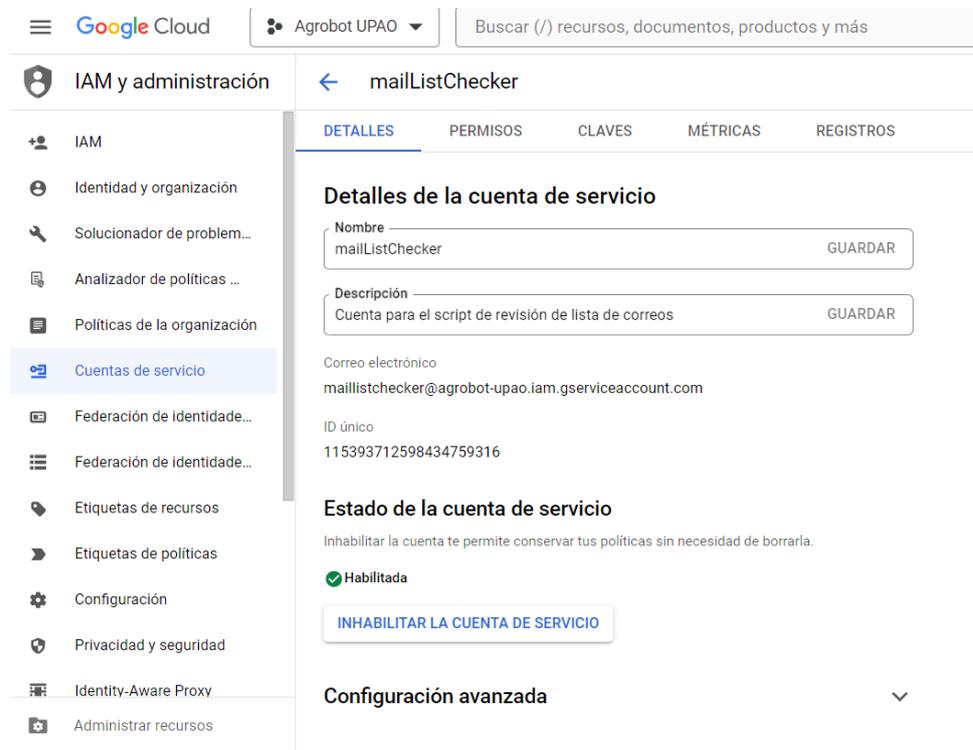
A partir de la línea 70 a 80, mostradas en la Figura 9 se observa cómo es que a través de un bucle For, se ordena la información en formato de valores separados por comas dentro del archivo creado en las líneas de la 49 a la 58 que tiene como nombre la información de hora y fecha de su generación.

c. **Leer de la lista de direcciones de correo de interesados del archivo compartido en drive:**

Para realizar esta tarea, es necesario que la hoja de cálculo que contiene la lista de direcciones de correo de los interesados esté compartida con la cuenta de correo de servicios de la aplicación, para la cual, se generó la cuenta de servicios: maillistchecker@agrobot-upao.iam.gserviceaccount.com, y se habilitó como se muestra en la Figura 10.

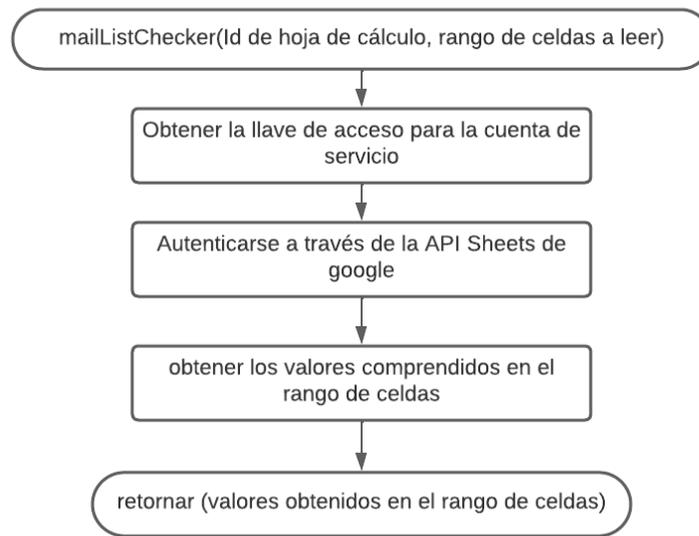
Figura 10

Parámetros de la cuenta de servicio para acceso a archivos compartidos.



Esta cuenta de servicio generó la llave de acceso en formato JSON para que el script pueda acceder y de esta manera se implementaron los pasos detallados en el diagrama de flujo que muestra la Figura 11.

Figura 11
Diagrama de flujo de script lector de direcciones de correo.



Este script es completamente independiente del script principal, encargado del procesamiento de imágenes y se implementó en el archivo mailListChecker.py, mostrado en la Figura 12.

Figura 12
Implementación de script lector de direcciones de correo en Python.

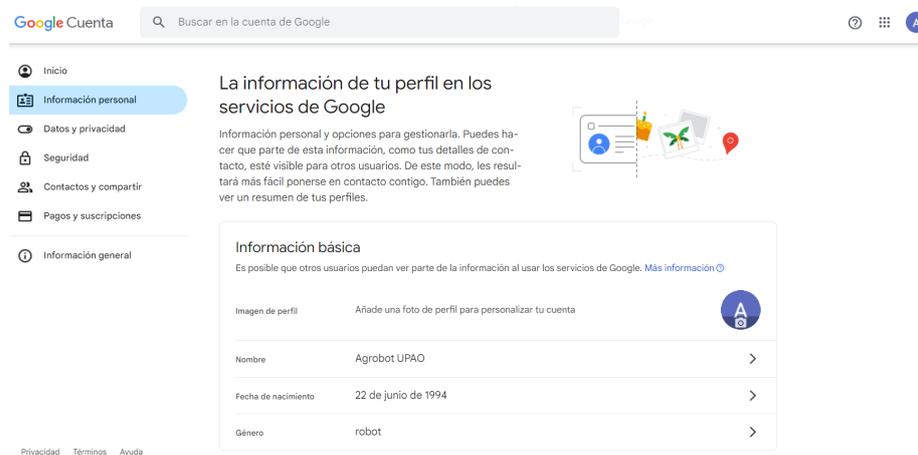
```
Code > mailListChecker.py > read_sheet
4 def read_sheet(spreadsheet_id, range_name):
5     # Ruta al archivo de clave de la cuenta de servicio
6     SERVICE_ACCOUNT_FILE = 'C:/Users/jmend/OneDrive/Documents/TesisMasterJM2023/Code/credenciales4mailChecking.json'
7
8     # Define los scopes requeridos
9     SCOPES = ['https://www.googleapis.com/auth/spreadsheets.readonly']
10
11    # Autentica usando la cuenta de servicio
12    credentials = service_account.Credentials.from_service_account_file(
13        SERVICE_ACCOUNT_FILE, scopes=SCOPES)
14
15    # Construye el servicio
16    service = build('sheets', 'v4', credentials=credentials)
17
18    # Realiza la solicitud a la API
19    sheet = service.spreadsheets()
20    result = sheet.values().get(spreadsheetId=spreadsheet_id, range=range_name).execute()
21    values = result.get('values', [])
22
23    return values # Devuelve los valores leídos
```

d. Redacción y envío de correo electrónico con el reporte de producción:

Esta actividad, haciendo uso de la data recuperada por el script mailListChecker.py, se encargó de generar un correo electrónico dirigido a todos los correos encontrados en la hoja de cálculo compartida con la cuenta de servicio, para ello fue necesario usar el servicio SMTP de Google y una

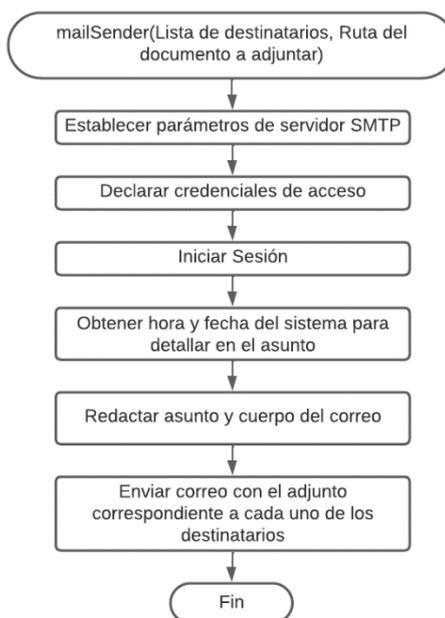
cuenta de correo que sea exclusiva para este fin, de esta forma, se creó la cuenta agrobotupao@gmail.com mostrada en la Figura 13.

Figura 13
Parámetros de dirección de correo del script gestor de correos.



A la cual, se le generó una contraseña de aplicación para permitir que el script mailSender.py redacte a través de ella el correo que contenga el reporte, de acuerdo con los pasos mostrados en el diagrama de flujo de la Figura 14.

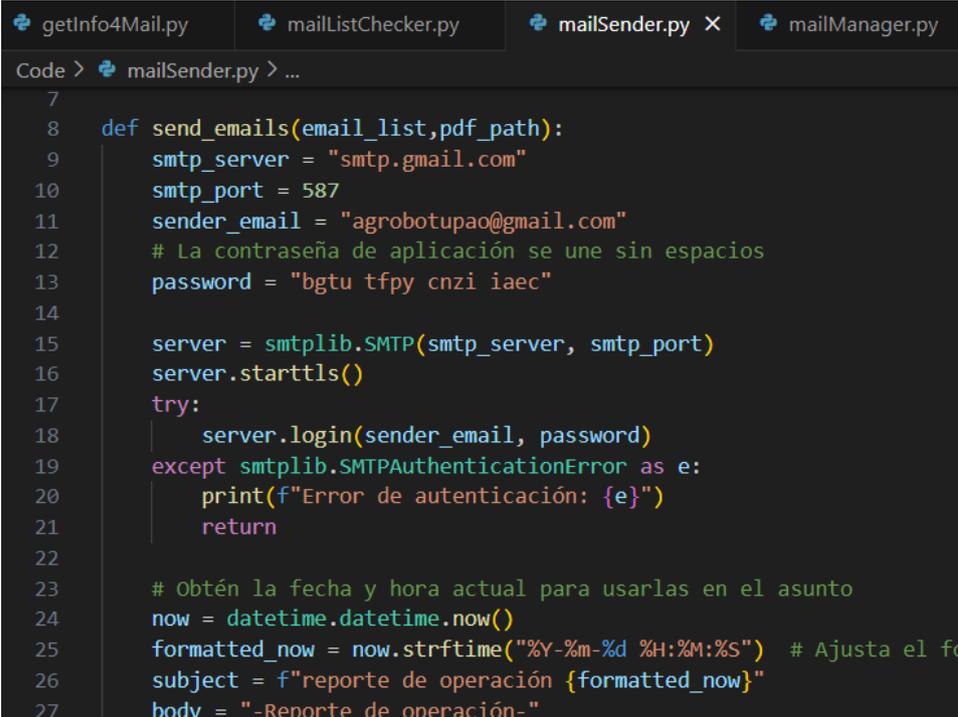
Figura 14
Diagrama de flujo de script gestor de correos.



Es te orden de ideas, se generó el script mailSender.py, mostrado en la Figura 15.

Figura 15

Implementación de script gestor de correos en Python líneas 7-27.

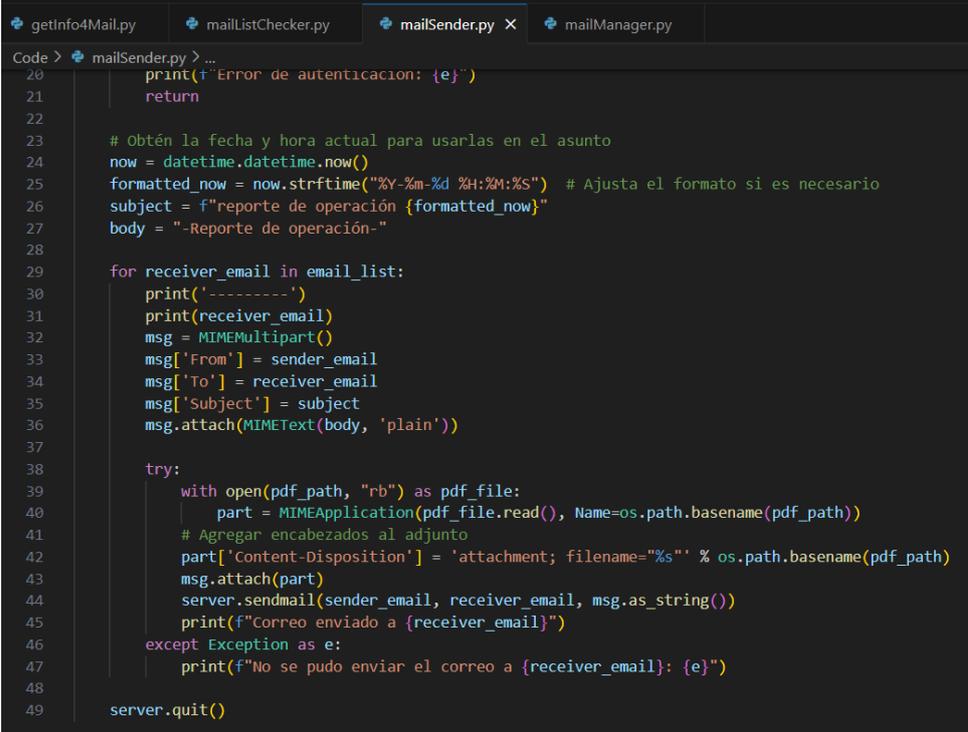


```
getInfo4Mail.py  mailListChecker.py  mailSender.py X  mailManager.py
Code > mailSender.py > ...
7
8 def send_emails(email_list, pdf_path):
9     smtp_server = "smtp.gmail.com"
10    smtp_port = 587
11    sender_email = "agrobotupao@gmail.com"
12    # La contraseña de aplicación se une sin espacios
13    password = "bgtu tfpy cnzi iaec"
14
15    server = smtplib.SMTP(smtp_server, smtp_port)
16    server.starttls()
17    try:
18        server.login(sender_email, password)
19    except smtplib.SMTPAuthenticationError as e:
20        print(f"Error de autenticación: {e}")
21        return
22
23    # Obtén la fecha y hora actual para usarlas en el asunto
24    now = datetime.datetime.now()
25    formatted_now = now.strftime("%Y-%m-%d %H:%M:%S") # Ajusta el fo
26    subject = f"reporte de operación {formatted_now}"
27    body = "-Reporte de operación-"
```

En donde se aprecia que la acción de iniciar sesión después de establecer los parámetros del servidor SMTP, se introduce en una estructura Try Except, con la finalidad de que, en caso exista un error en este proceso, será detallado en la terminal para poder ser evaluado y resuelto.

Figura 16

Implementación de script gestor de correos en Python líneas 24 a 49.



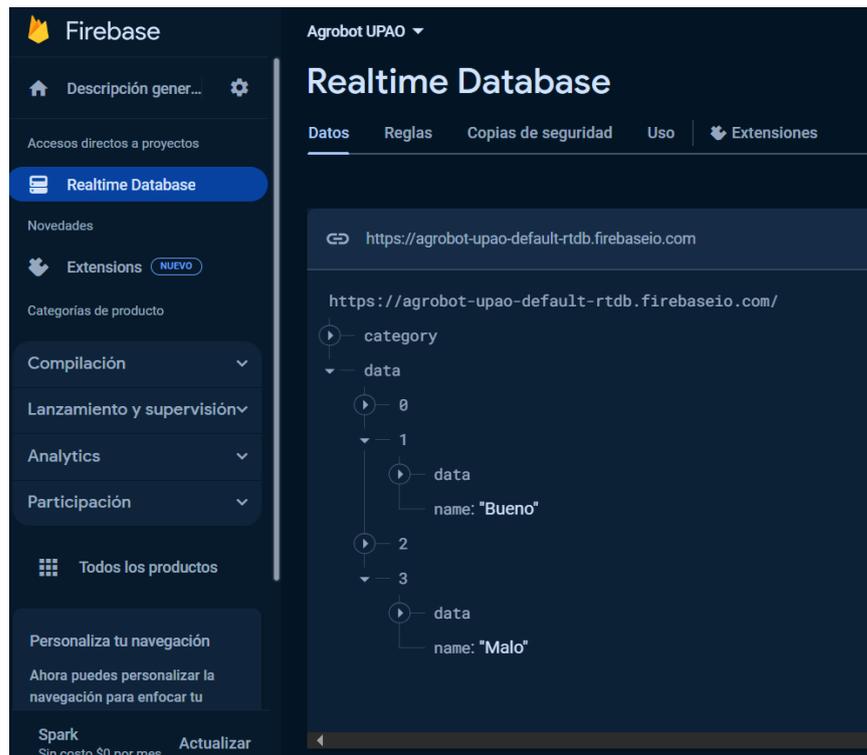
```
Code > mailSender.py > ...
20     print(f"Error de autenticación: {e}")
21     return
22
23     # Obtén la fecha y hora actual para usarlas en el asunto
24     now = datetime.datetime.now()
25     formatted_now = now.strftime("%Y-%m-%d %H:%M:%S") # Ajusta el formato si es necesario
26     subject = f"reporte de operación {formatted_now}"
27     body = "-Reporte de operación-"
28
29     for receiver_email in email_list:
30         print('-----')
31         print(receiver_email)
32         msg = MIME multipart()
33         msg['From'] = sender_email
34         msg['To'] = receiver_email
35         msg['Subject'] = subject
36         msg.attach(MIMEText(body, 'plain'))
37
38         try:
39             with open(pdf_path, "rb") as pdf_file:
40                 part = MIMEApplication(pdf_file.read(), Name=os.path.basename(pdf_path))
41                 # Agregar encabezados al adjunto
42                 part['Content-Disposition'] = 'attachment; filename="%s"' % os.path.basename(pdf_path)
43                 msg.attach(part)
44                 server.sendmail(sender_email, receiver_email, msg.as_string())
45                 print(f"Correo enviado a {receiver_email}")
46         except Exception as e:
47             print(f"No se pudo enviar el correo a {receiver_email}: {e}")
48
49     server.quit()
```

De la misma forma y como se muestra en la Figura 16, el proceso de adjuntar el documento desde la ruta especificada en el parámetro también está dentro de una estructura Try Except, con la finalidad de poder recuperar el error en caso el script no pueda acceder a la ruta especificada o ésta no exista.

e. Generación de tablas resumen

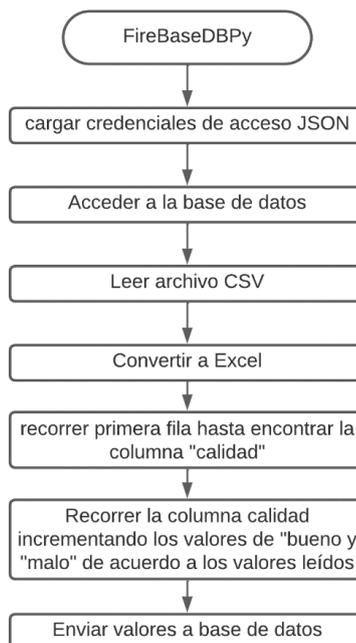
La generación del resumen de la información generada por el procesamiento de imágenes se obtuvo directamente del archivo CSV que fue exportado por el script principal; con lo cual, ha sido necesario acceder al archivo, leer las columnas y filas correspondientes para poder obtener las cantidades de semillas en estado “bueno” y de la misma manera con las de estado “malo” y esta información se ha enviado a la base datos en la nube alojada en Firebase mostrada en la Figura 17

Figura 17
Base datos implementada en Firebase.



Para lograr esto, se siguieron los pasos mostrados en el diagrama de flujo de la Figura 18

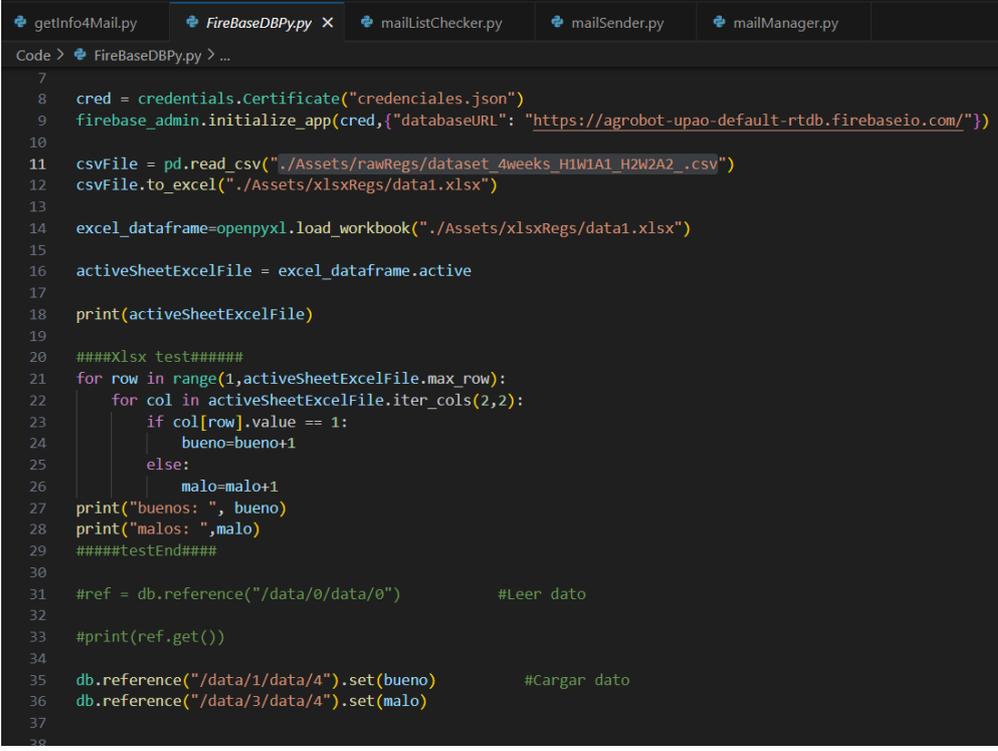
Figura 18
Diagrama de flujo de script de gestión de base de datos.



La implementación de este diagrama de bloques se da en el script `FireBaseDBPy.py` mostrado en la Figura 19

Figura 19

Implementación de script de gestión de base de datos en Python.



```
Code > FireBaseDBPy.py > ...
7
8 cred = credentials.Certificate("credenciales.json")
9 firebase_admin.initialize_app(cred, {"databaseURL": "https://agrobot-upao-default-rtdb.firebaseio.com/"})
10
11 csvFile = pd.read_csv("./Assets/rawRegs/dataset_4weeks_H1W1A1_H2W2A2 .csv")
12 csvFile.to_excel("./Assets/xlsxRegs/data1.xlsx")
13
14 excel_dataframe=openpyxl.load_workbook("./Assets/xlsxRegs/data1.xlsx")
15
16 activeSheetExcelFile = excel_dataframe.active
17
18 print(activeSheetExcelFile)
19
20 #####Xlsx test#####
21 for row in range(1,activeSheetExcelFile.max_row):
22     for col in activeSheetExcelFile.iter_cols(2,2):
23         if col[row].value == 1:
24             bueno=bueno+1
25         else:
26             malo=malo+1
27 print("buenos: ", bueno)
28 print("malos: ",malo)
29 #####testEnd###
30
31 #ref = db.reference("/data/0/data/0") #Leer dato
32
33 #print(ref.get())
34
35 db.reference("/data/1/data/4").set(bueno) #Cargar dato
36 db.reference("/data/3/data/4").set(malo)
37
38
```

Como se aprecia en las líneas 8 y 9, para acceder a la base de datos desde el script, es necesario usar las credenciales JSON, por lo cual, es necesario especificar la ruta en la que estas están almacenadas, para posteriormente inicializar el puente de comunicación con Firebase.

Otra ruta importante que especificar es la del archivo CSV a leer para poder obtener la data a resumir, para este fin se realizó una conversión a formato Excel, con el objetivo de poder hacer un recorrido de la información por filas y columnas, por medio del cual se logre incrementar las variables “bueno” y “malo”, las cuales hacen referencia a las calidades de la semilla.

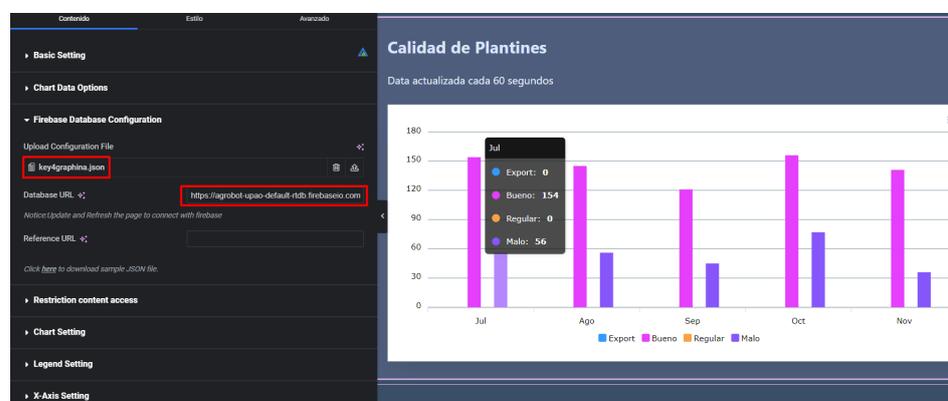
Una vez terminado el recorrido de la tabla, los valores de estas variables son enviadas a la base de datos (líneas 14 a 36).

f. Generación de gráficas de tablas resumen

Considerando que la data a representar en las gráficas se encuentra alojada en la nube, es posible delegar la tarea de generarlas al cliente web y en este caso y para ese fin, se usó la herramienta Graphina Charts, la cual puede vincularse a Firebase y generar gráficas a partir de la data referenciada, para ello se generaron las credenciales de acceso en formato JSON y se cargaron en Graphina al igual que el enlace único a la base de datos, como se muestra en la Figura 20.

Figura 20.

Enlace de cliente Web con base de datos.



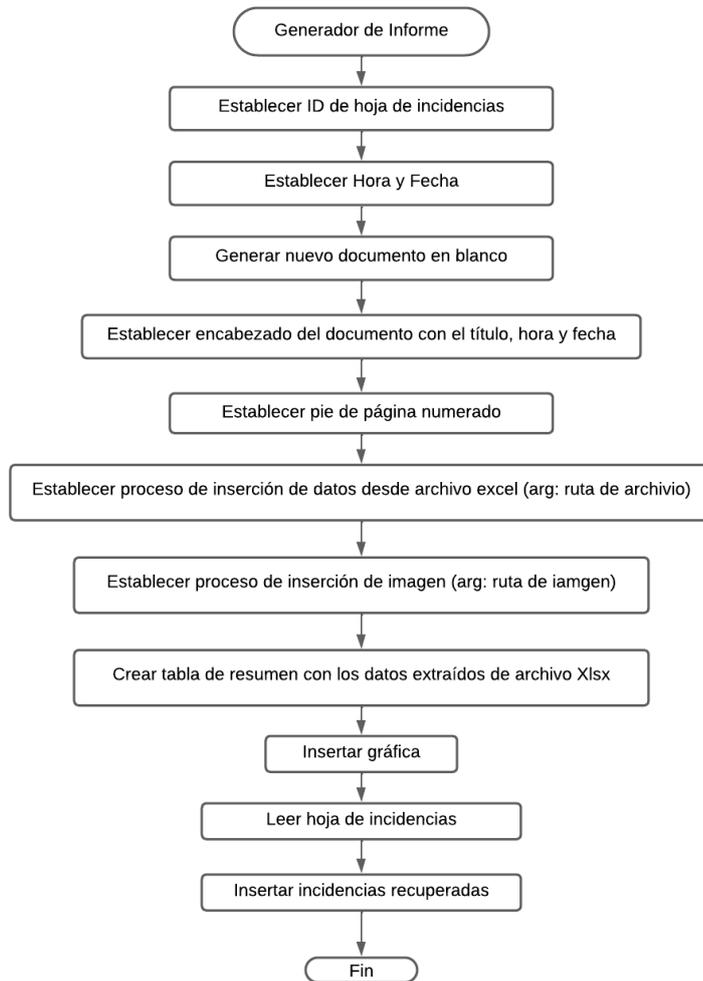
A su vez, el elemento generador de gráficas está alojado en el cliente web desplegado en el dominio <https://agrobot.website/>, a través del gestor web Elementor, permitiendo que esta data sea accesible desde cualquier navegador.

g. Redacción de informe diario con incidencias

La redacción del informe diario de producción, de forma adicional a la data generada por el sistema, también incluye un listado de incidencias registradas al momento en el que ocurren en la hoja de cálculo correspondiente, también compartida con la cuenta de servicio maillistchecker@agrobot-upao.iam.gserviceaccount.com, lo que permite que sean accesibles por los scripts, de esta manera, el informe diario es generado siguiendo los pasos detallados en el diagrama de flujo de la Figura 21

Figura 21

Diagrama de flujo de script generador de informe diario de producción.



En este orden de ideas, se procedió con la implementación del script `infGenerator.py`, mostrado en la Figura 22

Figura 22

Implementación de script generador de informe diario de producción en Python líneas 6 a 38.

```
getInfo4Mail.py  infGenerator.py X  mailListChecker.py  mailSender.py  mailManager.py
Code > infGenerator.py > ...
5
6  spreadsheet_id = '1q1gMa3gA1We3vzLzGsJtYX5KkfiHYkdPpsgB1xnMEKU'
7  range_name = 'A1:B8'
8  fecha_de_informe = datetime.now().strftime("%Y-%m-%d %H:%M:%S")
9
10 class PDF(FPDF):
11     def header(self):
12         # Encabezado con fecha
13         self.set_font('Arial', 'B', 12)
14         self.cell(0, 10, f'Informe de Producción Agrobot {fecha_de_informe}', 0, 1, 'C')
15
16     def footer(self):
17         # Pie de página con número de página
18         self.set_y(-15)
19         self.set_font('Arial', 'I', 8)
20         self.cell(0, 10, f'Página {self.page_no()}', 0, 0, 'C')
21
22     def agregar_tabla(pdf, path_excel):
23         # Leer el archivo Excel
24         df = pd.read_excel(path_excel, engine='openpyxl')
25         pdf.set_font("Arial", size=6)
26         # Añadir las filas de la tabla
27         for i in range(len(df)):
28             for col in df.columns:
29                 pdf.cell(23, 7, str(df[col].iloc[i]), border=1)
30             pdf.ln(7)
31
32     def agregar_imagen(pdf, path_imagen):
33         # Añadir imagen
34         pdf.image(path_imagen, x=10, y=pdf.get_y(), w=180, h=80)
35
36 # Crear el PDF
37 pdf = PDF()
38 pdf.add_page()
```

De acuerdo con la Figura 22, después de establecer los parámetros fijos del documento en las líneas de la 6 a la 8, se procede a crear las funciones recurrentes en el documento, de la línea 11 a la 20, se establecen el encabezado y el pie de página; de la 22 a la 30 se extrae la data del archivo Excel especificado en “path_excel”, de esta forma se puede pasar al documento desde un conjunto de totales a la totalidad de la tabla; continuando con la función de agregar imagen desde una ruta “path_imagen” a partir de la línea 32, para finalmente, agregar la primera hoja del documento en la línea 38.

Figura 23

Implementación de script generador de informe diario de producción en Python líneas 39 a 56.

```
38 pdf.add_page()
39
40 # Agregar la tabla
41 path_excel = 'Code/Assets/xlsxRegs/data1.xlsx' # ruta
42 agregar_tabla(pdf, path_excel)
43
44 # Añadir página para continuar después de los datos de excel
45 pdf.add_page()
46 pdf.ln(20)
47
48 # Título de la imagen
49 pdf.set_font("Arial", 'B', size=24)
50 pdf.cell(0, 10, 'Gráfica Resumen', 0, 1, 'C')
51 pdf.ln(10)
52
53 # Agregar la imagen
54 path_imagen = 'C:/Users/jmend/OneDrive/Documents/TesisMasterJM2023/captur
55 agregar_imagen(pdf, path_imagen)
56 pdf.ln(100)
```

Dado que el script ya tiene las funciones definidas, en la Figura 22, se aprecia la implementación de estas ya con las rutas reales tanto de los datos para la tabla como los de la imagen del gráfico, dando como resultado el encabezado previo a la tabla que se muestra en la Figura 24.

Figura 24

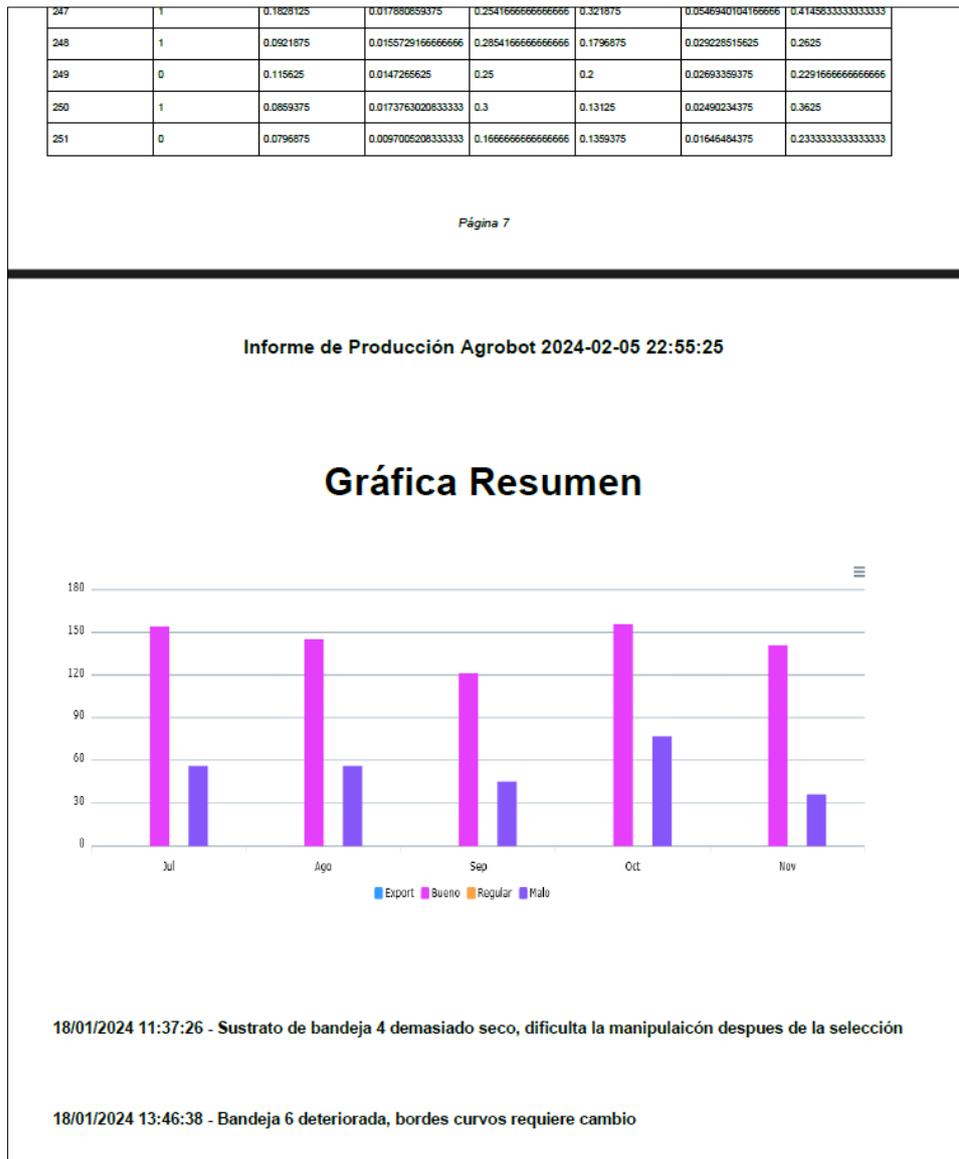
Cabecera de informe diario de producción generado por script.

Informe de Producción Agrobot 2024-02-05 22:55:25							
0	0	0.1359375	0.01884765625	0.3041666666666666	0.1265625	0.0191471354166666	0.2520833333333333
1	0	0.1640625	0.01943359375	0.2791666666666667	0.2359375	0.0468880208333333	0.3895833333333333
2	1	0.140625	0.01900390625	0.2583333333333333	0.24375	0.0558040364583333	0.5020833333333333
3	0	0.15	0.013994140625	0.19375	0.2359375	0.02416015625	0.2041666666666666
4	0	0.1603375	0.0186881510416666	0.28125	0.1140625	0.0200748697916666	0.3458333333333333
5	0	0.153125	0.019052734375	0.2625	0.23125	0.0409733072916666	0.3125
6	0	0.1296875	0.015168015625	0.2375	0.1515625	0.028671875	0.3729166666666667

De igual manera la imagen de la gráfica de calidad en el tiempo posterior a la tabla con los datos tal y como se muestra en la Figura 25

Figura 25

Gráfica recuperada de cliente web insertada en informe.



Nota. En la parte inferior de la imagen se muestran las incidencias obtenidas por el script desde la hoja de cálculo compartida.

La Figura 25 también muestra una lista de incidencias obtenidas de la hoja compartida a través de la cuenta de servicio, de la misma forma en la que se obtienen los correos para los destinatarios del reporte, con lo cual, es necesario que este archivo también esté compartido con la cuenta de servicio.

Figura 26

Implementación de script generador de informe diario de producción en Python líneas 54 a 75.



```
Code > infGenerator.py > ...
52
53 # Agregar la imagen
54 path_imagen = 'C:/Users/jmend/OneDrive/Documents/TesisMasterJM2023/captura_elementos.png'
55 agregar_imagen(pdf, path_imagen)
56 pdf.ln(100)
57
58 # Título de Incidencias
59 pdf.set_font("Arial", 'B', size=24)
60 pdf.cell(0, 10, 'Incidencias', 0, 1, 'C')
61 pdf.ln(10)
62
63 #Leer tabla de incidencias
64 values = read_sheet(spreadsheet_id, range_name)
65
66 # Escribir incidencias
67 for value in values:
68     pdf.set_font("Arial", 'B', size=8)
69     pdf.cell(0, 10, value[0]+" - "+value[1], 0, 1, 'L')
70
71 # Guardar el PDF
72 fecha_para_path=fecha_de_informe.replace(":", "_")
73 nombre_pdf = f'informe_produccion_{fecha_para_path}.pdf'
74 pdf.output(nombre_pdf)
75 print(f"PDF generado: {nombre_pdf}")
76
```

Como se aprecia en la Figura 26, la acción de integrar las incidencias al informe se realiza con la misma función con la que se lee la lista de correos, la función “read_sheet” que recibe como parámetro el ID único de la hoja de incidencias y las agrega una a una a través de un bucle for mostrado en las líneas 67 a la 69 para posteriormente guardar el archivo con el nombre Informe de producción y la fecha y hora correspondiente.

h. Verificación por parte del operador de la lista de direcciones de correos de los interesados

Para esta acción se hace uso del script mailListChecker.py de la misma forma que se realiza para enviar el reporte

i. Redacción y envío de correo electrónico con el informe diario de producción

Para esta acción se hace uso de mailManager.py con la diferencia de que para esta actividad se usa como archivo adjunto el informe diario de producción.

4.3.3. Pruebas de operatividad

Una vez desarrollados los algoritmos que automatizan las tareas del operador, es necesario desplegar los scripts en el entorno de trabajo del PE501079170-2022-PROCIENCIA, siendo el más crítico el CSV_Report_Builder, ya que es el único que requiere correr dentro del algoritmo principal como una función más del sistema, encargándose de extraer la data al archivo CSV del cual se alimentarán todos los otros scripts que pueden desenvolverse de manera externa.

Posterior a la integración de CSV_Report_Builder, se desplegaron los scripts independientes, logrando automatizar las tareas del operador.

4.3.4. Corrección de errores y despliegue de módulo funcional

Tras la implementación de los scripts independientes se detectó una falta de comunicación de estos con el cliente web que contiene las gráficas obtenidas de Firebase, por lo que se realizó un script adicional encargado únicamente de extraer la gráfica actualizada de la web; lo que permitió la implementación del script getInfo4mail.py mostrado en la Figura 27.

Figura 27

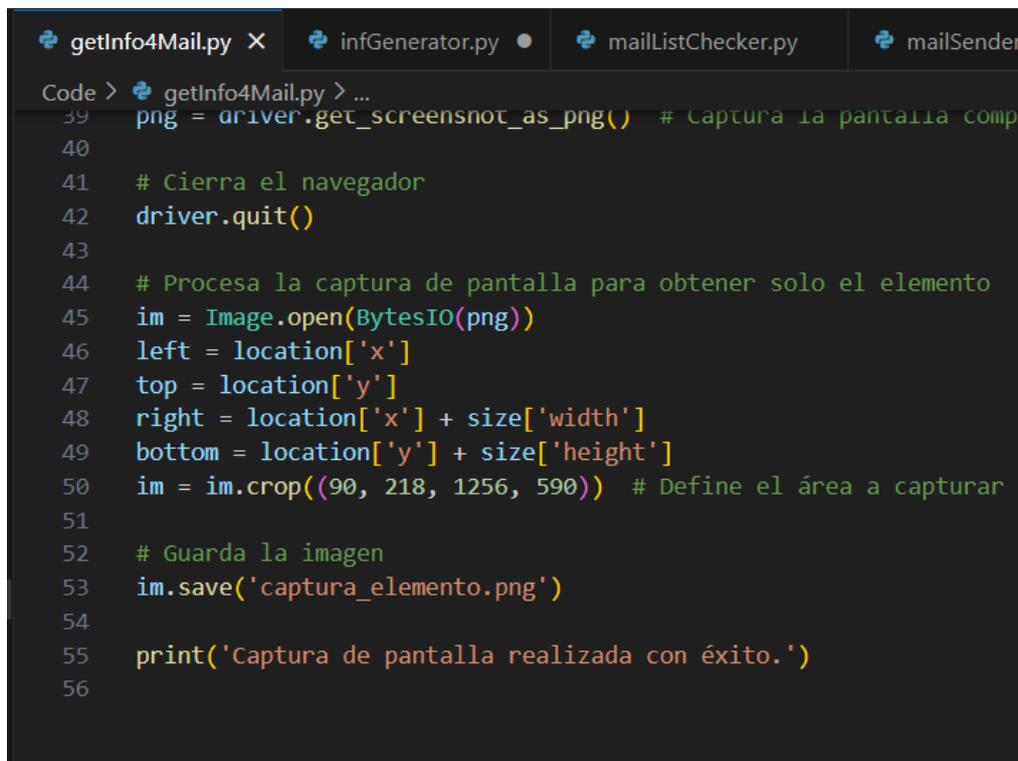
Implementación de script de extracción de datos de cliente web en Python líneas 3 a 41.

```
getInfo4Mail.py ● infGenerator.py mailListChecker.py mailSender.py
Code > getInfo4Mail.py > ...
3 from selenium import webdriver
4 from selenium.webdriver.common.by import By
5 from PIL import Image
6 from io import BytesIO
7
8 # Establecer la ruta al GeckoDriver en la variable de entorno PATH
9 geckodriver_path = 'C:/Users/jmend/Downloads/geckodriver-v0.34.0-
10 os.environ['PATH'] += os.pathsep + geckodriver_path
11
12 # Configuración de Selenium para usar Firefox
13 firefox_options = webdriver.FirefoxOptions()
14 firefox_options.add_argument('--headless')
15 driver = webdriver.Firefox(options=firefox_options)
16
17 # Ajustar el tamaño de la ventana del navegador
18 driver.set_window_size(1366, 768)
19 # URL de la página web
20 url = 'https://agrobot.website/'
21
22 # Abre la página web
23 driver.get(url)
24
25 # Hacer scroll por una cantidad específica de píxeles
26 scroll_height = 2000
27 driver.execute_script(f"window.scrollTo(0, {scroll_height});")
28
29 # Espera para que el scroll se complete
30 time.sleep(2.5)
31
32 # buscar el elemento por su clase
33 element = driver.find_element(By.CLASS_NAME, 'chart-box')
34
35 # Obtener la ubicación y el tamaño del elemento para definir el área
36 location = element.location
37 size = element.size
38 png = driver.get_screenshot_as_png() # Captura la pantalla completa
39
40 # Cierra el navegador
41 driver.quit()
```

Este script establece una instancia de navegador a través de la cual, accede a la web <https://agrobot.website/> y busca la gráfica por su id de elemento y lo captura para finalmente procesar y guardar la imagen como se muestra en la Figura 28.

Figura 28

Implementación de script de extracción de datos de cliente web en Python líneas 41 a 55.



```
Code > getInfo4Mail.py X infGenerator.py mailListChecker.py mailSender
39 png = driver.get_screenshot_as_png() # Captura la pantalla completa
40
41 # Cierra el navegador
42 driver.quit()
43
44 # Procesa la captura de pantalla para obtener solo el elemento
45 im = Image.open(BytesIO(png))
46 left = location['x']
47 top = location['y']
48 right = location['x'] + size['width']
49 bottom = location['y'] + size['height']
50 im = im.crop((left, top, right, bottom)) # Define el área a capturar
51
52 # Guarda la imagen
53 im.save('captura_elemento.png')
54
55 print('Captura de pantalla realizada con éxito.')
56
```

De esta forma, se obtiene la gráfica generada en el cliente web de manera automática completando la automatización del proceso de monitoreo de producción del PE501079170-2022-PROCIENCIA.

4.4. Aplicación de instrumento en etapa post-test

A los procedimientos previamente descritos, se les aplicó la guía de observación de tiempos, los valores de tiempo fueron obtenidos a través de la función `time.perf_counter()`

4.4.1. Medición de tiempos de generación de reporte de operación

De acuerdo con el proceso analizado y las tareas que lo conforman, se ha generado su automatismo correspondiente, detallado en la Tabla 7.

Tabla 7

Correspondencia de tareas y scripts de automatización en generación de reporte de producción.

	<i>Tarea</i>	<i>Script de automatismo</i>
1	Ejecución de algoritmo de reconocimiento	N
2	Extracción de data cruda de la terminal	csv_report_builder()
3	Procesamiento de datos en hoja de cálculo	csv_report_builder()

De tal manera que, aplicando la guía de observación de tiempos a 8 lotes de 6 bandejas cada uno, se obtienen los tiempos comprendidos en la Tabla 8.

Tabla 8

Tiempos por subproceso de la generación de reporte de producción post-test.

LOTE	TIEMPO POR SUBPROCESO	B1	B2	B3	B4	B5	B6	SUBTOTAL
L1	tiempo algoritmo (S)	0.055501	0.048786	0.055011	0.049131	0.054665	0.055225	0.31832
	tiempo extracción (S)	0.012382	0.012954	0.013150	0.012740	0.013034	0.012544	0.07680
	tiempo procesamiento (S)	6.530670	6.435553	6.115416	6.279783	6.174242	6.376272	37.91194
L2	tiempo algoritmo (S)	0.048555	0.053490	0.050332	0.054335	0.054920	0.051526	0.31316
	tiempo extracción (S)	0.012633	0.012876	0.012573	0.012958	0.012550	0.012636	0.07623
	tiempo procesamiento (S)	6.464036	6.508887	6.418954	6.530909	6.347128	6.547925	38.81784
L3	tiempo algoritmo (S)	0.049231	0.053791	0.055898	0.054355	0.048278	0.054530	0.31608
	tiempo extracción (S)	0.012422	0.012885	0.012760	0.012907	0.012510	0.013074	0.07656
	tiempo procesamiento (S)	6.395666	6.397785	6.590109	6.455430	6.174035	6.369876	38.38290
L4	tiempo algoritmo (S)	0.053063	0.051863	0.051884	0.051341	0.055854	0.048157	0.31216
	tiempo extracción (S)	0.013047	0.012880	0.012843	0.012407	0.012839	0.012568	0.07658
	tiempo procesamiento (S)	6.168602	6.434335	6.163041	6.501352	6.577771	6.427253	38.27235
L5	tiempo algoritmo (S)	0.055365	0.051593	0.053459	0.050764	0.055852	0.054825	0.32186
	tiempo extracción (S)	0.013236	0.012783	0.013280	0.012346	0.012417	0.012561	0.07662
	tiempo procesamiento (S)	6.512166	6.385460	6.587284	6.603398	6.569003	6.233302	38.89061
L6	tiempo algoritmo (S)	0.052402	0.053078	0.049867	0.051338	0.049036	0.050070	0.30579
	tiempo extracción (S)	0.012430	0.013256	0.012628	0.012684	0.013276	0.012595	0.07687
	tiempo procesamiento (S)	6.566772	6.278668	6.558603	6.439602	6.291041	6.320283	38.45497
L7	tiempo algoritmo (S)	0.051197	0.049936	0.052657	0.049387	0.048332	0.055927	0.30744
	tiempo extracción (S)	0.012656	0.012717	0.013089	0.013299	0.012974	0.013088	0.07782
	tiempo procesamiento (S)	6.317497	6.617494	6.540576	6.611853	6.483107	6.319443	38.88997
L8	tiempo algoritmo (S)	0.051380	0.054961	0.055643	0.053143	0.053144	0.048979	0.31725
	tiempo extracción (S)	0.013196	0.012952	0.012409	0.012368	0.012734	0.013160	0.07682
	tiempo procesamiento (S)	6.264722	6.238251	6.419731	6.202526	6.602485	6.616938	38.34465
TOTAL								311.09160

De acuerdo con los datos detallados en la Tablas 8, el tiempo de procesamiento no muestra una variación importante con respecto de la etapa pre-test ya que el script principal, encargado de procesar las imágenes, no se ha alterado y la tarea que consume más tiempo es el procesamiento de los datos en el archivo de tipo tabla, ya que se basa en operaciones de lectura escritura en el disco duro; de esta forma se ha obtenido un tiempo total de 311.09160 segundos, lo que equivale aproximadamente a 5.18 minutos o 0.09 horas.

4.4.2. Medición de tiempos de difusión de reporte de operación

De acuerdo con el proceso analizado y las tareas que lo conforman, se ha generado su automatismo correspondiente, detallado en la Tabla 9.

Tabla 9

Correspondencia de tareas y scripts de automatización en difusión de reporte de producción.

	TAREA	SCRIPT DE AUTOMATISMO
1	Verificación por parte del operador de la lista de direcciones de correos de los interesados	mailListChecker.py
2	Redacción y envío de correo electrónico con el reporte de producción	mailSender.py + mailManager.py

De tal manera que, aplicando la guía de observación de tiempos a 8 lotes de 6 bandejas cada uno, se obtienen los tiempos comprendidos en la Tabla 10.

Tabla 10*Tiempos por subproceso de la difusión de reporte de producción post-test.*

LOTE	TIEMPO POR SUBPROCESO	B1	B2	B3	B4	B5	B6	SUBTOTAL
L1	tiempo de verificación de lista de correos	2.66954515	2.92260269	2.86263414	2.62515673	2.48847336	3.10761217	16.67602424
	tiempo de redacción y envío de correo	9.09454231	8.58505749	9.75773631	8.7253101	9.40604157	9.89132662	55.46001439
L2	tiempo de verificación de lista de correos	2.83256734	3.02074225	2.49292666	2.52139014	2.67024482	2.49081524	16.02868645
	tiempo de redacción y envío de correo	8.44175291	9.97607144	9.52807867	9.01543239	9.69518115	9.50303762	56.15955418
L3	tiempo de verificación de lista de correos	3.06591588	2.85526533	3.07003969	2.7168072	3.04387693	2.57157372	17.32347874
	tiempo de redacción y envío de correo	8.93319443	9.38557352	9.83147991	8.63398496	8.78379302	9.93615511	55.50418095
L4	tiempo de verificación de lista de correos	2.56380585	2.70188684	2.93455131	2.78709641	2.54540682	3.0546557	16.58740293
	tiempo de redacción y envío de correo	8.18786428	9.50198769	9.91466836	9.76454747	9.62396175	8.91563171	55.90866127
L5	tiempo de verificación de lista de correos	2.72039163	2.81401508	2.94234948	2.46067433	3.08372493	2.59070031	16.61185576
	tiempo de redacción y envío de correo	9.66808875	9.07750351	9.7196871	9.35065336	9.32842553	9.94281769	57.08717594
L6	tiempo de verificación de lista de correos	2.63872762	3.08763486	2.47350746	2.57948555	3.08290798	2.45162318	16.31388664
	tiempo de redacción y envío de correo	8.53944512	8.97249507	9.46319238	9.53943718	9.8531458	9.92336031	56.29107587
L7	tiempo de verificación de lista de correos	2.8565974	2.60026214	2.98142914	2.47902817	2.61476506	2.49344935	16.02553126
	tiempo de redacción y envío de correo	8.24551129	9.60056979	9.60035151	8.48555323	8.8342077	9.92156367	54.6877572
L8	tiempo de verificación de lista de correos	2.64417699	3.01293108	2.61775064	2.96348865	2.56814035	2.8407152	16.64720291
	tiempo de redacción y envío de correo	8.21401888	8.59107725	9.34061287	8.44154209	8.90580795	8.73262531	52.22568434
TOTAL								575.5381731

De acuerdo con los datos detallados en la Tabla 10, los tiempos presentan variación con respecto a la etapa pre-test y la tarea que más tiempo consume es la redacción y envío de correo, ya que el script debe acceder a la cuenta de envío de correo y esa acción es dependiente de la velocidad de conexión a internet; de esta forma se ha obtenido un tiempo total de 575.5381731 segundos, lo que equivale aproximadamente a 9.59 minutos o 0.16 horas.

4.4.3. Medición de tiempos de generación-difusión de informe diario de operación

De acuerdo con el proceso analizado y las tareas que lo conforman, se ha generado su automatismo correspondiente, detallado en la Tabla 11.

Tabla 11

Correspondencia de tareas y scripts de automatización en generación-difusión de informe diario de producción.

	TAREA	SCRIPT DE AUTOMATISMO
1	Generación de tablas resumen	FireBaseDBPy.py
2	Generación de gráficas de tablas resumen	FireBaseDBPy.py + Graphina Charts
3	Redacción de informe con incidencias	infGenerator.py
4	Verificación por parte del operador de la lista de direcciones de correos de los interesados	mailListChecker.py
5	Redacción y envío de correo electrónico con el informe diario de producción	mailSender.py + mailManager.py

De tal manera que, aplicando la guía de observación de tiempos a 8 lotes de 6 bandejas cada uno, se obtienen los tiempos comprendidos en la Tabla 11.

Tabla 12

Tiempos por subproceso de la generación- difusión de informe diario de producción post-test.

LOTE	TIEMPOS POR ACTIVIDAD	DURACIÓN
L1	Generación de tablas resumen (S)	5.008913092
	Generación de gráficas de tablas resumen (S)	3.534015143
	Redacción de informe con incidencias (S)	10.26460146
	Verificación de lista de correos de los interesados (S)	2.524483923
	Redacción y envío de correo con el informe diario (S)	9.84336463
L2	Generación de tablas resumen con gráficas (S)	4.834568576
	Generación de tablas resumen (S)	3.511757578
	Redacción de informe con incidencias (S)	9.458191157
	Verificación de lista de correos de los interesados (S)	3.076690345
	Redacción y envío de correo con el informe diario (S)	9.777600836
L3	Generación de tablas resumen con gráficas (S)	4.414901296
	Generación de tablas resumen (S)	3.484359771
	Redacción de informe con incidencias (S)	9.77879766
	Verificación de lista de correos de los interesados (S)	2.916029432
	Redacción y envío de correo con el informe diario (S)	9.399880476
L4	Generación de tablas resumen con gráficas (S)	4.570756089
	Generación de tablas resumen (S)	3.889496575
	Redacción de informe con incidencias (S)	10.06221896
	Verificación de lista de correos de los interesados (S)	2.741259709
	Redacción y envío de correo con el informe diario (S)	8.732432292
L5	Generación de tablas resumen con gráficas (S)	5.077281176
	Generación de tablas resumen (S)	3.699448704
	Redacción de informe con incidencias (S)	11.04643035
	Verificación de lista de correos de los interesados (S)	3.086870802
	Redacción y envío de correo con el informe diario (S)	9.532420933
L6	Generación de tablas resumen con gráficas (S)	4.870812705
	Generación de tablas resumen (S)	3.589201601
	Redacción de informe con incidencias (S)	9.565906476
	Verificación de lista de correos de los interesados (S)	3.075895032
	Redacción y envío de correo con el informe diario (S)	8.649917341
L7	Generación de tablas resumen con gráficas (S)	4.365734703
	Generación de tablas resumen (S)	4.019041458
	Redacción de informe con incidencias (S)	9.576685774
	Verificación de lista de correos de los interesados (S)	3.090924979
	Redacción y envío de correo con el informe diario (S)	9.39496687
L8	Generación de tablas resumen con gráficas (S)	4.445389059
	Generación de tablas resumen (S)	3.81727479
	Redacción de informe con incidencias (S)	9.958380441
	Verificación de lista de correos de los interesados (S)	2.989658469
	Redacción y envío de correo con el informe diario (S)	9.578940297
TOTAL		245.255501

De acuerdo con los datos detallados en la Tabla 11, los tiempos presentan variación con respecto a la etapa pre-test, aunque la de redacción de informe se mantiene como la tarea que más tiempo consume ya que para obtener los datos necesarios para el informe se debe acceder a data almacenada en el ordenador y descargar la gráfica generada en el cliente web, ambas tareas dependen de la velocidad del disco duro y de la conectividad a internet respectivamente; de esta forma se ha obtenido un tiempo total de 245.255501 segundos, lo que equivale aproximadamente a 4.09 minutos o 0.07 horas.

4.4.4. Resumen de tiempos totales

En concordancia con la data mostrada en las Tablas 8, 10 y 12 se puede extraer los tiempos más relevantes en la Tabla 13 con la finalidad de resumir y dar una vista general de los tiempos consumidos por actividad automatizada en la etapa post-test del proceso de monitoreo de producción del sistema robótico PE501079170-2022-PROCIENCIA.

Tabla 13

Resumen de tiempos de subprocesos de monitoreo de producción de PE501079170-2022-PROCIENCIA Post-test.

<i>LOTE</i>	<i>TIEMPO POR ACTIVIDAD</i>	<i>DURACIÓN (S)</i>	<i>DURACIÓN TOTAL POR LOTE (S)</i>
<i>L1</i>	generación de reporte de operación (S)	38.30706	141.62
	difusión de reporte de operación (S)	72.13604	
	generación-difusión de informe diario de operación (S)	31.17538	
<i>L2</i>	generación de reporte de operación (S)	39.20722	142.05
	difusión de reporte de operación (S)	72.18824	
	generación-difusión de informe diario de operación (S)	30.65881	
<i>L3</i>	generación de reporte de operación (S)	38.77554	141.60
	difusión de reporte de operación (S)	72.82766	
	generación-difusión de informe diario de operación (S)	29.99397	
<i>L4</i>	generación de reporte de operación (S)	38.66110	141.15
	difusión de reporte de operación (S)	72.49606	
	generación-difusión de informe diario de operación (S)	29.99616	
<i>L5</i>	generación de reporte de operación (S)	39.28910	145.43
	difusión de reporte de operación (S)	73.69903	
	generación-difusión de informe diario de operación (S)	32.44245	
<i>L6</i>	generación de reporte de operación (S)	38.83763	141.19
	difusión de reporte de operación (S)	72.60496	
	generación-difusión de informe diario de operación (S)	29.75173	
<i>L7</i>	generación de reporte de operación (S)	39.27523	140.44
	difusión de reporte de operación (S)	70.71329	
	generación-difusión de informe diario de operación (S)	30.44735	
<i>L8</i>	generación de reporte de operación (S)	38.73872	138.40
	difusión de reporte de operación (S)	68.87289	
	generación-difusión de informe diario de operación (S)	30.78964	
TOTAL			1131.89

De acuerdo con la Tabla 13 existe variación con respecto de la etapa pre-test, en donde, ahora la tarea de mayor duración es difusión del reporte de operación seguida por la generación de reporte de operación y, por último, con una duración menor, la generación-difusión de informe diario de operación.

4.5. Procesamiento y análisis de datos

4.5.1. Procesamiento de datos

En pro de denotar las variaciones entre los datos correspondientes a la etapa pre-test y la etapa post-test, se presenta en la Tabla 14 la duración promedio de cada actividad.

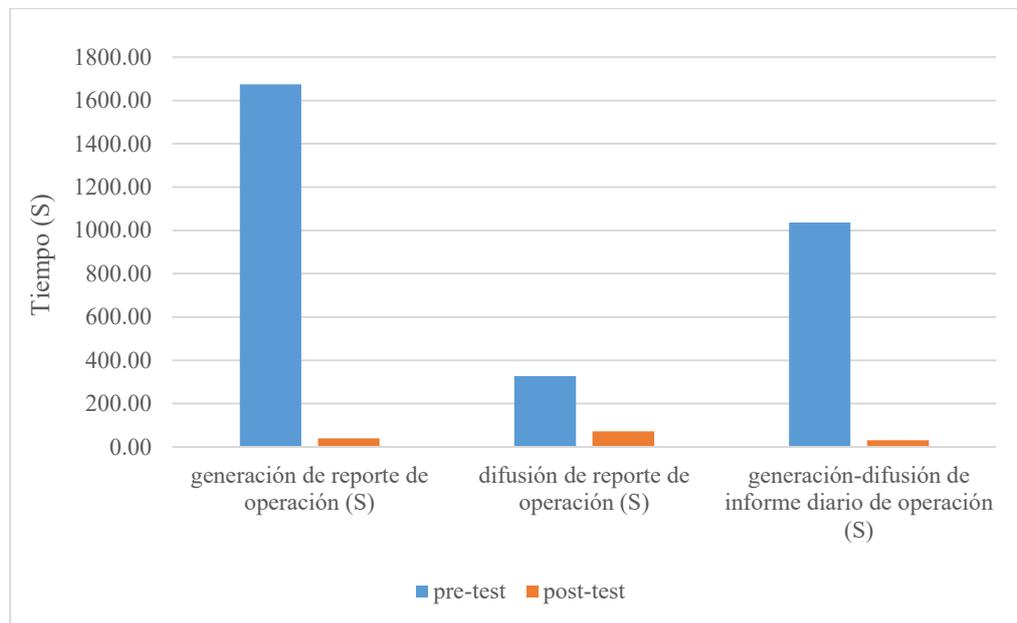
Tabla 14
Promedio de tiempos por subproceso pre y post-test.

SUBPROCESO	ETAPA	L1	L2	L3	L4	L5	L6	L7	L8	PROMEDIO
<i>generación de reporte de operación (S)</i>	pre-test	1678.27	1657.74	1643.01	1682.05	1706.03	1681.80	1693.94	1648.81	1673.96
	post-test	38.30706	39.20722	38.77554	38.66110	39.28910	38.83763	39.27523	38.73872	38.89
<i>difusión de reporte de operación (S)</i>	pre-test	328.03	322.86	317.35	337.90	329.07	328.25	318.40	335.62	327.19
	post-test	72.13604	72.18824	72.82766	72.49606	73.69903	72.60496	70.71329	68.87289	71.94
<i>generación-difusión de informe diario de operación (S)</i>	pre-test	1167.98	1138.41	1071.78	988.21	972.13	1016.78	849.10	1086.61	1036.38
	post-test	31.17538	30.65881	29.99397	29.99616	32.44245	29.75173	30.44735	30.78964	30.66

En la Figura 29, se aprecia la gráfica correspondiente a los promedios en cada actividad, pero la diferencia de promedios no es condición suficiente para establecer un impacto significativo.

Figura 29

Gráfica de barras de tiempos por subproceso pre y post-test.



Con la finalidad de evaluar si el impacto es o no, significativo para cada uno de los elementos del módulo de comunicación IoT-BI en sus actividades correspondientes, en contraste con los tiempos que se dan en el proceso sin usar este módulo, fue necesario comparar uno a uno los tiempos de cada subproceso tanto pre-test, como post-test; de acuerdo con (Montgomery, 2013) para esto se requiere realizar una prueba T o T-test, ya que la comparación necesaria implica contrastar los tiempos uno a uno, conformando una comparación por pares y dada la naturaleza de la actividad se estima que exista una distribución normal; mientras que, de acuerdo con (Kutner et al., 2004) y para confirmar la normalidad de los datos, se debe realizar una prueba, siendo así que para la cantidad y naturaleza de los datos que conforman la muestra, la más adecuada es la prueba Shapiro-Wilk ya que ésta es ideal para datos sobre los que ya se estima teóricamente que exista una distribución normal.

4.5.2. Análisis de normalidad de datos

En la Tabla 15 se muestran los resultados de la prueba de normalidad de Shapiro-Wilk, aplicada a cada uno de los subprocesos indicadores y al proceso de monitoreo tanto pre-test, como post-test.

Tabla 15

Resultados de prueba de normalidad Shapiro-Wilk a valores de tiempo de los indicadores pre y post-test.

	<i>Pre-test</i>		<i>Post-test</i>	
	Variable	Valor	Variable	Valor
Tiempo total del proceso de monitoreo de producción (S)	N	8	N	8
	Media	3038.00	Media	141.50
	Desviación Estándar	92.11	Desviación Estándar	1.9540
	Estadístico W	0.94	Estadístico W	0.8871
	p-valor	0.59	p-valor	0.2200
	Conclusión	No hay evidencia suficiente para decir que los datos no siguen una distribución normal.	Conclusión	No hay evidencia suficiente para decir que los datos no siguen una distribución normal.
Generación de reporte de operación (S)	N	8	N	8
	Media	1674.00	Media	38.89
	Desviación Estándar	22.13	Desviación Estándar	0.35
	Estadístico W	0.94	Estadístico W	0.91
	p-valor	0.65	p-valor	0.36
	Conclusión	No hay evidencia suficiente para afirmar que los datos no siguen una distribución normal.	Conclusión	No hay evidencia suficiente para decir que los datos no siguen una distribución normal.
Difusión de reporte de operación (S)	N	8	N	8
	Media	327.20	Media	71.94
	Desviación Estándar	7.41	Desviación Estándar	1.50
	Estadístico W	0.94	Estadístico W	0.87
	p-valor	0.59	p-valor	0.15
	Conclusión	No hay evidencia suficiente para decir que los datos no siguen una distribución normal.	Conclusión	No hay evidencia suficiente para decir que los datos no siguen una distribución normal.
Generación-difusión de informe diario de operación (S)	N	8	N	8
	Media	1036.00	Media	30.66
	Desviación Estándar	102.40	Desviación Estándar	0.86
	Estadístico W	0.96	Estadístico W	0.89
	p-valor	0.84	p-valor	0.23
	Conclusión	No hay evidencia suficiente para decir que los datos no siguen una distribución normal.	Conclusión	No hay evidencia suficiente para decir que los datos no siguen una distribución normal.

De acuerdo con (Kutner et al., 2004) la valoración de parámetros de la prueba de normalidad de Shapiro-Wilk se basa en el estadístico de prueba W y el valor P en donde W mide qué tan bien los datos se ajustan a una distribución normal, oscilando entre 0 y 1, donde un valor cercano a 1 indica que los datos se ajustan bien a una distribución normal; P indica la probabilidad de obtener un resultado al menos tan extremo como el observado, bajo la suposición de que la hipótesis nula es verdadera. En el contexto de la prueba de Shapiro-Wilk, la hipótesis nula

(H₀) afirma que los datos siguen una distribución normal, en este orden de ideas, los resultados se pueden interpretar de la siguiente manera:

Si el p-valor $< \alpha$ (donde α es el nivel de significancia, establecido en 0.05): Hay evidencia suficiente para rechazar la hipótesis nula; significando esto, que los datos no siguen una distribución normal. En la práctica, si el p-valor es menor que el nivel de significancia elegido se concluye que los datos no se ajustan a una distribución normal, y se debería considerar usar técnicas estadísticas no paramétricas o transformar los datos antes de realizar análisis adicionales.

Si el p-valor $> \alpha$: No hay evidencia suficiente para rechazar la hipótesis nula. Esto sugiere que los datos podrían ser normales y, por lo tanto, es razonable proceder con métodos estadísticos que asuman la normalidad de los datos.

4.5.3. Prueba de hipótesis

De acuerdo con los datos mostrados en la Tabla 14, se asume que los valores de tiempos de duración del proceso de monitoreo de producción y los subprocesos que lo conforman, tanto en la etapa pre-test, como post-test se ajustan a una distribución normal, con lo cual, se procedió con la prueba T para evaluar si existe impacto significativo de acuerdo con la hipótesis y objetivos.

De acuerdo con (Kutner et al., 2004), al realizar un t-test, se obtiene un p-valor, que indica la probabilidad de observar datos iguales a los actuales, hasta datos diferentes en extremo, en esta prueba, la decisión de aceptar o rechazar la hipótesis nula está basada en este p-valor, tal es así que:

Si el p-valor es menor que el nivel de significancia α (establecido en 0.05), entonces se rechaza la hipótesis nula (H₀). Esto sugiere que hay suficiente evidencia en los datos para apoyar la hipótesis alternativa (H₁), indicando que podría haber una diferencia significativa.

Si el p-valor es mayor que α , no se rechaza la hipótesis nula. Esto no necesariamente significa que H₀ es verdadera, sino que no hay suficiente evidencia en los datos para apoyar H₁.

4.5.3.1. Objetivo General

Determinar cómo influye la implementación de un módulo de conectividad IoT-BI en el monitoreo de la producción del sistema robótico Agrobot UPAO en la ciudad de Trujillo - 2024.

4.5.3.1.1. Comprobación de Hipótesis

H₁: La implementación de un módulo de comunicación IoT-BI influye significativamente en el monitoreo de la producción del sistema robótico Agrobot UPAO en la ciudad de Trujillo - 2024.

H₀: La implementación de un módulo de comunicación IoT-BI no influye significativamente en el monitoreo de la producción del sistema robótico Agrobot UPAO en la ciudad de Trujillo - 2024.

Tabla 16

Resultado de prueba T de los valores de tiempo total del proceso de monitoreo de PE501079170-2022-PROCIENCIA

<i>Grupo de datos</i>	<i>Media</i>	<i>Desviación Estándar</i>	<i>Estadístico T</i>	<i>Valor p</i>
<i>Pre-Test tiempo total en el proceso de monitoreo de producción</i>	3037.515	92.11232	-	-
<i>Post-Test tiempo total en el proceso de monitoreo de producción</i>	141.4857	1.953998	-	-
<i>T-test</i>	-	-	88.906	1.132e-20

La Tabla 16 muestra los resultados de la prueba T realizada a los tiempos del proceso de monitoreo de producción del sistema robótico PE501079170-2022-PROCIENCIA en etapa pre-test vs etapa post-test, el cual tiene como resultado un valor significancia P de 1.132e-20, siendo éste mucho menor a 0.05; con lo cual, es posible rechazar la hipótesis nula H₀, aceptando la hipótesis general H₁.

4.5.3.2. Objetivo Específico 1

Determinar cómo influye la implementación de un módulo de comunicación IoT-BI con base de datos NoSQL en la nube en el monitoreo de la producción del sistema robótico Agrobot UPAO en la ciudad de Trujillo – 2024.

4.5.3.2.1. Comprobación de Hipótesis Específica 1

H₁: La implementación de un módulo de comunicación IoT-BI con base de datos NoSQL en la nube influye significativamente en el monitoreo de la producción del sistema robótico Agrobot UPAO en la ciudad de Trujillo – 2024.

H₀: La implementación de un módulo de comunicación IoT-BI con base de datos NoSQL en la nube no influye significativamente en el monitoreo de la producción del sistema robótico Agrobot UPAO en la ciudad de Trujillo – 2024.

Tabla 17

Resultado de prueba T de los valores de tiempo del indicador "Generación de reporte de operación" PE501079170-2022-PROCIENCIA

<i>Dimensión</i>	<i>Indicador</i>	<i>Grupo de datos</i>	<i>Media</i>	<i>Desviación Estándar</i>	<i>Estadístico T</i>	<i>Valor p</i>
Tiempo necesario para la generación de reporte de producción	Horas/Hombre dedicadas a la generación del reporte de producción	Pre-Test subproceso de generación de reporte de operación	1673.956	22.12526	-	-
Comunicación con base de datos NoSQL almacenada en la nube	Horas/Máquina dedicadas a la generación del reporte de producción	Post-Test subproceso de generación de reporte de operación	38.88645	0.346236	-	-
		T-test	-	-	208.997	7.264e-26

La Tabla 17 muestra los resultados de la prueba T realizada a los tiempos del subproceso de generación de reporte de operación del sistema robótico PE501079170-2022-PROCIENCIA como indicadores de las dimensiones “Tiempo necesario para la generación

reporte de producción” y “Comunicación con base de datos NoSQL almacenada en la nube” en etapa pre-test vs etapa post-test respectivamente, el cual tiene como resultado un valor significancia P de $7.64e-26$, siendo éste mucho menor a 0.05; con lo cual, es posible rechazar la hipótesis nula H_0 , aceptando la hipótesis general H_1 .

4.5.3.3. Objetivo Específico 2

Determinar cómo influye la implementación de un módulo de comunicación IoT-BI con comunicación a una interfaz de usuario web frontend con gráficas BI, en el monitoreo de la producción del sistema robótico Agrobot UPAO en la ciudad de Trujillo - 2024

4.5.3.3.1. Comprobación de Hipótesis Específica 2

H_1 : La implementación de un módulo de comunicación IoT-BI con comunicación a una interfaz de usuario web frontend con gráficas BI influye significativamente en el monitoreo de la producción del sistema robótico Agrobot UPAO en la ciudad de Trujillo – 2024.

H_0 : La implementación de un módulo de comunicación IoT-BI con comunicación a una interfaz de usuario web frontend con gráficas BI no influye significativamente en el monitoreo de la producción del sistema robótico Agrobot UPAO en la ciudad de Trujillo – 2024.

Tabla 18

Resultado de prueba T aplicada a los valores de tiempo del indicador "Difusión de reporte de operación" PE501079170-2022-PROCIENCIA

<i>Dimensión</i>	<i>Indicador</i>	<i>Grupo</i>	<i>Media</i>	<i>Desviación Estándar</i>	<i>Estadístico T</i>	<i>Valor p</i>
Tiempo necesario para la difusión del reporte de producción	Horas/hombre dedicadas a la difusión del reporte de producción	Pre-Test subproceso de difusión de reporte de operación	327.185	7.406281	-	-
Comunicación interfaz de usuario Frontend BI	Horas/Máquina dedicadas a la difusión del reporte de producción	Post-Test subproceso de difusión de reporte de operación	71.94227	1.495015	-	-
		T-test	-	-	95.549	4.135e-21

La Tabla 18 muestra los resultados de la prueba T realizada a los tiempos del subproceso de difusión de reporte de operación del sistema robótico PE501079170-2022-PROCIENCIA como indicadores de las dimensiones “Tiempo necesario para la difusión del reporte de producción” y “Comunicación interfaz de usuario Frontend BI” en etapa pre-test vs etapa post-test respectivamente, el cual tiene como resultado un valor significancia P de 4.135e-21, siendo éste mucho menor a 0.05; con lo cual, es posible rechazar la hipótesis nula H_0 , aceptando la hipótesis general H_1 .

4.5.3.4. Objetivo Específico 3

Determinar cómo influye la implementación de un módulo de comunicación IoT-BI que puede generar y enviar por correo los informes diarios de producción, en el monitoreo de la producción del sistema robótico Agrobot UPAO en la ciudad de Trujillo – 2024.

4.5.3.4.1. Comprobación de Hipótesis Específica 3

H₁: La implementación de un módulo de comunicación IoT-BI que puede generar y enviar por correo los informes diarios de producción influye significativamente en el monitoreo de la producción del sistema robótico Agrobot UPAO en la ciudad de Trujillo – 2024.

H₀: La implementación de un módulo de comunicación IoT-BI que puede generar y enviar por correo los informes diarios de producción no influye significativamente en el monitoreo de la producción del sistema robótico Agrobot UPAO en la ciudad de Trujillo – 2024.

Tabla 19

Resultado de prueba T de los valores de tiempo del indicador "Generación-difusión del informe diario de operación" PE501079170-2022-PROCIENCIA

<i>Dimensión</i>	<i>Indicador</i>	<i>Grupo</i>	<i>Media</i>	<i>Desviación Estándar</i>	<i>Estadístico T</i>	<i>Valor p</i>
Tiempo necesario para la generación y difusión de informe diario de producción	Horas/hombre dedicadas a la generación y difusión de informe diario de producción	Pre-Test subproceso de generación-difusión de informe diario de operación	1036.375	102.4076	-	-
Generación y difusión de informe diario de producción	Horas/Máquina dedicadas a la generación y difusión de informe diario de producción	Post-Test subproceso de generación-difusión de informe diario de operación	30.65694	0.863045	-	-
		T-test	-	-	27.776	1.206e-13

La Tabla 19 muestra los resultados de la prueba T realizada a los tiempos del subproceso de generación-difusión de informe diario de producción del sistema robótico PE501079170-2022-PROCIENCIA como indicadores de las dimensiones “Tiempo de generación y difusión de

informe diario de producción” y “Generación y difusión de informe diario de producción” en etapa pre-test vs etapa post-test respectivamente, el cual tiene como resultado un valor significancia P de $1.206e-13$, siendo éste mucho menor a 0.05; con lo cual, es posible rechazar la hipótesis nula H_0 , aceptando la hipótesis general H_1 .

CAPÍTULO V

DISCUSIÓN

CAPÍTULO V: DISCUSIÓN

5.1. DISCUSIÓN

Teniendo en consideración en primer lugar a los microprocesos, la implementación del módulo de comunicaciones IoT-BI ha demostrado impactar significativamente en la optimización del microproceso de generación del reporte de producción dentro del sistema robótico de producción agrícola PE501079170-2022-PROCIENCIA. Este conjunto de subsistemas de automatización, implementados a través de scripts de Python, no solo ha agilizado la gestión de datos, sino que también ha permitido que el sistema se integre en tiempo real con una base de datos en la nube, permitiendo el acceso a estos desde cualquier aplicación conectada a internet. Este hallazgo concuerda con el estudio de (Shaukat et al., 2020), que subraya cómo la automatización en la gestión de información para BI se está convirtiendo en una práctica esencial en la agricultura 4.0, promoviendo eficiencias operativas notables. La reducción de tiempos en este microproceso no solo valida la hipótesis específica 1 que sostiene que “La implementación de un módulo de comunicación IoT-BI con base de datos NoSQL en la nube influye significativamente en el monitoreo de la producción en del sistema robótico Agrobot UPAO en la ciudad de Trujillo – 2024”, sino que también se alinea con tendencias observadas por Loyola Diaz (2022), quienes encontraron mejoras similares en sistemas agrícolas automatizados.

Considerando que el segundo microproceso evaluado en este estudio, la difusión automatizada del reporte de producción, lograda por el módulo IoT-BI, ha evidenciado una reducción significativa de tiempos. Este avance refleja la relevancia de incorporar automatismos a las tareas de gestión de información, tal como sugiere (Ahmad et al., 2020), dentro de los paradigmas de la industria 4.0 para la mejora continua de procesos productivos. Al automatizar la difusión de informes, no solo se incrementa la velocidad de comunicación sino también la accesibilidad de la información crítica para la toma de decisiones. Este impacto positivo en la eficiencia operativa encuentra eco en las observaciones de (Mendoza Vásquez, 2021), quien destacó cómo los automatismos con Python potencian las capacidades de las plataformas BI, subrayando la transformación digital en la gestión agrícola.

El tercer microproceso, concerniente a la generación y difusión del informe diario de producción, ha experimentado igualmente mejoras notorias en la etapa post-implementación del módulo de comunicación IoT-BI. Este avance es coherente con los planteamientos de (Muñoz Choque, 2021), quien afirmó que la reducción de tiempos mediante automatismos conduce directamente a una mayor productividad. La eficiencia mejorada en este microproceso subraya el potencial de las soluciones automatizadas para optimizar las operaciones diarias y facilitar una comunicación efectiva. Este resultado es una contribución significativa al cuerpo de conocimiento que respalda la integración de tecnologías de automatización para fortalecer los sistemas de producción agrícola, en línea con los hallazgos de (Arcentales Toledo & Slacedo Mena, 2023).

La acumulación de mejoras en los microprocesos ha tenido un impacto transformador en el macroproceso de monitoreo de la producción, evidenciando una optimización general de éste en el sistema robótico de producción agrícola PE501079170-2022-PROCIENCIA. La implementación del módulo IoT-BI no solo ha facilitado una reducción significativa en los tiempos de cada proceso involucrado, sino que también ha mejorado el flujo y la disponibilidad de la información de producción. Este impacto positivo en el macroproceso refleja una sinergia entre la automatización y la inteligencia de negocios, marcando un hito importante en la adopción de tecnologías emergentes para la mejora de la productividad agrícola. Los hallazgos de este estudio están en concordancia con las investigaciones previas de (Loyola Diaz, 2022) y (Mendoza Vásquez, 2021), quienes destacaron beneficios similares de la automatización en contextos agrícolas, reafirmando la pertinencia de los sistemas IoT-BI en la evolución hacia la agricultura inteligente.

CAPÍTULO VI

CONCLUSIONES Y RECOMENDACIONES

CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES

6.1. CONCLUSIONES

El estudio realizado para evaluar el impacto de la implementación de un módulo de comunicación IoT-BI en el sistema robótico Agrobot UPAO PE501079170-2022-PROCIENCIA presentó hallazgos significativos que no solo validan la hipótesis general propuesta, sino que también confirman las hipótesis específicas delineadas. A través del análisis y la aplicación metodológica de pruebas T para comparar los tiempos de duración de los procesos y subprocesos antes y después de la implementación, se ha demostrado una influencia significativa en la eficiencia dentro del proceso de monitoreo de la producción.

Respecto al Objetivo General, se ha determinado con claridad cómo la integración del módulo de conectividad IoT-BI influye significativamente a través de una reducción en los tiempos del proceso de monitoreo de la producción dentro del sistema robótico PE501079170-2022-PROCIENCIA; conformando de esta manera, evidencia empírica que conlleva a la corroborar la hipótesis general e implicando directamente que la tecnología IoT-BI ejerce una influencia significativa en la mejora de los sistemas de monitoreo de producción sistemas robóticos agrícolas.

En relación con los Objetivos Específicos, los resultados han mostrado evidencia contundente, de esta manera es posible sostener que la implementación del módulo de comunicación IoT-BI, utilizando una base de datos NoSQL en la nube, ha demostrado un impacto significativo en la agilización del flujo de información del monitoreo de producción, validando la primera hipótesis específica. Este hallazgo resalta la importancia de las tecnologías de base de datos avanzadas en la gestión eficaz de datos en tiempo real.

La transmisión de data IoT-BI hacia una interfaz de usuario web frontend con gráficas BI ha mejorado notablemente la interpretabilidad y accesibilidad de los datos de

producción. Este resultado, que confirma la segunda hipótesis específica, subraya el valor añadido de las interfaces de usuario intuitivas en el contexto del análisis BI.

En este orden de ideas, es posible denotar que la capacidad del módulo IoT-BI para generar y enviar informes diarios de producción por correo electrónico ha reforzado significativamente el proceso de comunicación y difusión de información, evidenciando la veracidad de la tercera hipótesis específica. Este avance subraya el rol crítico de la automatización en la mejora de la eficiencia operativa y la productividad.

Finalmente se concluye que la implementación del módulo de comunicación IoT-BI en el sistema robótico PE501079170-2022-PROCIENCIA representa un paso adelante en la integración de tecnologías de la información en el ámbito agrícola, alineándose con las tendencias de la industria 4.0 aplicada al agro y marcando un hito en la optimización de los procesos de producción. Los resultados no solo validan la hipótesis general y específicas planteadas al inicio del estudio, sino que también ofrecen un marco sólido para futuras investigaciones en este campo. La confirmación de las hipótesis resalta la efectividad de los automatismos informáticos y la tecnología IoT-BI en la transformación de los sistemas productivos agrícolas, apoyando la noción de que la adopción de estas tecnologías es fundamental para la mejora continua y la sustentabilidad en la producción agrícola moderna.

6.2. RECOMENDACIONES

Estas recomendaciones buscan no solo abordar algunas de las limitaciones identificadas en el estudio actual sino también proporcionar una hoja de ruta para la mejora y expansión futura del sistema robótico PE501079170-2022-PROCIENCIA y sistemas similares. La implementación de estas sugerencias podría conducir a un sistema más seguro, eficiente y adaptable, listo para enfrentar los desafíos de la agricultura moderna y la gestión de datos, por lo que se recomienda lo siguiente:

Implementación de un Esquema de Seguridad Robusto: Es importante desarrollar e implementar un esquema de seguridad integral que abarque toda la arquitectura IoT-BI. Esto debería incluir la encriptación de datos en tránsito y en reposo, la autenticación robusta de dispositivos y usuarios, y la implementación de firewalls y sistemas de detección de intrusos específicos para IoT. Dado que la seguridad no fue un foco primordial de este estudio, futuras investigaciones deberían explorar en detalle las vulnerabilidades potenciales y proponer marcos de seguridad específicos para mitigarlos.

Expansión del Alcance de la Muestra: A pesar de la limitada disponibilidad de sistemas robóticos agrícolas similares, se recomienda buscar estrategias para ampliar la muestra en estudios futuros. Esto podría incluir la colaboración con otras instituciones o la inclusión de sistemas robóticos en diferentes etapas de desarrollo. Una muestra más grande permitiría una validación más robusta de los hallazgos actuales y ayudaría a entender mejor la generalización de los resultados.

Desarrollo de un Sistema de Gestión de Usuarios: Para controlar el acceso a la información generada y procesada por el módulo IoT-BI, se recomienda la implementación de un sistema de gestión de usuarios que defina claramente los roles y permisos. Esto asegurará que solo los usuarios autorizados puedan acceder a datos sensibles, contribuyendo a la integridad y confidencialidad de la información.

Adopción de Tecnologías de IA para Análisis y Redacción de Informes: Se sugiere explorar la integración de soluciones de inteligencia artificial y aprendizaje automático para automatizar y optimizar aún más las tareas de análisis de datos y redacción de informes. La IA podría ofrecer análisis predictivos avanzados y generar insights más profundos a partir de los datos recopilados, mejorando la toma de decisiones y la eficiencia operativa.

Estandarización del Módulo de Comunicación IoT-BI: Para maximizar la compatibilidad y adaptabilidad del sistema, se recomienda trabajar hacia la estandarización del módulo de comunicación IoT-BI. Esto facilitaría su integración con una variedad de sistemas, no limitándose solo a robots agrícolas, sino extendiéndose a otros contextos donde la automatización y el análisis de datos son cruciales, esto también permitirá una personalización y escalabilidad más sencillas, adaptándose a las necesidades cambiantes de los usuarios y las tecnologías emergentes.

7. REFERENCIAS BIBLIOGRÁFICAS

- Ahmad, S., Miskon, S., Alabdan, R., & Tlili, I. (2020). Towards Sustainable Textile and Apparel Industry: Exploring the Role of Business Intelligence Systems in the Era of Industry 4.0. *Sustainability*, 12(7), 2632-2655. <https://doi.org/https://doi.org/10.3390/su12072632>
- Amazon Web Services, I. (21 de 01 de 2024). *AWS | Informática en la nube. Ventajas y beneficios*. AWS | Informática en la nube. Ventajas y beneficios: https://aws.amazon.com/es/what-is-cloud-computing/?nc2=h_ql_le_int_cc
- Arcentales Toledo, J. W., & Slacedo Mena, T. J. (2023). *IMPLEMENTACIÓN DE UN APLICATIVO WEB PARA EL CONTROL DE PRODUCCIÓN Y LA GESTIÓN DE VENTAS DE PLANTAS E INJERTOS DE CACAO, APLICANDO INTELIGENCIA DE NEGOCIOS (BI)*. FACULTAD DE CIENCIAS AGRARIAS UNIVERSIDAD AGRARIA DEL ECUADOR.
- Barón Ramírez, E., García Estrella, C. W., & Sánchez Gárate, S. (2021). La inteligencia de negocios y la analítica de datos en los procesos empresariales. *REVISTA CIENTÍFICA DE SISTEMAS E INFORMÁTICA*, 1(2), 38-53. <https://doi.org/doi.org/10.51252/rcsi.v1i2.167>
- Basco, A. I., Beliz, G., Coatz, D., & Garnerero, P. (2018). *Industria 4.0: Fabricando el Futuro*. Inter-American Development Bank.
- Boonsiritomachai, W., McGrath, G., & Burgess, S. (2016). Exploring business intelligence and its depth of maturity in Thai SMEs. *Cogent Business & Management*, 3(1), 1220663. <https://doi.org/10.1080/23311975.2016.1220663>
- De La Cruz Mariños, G. F. (2022). *SISTEMA DE MEDICIÓN DEL ÁREA FOLIAR EN PLANTINES DE ALCACHOFA MEDIANTE PROCESAMIENTO DIGITAL DE IMÁGENES AÉREAS EN UN INVERNADERO EN HUAMACHUCO*. FACULTAD DE INGENIERÍA DE LA UNIVERSIDAD PRIVADA ANTENOR ORREGO.
- English, V., & Hoffmann, M. (2018). Business Intelligence as a Source of Competitive Advantage in SMEs: A Systematic Review. *DBS Business Review*, 10-32. <https://doi.org/DOI:10.22375/DBR.V2I0.23>

- García Honores, M. G., & Moncada Calmet, L. F. (2022). *Autosintonización difusa orientada al control de velocidad de un robot diferencial tipo oruga diseñado para un terreno agrícola de La Libertad*. Facultad de Ingeniería de la Universidad Privada Antenor Orrego.
- Herrera German, J. C. (2022). *Diseño e implementación de un sistema robótico para el posicionamiento efectivo de plantines de alcachofa contenidos en una bandeja multiceldas en viveros industriales de la región La Libertad*. Facultad de Ingeniería de la Universidad Nacional de Trujillo.
- Kutner, M. H., Nachtsheim, C. J., Neter, J., & Li, W. (2004). *Applied Linear Statistical Models*. McGraw-Hill.
https://doi.org/https://users.stat.ufl.edu/~winner/sta4211/ALSM_5Ed_Kutner.pdf
- Loyola Diaz, J. A. (2022). *Solución Business Intelligence para Mejorar la Toma de Decisiones del Área de Certificación de Cultivos Orgánicos de la Cooperativa Agraria CEPROVASC - 2022*. Escuela de Posgrado de la Universidad Cesar Vallejo.
- Luján Mora, S. (2001). *Programación en Internet: clientes web*. Universidad de Alicante. Departamento de Lenguajes y Sistemas Informáticos.
<https://doi.org/https://rua.ua.es/dspace/handle/10045/16994>
- Matamoros Zapata, R. (2010). *Implantación en una empresa de un sistema Business Intelligence SaaS / On Demand a través de la plataforma LITEBI*. VALENCIA: UNIVERSIDAD POLITÉCNICA DE VALENCIA.
- Mendoza Vásquez, E. A. (2021). *Implementación de herramientas Python en el proceso de producción de cultivos agrícolas del fundo "San Juan de Buenavista"*. FACULTAD DE INGENIERÍA Y ARQUITECTURA de la UNIVERSIDAD CÉSAR VALLEJO.
- Montgomery, D. C. (2013). *Design and analysis of experiments*. J. Wiley.
https://doi.org/https://indaga.ual.es/permalink/34CBUA_UAL/t0rgfc/alma991001387949704991
- Muntean, M. (2015). Considerations Regarding Business Intelligence in Cloud Context. *Informatica Economică*, 19(4), 55-67.
<https://doi.org/10.12948/issn14531305/19.4.2015.05>

- Muñoz Choque, A. M. (2021). ESTUDIO DE TIEMPOS Y SU RELACIÓN CON LA PRODUCTIVIDAD. *Enfoques*, 5(17), 40-54. <https://doi.org/https://www.redalyc.org/journal/6219/621968429003/html/>
- Pisco Gómez, Á., Regalado Jalca, J., Gutiérrez García, J., Quimis Sánchez, O., Marcillo Parrales, K., & Marcillo Merino, J. (2017). *FUNDAMENTOS SOBRE LA GESTIÓN DE BASE DE DATOS*. 3Ciencias. <https://doi.org/dx.doi.org/10.17993/ingyTec.2017.23>
- Shaukat, K., Iqbal, F., Mahboob Alam, T., Kaur Aujla, G., Devnath, L., Ghaffar Khan, A., . . . Rubab, A. (2020). The Impact of Artificial intelligence and Robotics on the Future Employment Opportunities. *Trends in Computer Science and Information Technology*, 5(1), 50-54. <https://doi.org/https://doi.org/10.17352/tcsit.000022>
- Singh Gurjar, Y., & Singh Rathore, V. (2013). Cloud Business Intelligence – Is What Business Need Today. *International Journal of Recent Technology and Engineering (IJRTE)*, 1(6), 81-86. <https://doi.org/https://www.ijrte.org/wp-content/uploads/papers/v1i6/F0431021613.pdf>
- Valverde, V., Portalanza, N., & Mora, P. (2019). ANÁLISIS DESCRIPTIVO DE BASE DE DATOS RELACIONAL Y NO RELACIONAL. *Revista Atlante: Cuadernos de Educación y*, 1, 1-16. <https://doi.org/https://www.eumed.net/rev/atlante/2019/06/base-datos-relacional.html>
- Vele Zhingri, C. (2015). *Análisis de rendimiento entre base de datos relacional: MySQL y una base de datos no relacional MongoDB*. Facultad de Ciencias de la Administración de la Universidad del Azuay.

8. ANEXOS

Anexo 1: Guía de observación de tiempos

Guía de Observación de tiempos

Fecha:			
Investigador:	Jorge Mendoza Chomba		
Variable:			
Dimensión:		Indicador:	
Subproceso:		Tarea:	

Pre-test		Post-test	
-----------------	--	------------------	--

N° de Obs.	Identificador de bandeja	Tiempo en segundos
1		
2		
3		
4		
5		
6		

Anexo 2: Script FireBaseDBPy.py

```
1. import firebase_admin
2. from firebase_admin import db, credentials
3. import pandas as pd
4. import openpyxl
5. bueno = 0
6. malo = 0
7.
8. cred = credentials.Certificate("credenciales.json")
9. firebase_admin.initialize_app(cred, {"databaseURL": "https://ag
robot-upao-default-rtdb.firebaseio.com/"})
10.
11.     csvFile = pd.read_csv("./Assets/rawRegs/dataset_4weeks_H1
W1A1_H2W2A2_.csv")
12.     csvFile.to_excel("./Assets/xlsxRegs/data1.xlsx")
13.
14.     excel_dataframe=openpyxl.load_workbook("./Assets/xlsxRegs
/data1.xlsx")
15.
16.     activeSheetExcelFile = excel_dataframe.active
17.
18.     print(activeSheetExcelFile)
19.
20.     #####Xlsx test#####
21.     for row in range(1, activeSheetExcelFile.max_row):
22.         for col in activeSheetExcelFile.iter_cols(2,2):
23.             if col[row].value == 1:
24.                 bueno=bueno+1
25.             else:
26.                 malo=malo+1
27.     print("buenos: ", bueno)
28.     print("malos: ", malo)
29.     #####testEnd#####
30.
31.     #ref = db.reference("/data/0/data/0")           #Leer
dato
32.
33.     #print(ref.get())
34.
35.     db.reference("/data/1/data/4").set(bueno)     #Carga
r dato
36.     db.reference("/data/3/data/4").set(malo)
37.
38.
```

Anexo 3: Script mailListChecker.py

```
1. from google.oauth2 import service_account
2. from googleapiclient.discovery import build
3.
4. def read_sheet(spreadsheet_id, range_name):
5.     # Ruta al archivo de clave de la cuenta de servicio
6.     SERVICE_ACCOUNT_FILE = 'C:/Users/jmend/OneDrive/Documents/
TesisMasterJM2023/Code/credenciales4mailChecking.json'
7.
8.     # Define los scopes requeridos
9.     SCOPES = ['https://www.googleapis.com/auth/spreadsheets.re
adonly']
10.
11.     # Autentica usando la cuenta de servicio
12.     credentials = service_account.Credentials.from_servic
e_account_file(
13.         SERVICE_ACCOUNT_FILE, scopes=SCOPES)
14.
15.     # Construye el servicio
16.     service = build('sheets', 'v4', credentials=credentia
ls)
17.
18.     # Realiza la solicitud a la API
19.     sheet = service.spreadsheets()
20.     result = sheet.values().get(spreadsheetId=spreadsheet
_id, range=range_name).execute()
21.     values = result.get('values', [])
22.
23.     return values # Devuelve los valores leídos
```

Anexo 4: Script mailSender.py

```
1. import smtplib
2. from email.mime.multipart import MIMEMultipart
3. from email.mime.text import MIMEText
4. import datetime
5. import os
6. from email.mime.application import MIMEApplication
7.
8. def send_emails(email_list, pdf_path):
9.     smtp_server = "smtp.gmail.com"
10.    smtp_port = 587
11.    sender_email = "agrobotupao@gmail.com"
12.    password = "bgtu tfpy cnzi iaec"
13.
14.    server = smtplib.SMTP(smtp_server, smtp_port)
15.    server.starttls()
16.    try:
17.        server.login(sender_email, password)
18.    except smtplib.SMTPAuthenticationError as e:
19.        print(f"Error de autenticación: {e}")
20.        return
21.
22.    # Obtener fecha y hora actual para asunto
23.    now = datetime.datetime.now()
24.    formatted_now = now.strftime("%Y-%m-%d %H:%M:%S")
25.    subject = f"reporte de operación {formatted_now}"
26.    body = "-Reporte de operación-"
27.
28.    for receiver_email in email_list:
29.        print('-----')
30.        print(receiver_email)
31.        msg = MIMEMultipart()
32.        msg['From'] = sender_email
33.        msg['To'] = receiver_email
34.        msg['Subject'] = subject
35.        msg.attach(MIMEText(body, 'plain'))
36.
37.        try:
38.            with open(pdf_path, "rb") as pdf_file:
39.                part = MIMEApplication(pdf_file.read(), N
ame=os.path.basename(pdf_path))
40.                # Agregar encabezados al adjunto
41.                part['Content-Disposition'] = 'attachment;
filename="%s"' % os.path.basename(pdf_path)
42.                msg.attach(part)
43.                server.sendmail(sender_email, receiver_email,
msg.as_string())
44.                print(f"Correo enviado a {receiver_email}")
45.            except Exception as e:
46.                print(f"No se pudo enviar el correo a
{receiver_email}: {e}")
47.
48.        server.quit()
```

Anexo 5: Script mailManager.py

```
1. from mailListChecker import read_sheet
2. from mailSender import send_emails
3. import time
4.
5. # ID de tu hoja de cálculo y el rango de celdas a leer
6. pdf_rep_path = "C:/Users/jmend/OneDrive/Documents/TesisMasterJ
   M2023/Code/Assets/pdfRegs/data1.pdf"
7. spreadsheet_id = '1QWWWcqQEhPkcy7yr-
   kZeolISmwJGipJTkeoC2FP4MH4'
8. range_name = 'A1:A4'
9.
10.     # Llama a la función y recibe los valores
11.     start=time.perf_counter()
12.     values = read_sheet(spreadsheet_id, range_name)
13.     duracion=time.perf_counter()-start
14.     print(duracion)
15.
16.     # Imprime lo que recibió
17.     if not values:
18.         print('No se encontraron datos.')
19.     else:
20.         print(values)
21.         email_list = [item[0] for item in values]
22.         print(email_list)
23.
24.         start=time.perf_counter()
25.         send_emails(email_list,pdf_rep_path)
26.         duracion=time.perf_counter()-start
27.         print(f'tiempo de envío {duracion}')
```

Anexo 6: Script infGenerator.py

```
1. from fpdf import FPDF
2. import pandas as pd
3. from datetime import datetime
4. from mailListChecker import read_sheet
5.
6. spreadsheet_id = '1q1gMa3gAiWe3vzLzGsjtYX5KkfiHYkdPpsgB1xnMEKU
   '
7. range_name = 'A1:B8'
8. fecha_de_informe = datetime.now().strftime("%Y-%m-%d
   %H:%M:%S")
9.
10.     class PDF(FPDF):
11.         def header(self):
12.             # Encabezado con fecha
13.             self.set_font('Arial', 'B', 12)
14.             self.cell(0, 10, f'Informe de Producción Agrobot
   {fecha_de_informe}', 0, 1, 'C')
15.
16.         def footer(self):
17.             # Pie de página con número de página
18.             self.set_y(-15)
19.             self.set_font('Arial', 'I', 8)
20.             self.cell(0, 10, f'Página
   {self.page_no()}', 0, 0, 'C')
21.
22.         def agregar_tabla(pdf, path_excel):
23.             # Leer el archivo Excel
24.             df = pd.read_excel(path_excel, engine='openpyxl')
25.             pdf.set_font("Arial", size=6)
26.             # Añadir las filas de la tabla
27.             for i in range(len(df)):
28.                 for col in df.columns:
29.                     pdf.cell(23, 7, str(df[col].iloc[i]), border=
   1)
30.
31.                 pdf.ln(7)
32.
33.         def agregar_imagen(pdf, path_imagen):
34.             # Añadir imagen
35.             pdf.image(path_imagen, x=10, y=pdf.get_y(), w=180, h=
   80)
36.
37.             # Crear el PDF
38.             pdf = PDF()
39.             pdf.add_page()
40.
41.             # Agregar la tabla
42.             path_excel = 'Code/Assets/xlsxRegs/data1.xlsx' # ruta
43.             agregar_tabla(pdf, path_excel)
44.
45.             # Añadir página para continuar después de los datos de
46.             excel
47.             pdf.add_page()
48.             pdf.ln(20)
```

```

47.
48.     # Título de la imagen
49.     pdf.set_font("Arial", 'B', size=24)
50.     pdf.cell(0, 10, 'Gráfica Resumen', 0, 1, 'C')
51.     pdf.ln(10)
52.
53.     # Agregar la imagen
54.     path_imagen = 'C:/Users/jmend/OneDrive/Documents/TesisMas
55.     terJM2023/captura_elemento.png' # ruta
56.     agregar_imagen(pdf, path_imagen)
57.     pdf.ln(100)
58.
59.     # Título de Incidencias
60.     pdf.set_font("Arial", 'B', size=24)
61.     pdf.cell(0, 10, 'Incidencias', 0, 1, 'C')
62.     pdf.ln(10)
63.
64.     #Leer tabla de incidencias
65.     values = read_sheet(spreadsheet_id, range_name)
66.
67.     # Escribir incidencias
68.     for value in values:
69.         pdf.set_font("Arial", 'B', size=8)
70.         pdf.cell(0, 10, value[0]+" - "+value[1], 0, 1, 'L')
71.
72.     # Guardar el PDF
73.     fecha_para_path=fecha_de_informe.replace(":", "_")
74.     nombre_pdf = f'informe_produccion_{fecha_para_path}.pdf'
75.     pdf.output(nombre_pdf)
76.     print(f"PDF generado: {nombre_pdf}")

```

Anexo 7: Script getInfo4Mail.py

```
1. import os
2. import time
3. from selenium import webdriver
4. from selenium.webdriver.common.by import By
5. from PIL import Image
6. from io import BytesIO
7.
8. # Establecer la ruta al GeckoDriver en la variable de entorno
  PATH
9. geckodriver_path = 'C:/Users/jmend/Downloads/geckodriver-
  v0.34.0-win64/geckodriver.exe'
10.     os.environ['PATH'] += os.pathsep + geckodriver_path
11.
12.     # Configuración de Selenium para usar Firefox
13.     firefox_options = webdriver.FirefoxOptions()
14.     firefox_options.add_argument('--headless')
15.     driver = webdriver.Firefox(options=firefox_options)
16.
17.     # Ajustar el tamaño de la ventana del navegador
18.     driver.set_window_size(1366, 768)
19.     # URL de la página web
20.     url = 'https://agrobot.website/'
21.
22.     # Abre la página web
23.     driver.get(url)
24.
25.     # Hacer scroll por una cantidad específica de píxeles
26.     scroll_height = 2000
27.     driver.execute_script(f"window.scrollTo(0,
  {scroll_height});")
28.
29.     # Espera para que el scroll se complete
30.     time.sleep(2.5)
31.
32.     # buscar el elemento por su clase
33.     element = driver.find_element(By.CLASS_NAME, 'chart-box')
34.
35.     # Obtener la ubicación y el tamaño del elemento para
  definir el área de captura
36.     location = element.location
37.     size = element.size
38.     png = driver.get_screenshot_as_png() # Captura la
  pantalla completa
39.
40.     # Cierra el navegador
41.     driver.quit()
42.
43.     # Procesa la captura de pantalla para obtener solo el
  elemento
44.     im = Image.open(BytesIO(png))
45.     left = location['x']
46.     top = location['y']
47.     right = location['x'] + size['width']
```

```
48.     bottom = location['y'] + size['height']
49.     im = im.crop((90, 218, 1256, 590)) # Define el área a
capturar
50.
51.     # Guarda la imagen
52.     im.save('captura_elemento.png')
53.
54.     print('Captura de pantalla realizada con éxito.')
```

Anexo 8: Tiempos del proceso de Generación de Reporte de Producción PreTest

LOTE	TIEMPOS POR ACTIVIDAD	B1	B2	B3	B4	B5	B6	SUBTOTAL
L1	tiempo algoritmo (S)	0.04860	0.05015	0.05003	0.05082	0.05168	0.05331	0.30459
	tiempo extracción (S)	22.787	26.561	22.867	27.545	27.351	21.477	148.58746
	tiempo procesamiento (S)	245.35	251.77	262.47	259.8	258.32	251.68	1529.38002
L2	tiempo algoritmo (S)	0.05144	0.04953	0.05239	0.05233	0.05479	0.04841	0.30890
	tiempo extracción (S)	25.181	26.15	24.165	23.303	27.541	28.93	155.27058
	tiempo procesamiento (S)	255.61	245.42	243.02	257.87	240.65	259.57	1502.15666
L3	tiempo algoritmo (S)	0.05078	0.05350	0.04832	0.05112	0.04928	0.04873	0.30175
	tiempo extracción (S)	24.351	21.393	26.409	21.971	28.909	24.826	147.85975
	tiempo procesamiento (S)	258.88	246.31	244.23	250.75	249.47	245.2	1494.84757
L4	tiempo algoritmo (S)	0.05311	0.05349	0.05463	0.05588	0.05479	0.05501	0.32691
	tiempo extracción (S)	29.279	29.274	21.682	21.571	25.646	25.143	152.59575
	tiempo procesamiento (S)	248.44	264.45	248.89	263.88	243.64	259.82	1529.12784
L5	tiempo algoritmo (S)	0.05184	0.05070	0.05157	0.05488	0.05134	0.05362	0.31395
	tiempo extracción (S)	21.468	27.163	26.875	28.86	28.472	26.069	158.90782
	tiempo procesamiento (S)	244.21	262.81	255.06	265.41	253.4	265.92	1546.80799
L6	tiempo algoritmo (S)	0.05341	0.05296	0.05512	0.05331	0.04990	0.05106	0.31575
	tiempo extracción (S)	24.286	24.602	22.32	25.47	27.42	24.587	148.68484
	tiempo procesamiento (S)	248.45	249.74	264.64	259.21	245.51	265.25	1532.79682
L7	tiempo algoritmo (S)	0.05083	0.04939	0.04819	0.05032	0.05143	0.05057	0.30072
	tiempo extracción (S)	23.851	28.874	27.952	25.493	21.652	24.11	151.93199
	tiempo procesamiento (S)	244.54	266.18	264.94	240.61	261.21	264.22	1541.70449
L8	tiempo algoritmo (S)	0.04933	0.05530	0.05452	0.05138	0.04836	0.05350	0.31240
	tiempo extracción (S)	22.418	22.112	27.593	22.913	27.832	21.598	144.46578
	tiempo procesamiento (S)	242.93	243.18	249.76	257.06	261.27	249.83	1504.03474
TOTAL								13391.64508

Anexo 9: Tiempos del proceso de Generación de Reporte de Producción PostTest

LOTE	TIEMPOS POR ACTIVIDAD	B1	B2	B3	B4	B5	B6	SUBTOTAL
L1	tiempo algoritmo (S)	0.055501	0.048786	0.055011	0.049131	0.054665	0.055225	0.31832
	tiempo extracción (S)	0.012382	0.012954	0.013150	0.012740	0.013034	0.012544	0.07680
	tiempo procesamiento (S)	6.530670	6.435553	6.115416	6.279783	6.174242	6.376272	37.91194
L2	tiempo algoritmo (S)	0.048555	0.053490	0.050332	0.054335	0.054920	0.051526	0.31316
	tiempo extracción (S)	0.012633	0.012876	0.012573	0.012958	0.012550	0.012636	0.07623
	tiempo procesamiento (S)	6.464036	6.508887	6.418954	6.530909	6.347128	6.547925	38.81784
L3	tiempo algoritmo (S)	0.049231	0.053791	0.055898	0.054355	0.048278	0.054530	0.31608
	tiempo extracción (S)	0.012422	0.012885	0.012760	0.012907	0.012510	0.013074	0.07656
	tiempo procesamiento (S)	6.395666	6.397785	6.590109	6.455430	6.174035	6.369876	38.38290
L4	tiempo algoritmo (S)	0.053063	0.051863	0.051884	0.051341	0.055854	0.048157	0.31216
	tiempo extracción (S)	0.013047	0.012880	0.012843	0.012407	0.012839	0.012568	0.07658
	tiempo procesamiento (S)	6.168602	6.434335	6.163041	6.501352	6.577771	6.427253	38.27235
L5	tiempo algoritmo (S)	0.055365	0.051593	0.053459	0.050764	0.055852	0.054825	0.32186
	tiempo extracción (S)	0.013236	0.012783	0.013280	0.012346	0.012417	0.012561	0.07662
	tiempo procesamiento (S)	6.512166	6.385460	6.587284	6.603398	6.569003	6.233302	38.89061
L6	tiempo algoritmo (S)	0.052402	0.053078	0.049867	0.051338	0.049036	0.050070	0.30579
	tiempo extracción (S)	0.012430	0.013256	0.012628	0.012684	0.013276	0.012595	0.07687
	tiempo procesamiento (S)	6.566772	6.278668	6.558603	6.439602	6.291041	6.320283	38.45497
L7	tiempo algoritmo (S)	0.051197	0.049936	0.052657	0.049387	0.048332	0.055927	0.30744
	tiempo extracción (S)	0.012656	0.012717	0.013089	0.013299	0.012974	0.013088	0.07782
	tiempo procesamiento (S)	6.317497	6.617494	6.540576	6.611853	6.483107	6.319443	38.88997
L8	tiempo algoritmo (S)	0.051380	0.054961	0.055643	0.053143	0.053144	0.048979	0.31725
	tiempo extracción (S)	0.013196	0.012952	0.012409	0.012368	0.012734	0.013160	0.07682
	tiempo procesamiento (S)	6.264722	6.238251	6.419731	6.202526	6.602485	6.616938	38.34465
TOTAL								311.09160

Anexo 10: Tiempos del proceso de Difusión de Reporte de Producción PreTest

LOTE	TIEMPOS POR ACTIVIDAD	B1	B2	B3	B4	B5	B6	SUBTOTAL
L1	tiempo de verificación de lista de correos	34.87	33.76	29.77	35.65	35.98	30.62	200.65
	tiempo de redacción y envío de correo	20.51	19.77	19.95	22.14	23.27	21.73	127.38
L2	tiempo de verificación de lista de correos	32.01	34.27	30.52	33.04	31.01	31.61	192.46
	tiempo de redacción y envío de correo	23.79	22.50	21.03	19.21	22.83	21.03	130.39
L3	tiempo de verificación de lista de correos	31.35	30.31	35.53	30.16	30.80	33.57	191.72
	tiempo de redacción y envío de correo	21.11	20.66	24.45	19.56	19.27	20.58	125.64
L4	tiempo de verificación de lista de correos	34.92	35.10	34.59	30.79	30.59	33.76	199.74
	tiempo de redacción y envío de correo	25.35	21.41	19.73	25.06	25.01	21.60	138.16
L5	tiempo de verificación de lista de correos	31.74	35.99	31.06	34.41	31.17	31.36	195.73
	tiempo de redacción y envío de correo	23.44	24.02	19.92	24.66	19.33	21.98	133.34
L6	tiempo de verificación de lista de correos	34.82	33.51	33.50	30.03	34.27	33.52	199.65
	tiempo de redacción y envío de correo	19.59	19.70	19.76	24.24	22.60	22.71	128.60
L7	tiempo de verificación de lista de correos	30.32	30.65	33.13	29.94	34.84	30.40	189.28
	tiempo de redacción y envío de correo	19.70	24.94	19.27	24.57	20.94	19.70	129.13
L8	tiempo de verificación de lista de correos	34.40	31.86	30.81	32.19	36.38	33.42	199.06
	tiempo de redacción y envío de correo	21.80	24.04	20.93	19.86	25.05	24.88	136.56
TOTAL								2617.48338

Anexo 11: Tiempos del proceso de Difusión de Reporte de Producción PostTest

LOTE	TIEMPOS POR ACTIVIDAD	B1	B2	B3	B4	B5	B6	SUBTOTAL
L1	tiempo de verificación de lista de correos	2.6695452	2.9226027	2.8626341	2.6251567	2.4884734	3.1076122	16.67602424
	tiempo de redacción y envío de correo	9.0945423	8.5850575	9.7577363	8.7253101	9.4060416	9.8913266	55.46001439
L2	tiempo de verificación de lista de correos	2.8325673	3.0207422	2.4929267	2.5213901	2.6702448	2.4908152	16.02868645
	tiempo de redacción y envío de correo	8.4417529	9.9760714	9.5280787	9.0154324	9.6951811	9.5030376	56.15955418
L3	tiempo de verificación de lista de correos	3.0659159	2.8552653	3.0700397	2.7168072	3.0438769	2.5715737	17.32347874
	tiempo de redacción y envío de correo	8.9331944	9.3855735	9.8314799	8.633985	8.783793	9.9361551	55.50418095
L4	tiempo de verificación de lista de correos	2.5638058	2.7018868	2.9345513	2.7870964	2.5454068	3.0546557	16.58740293
	tiempo de redacción y envío de correo	8.1878643	9.5019877	9.9146684	9.7645475	9.6239618	8.9156317	55.90866127
L5	tiempo de verificación de lista de correos	2.7203916	2.8140151	2.9423495	2.4606743	3.0837249	2.5907003	16.61185576
	tiempo de redacción y envío de correo	9.6680888	9.0775035	9.7196871	9.3506534	9.3284255	9.9428177	57.08717594
L6	tiempo de verificación de lista de correos	2.6387276	3.0876349	2.4735075	2.5794856	3.082908	2.4516232	16.31388664
	tiempo de redacción y envío de correo	8.5394451	8.9724951	9.4631924	9.5394372	9.8531458	9.9233603	56.29107587
L7	tiempo de verificación de lista de correos	2.8565974	2.6002621	2.9814291	2.4790282	2.6147651	2.4934494	16.02553126
	tiempo de redacción y envío de correo	8.2455113	9.6005698	9.6003515	8.4855532	8.8342077	9.9215637	54.6877572
L8	tiempo de verificación de lista de correos	2.644177	3.0129311	2.6177506	2.9634886	2.5681403	2.8407152	16.64720291
	tiempo de redacción y envío de correo	8.2140189	8.5910773	9.3406129	8.4415421	8.9058079	8.7326253	52.22568434
TOTAL								575.5381731

Anexo 12: Tiempos del proceso de Generación - Difusión de Informe Diario de Producción PreTest

LOTE	TIEMPOS POR ACTIVIDAD	DURACIÓN (S)
L1	Generación de tablas resumen (S)	201.89
	Generación de gráficas de tablas resumen (S)	68.30
	Redacción de informe con incidencias (S)	838.77
	Verificación lista de correos de los interesados (S)	34.18
	Redacción y envío de correo con el informe diario (S)	24.84
L2	Generación de tablas resumen con gráficas (S)	177.64
	Generación de tablas resumen (S)	76.68
	Redacción de informe con incidencias (S)	827.51
	Verificación lista de correos de los interesados (S)	35.57
	Redacción y envío de correo con el informe diario (S)	21.01
L3	Generación de tablas resumen con gráficas (S)	168.23
	Generación de tablas resumen (S)	60.91
	Redacción de informe con incidencias (S)	783.05
	Verificación lista de correos de los interesados (S)	36.20
	Redacción y envío de correo con el informe diario (S)	23.39
L4	Generación de tablas resumen con gráficas (S)	200.56
	Generación de tablas resumen (S)	70.76
	Redacción de informe con incidencias (S)	662.18
	Verificación lista de correos de los interesados (S)	32.49
	Redacción y envío de correo con el informe diario (S)	22.22
L5	Generación de tablas resumen con gráficas (S)	204.48
	Generación de tablas resumen (S)	69.73
	Redacción de informe con incidencias (S)	640.06
	Verificación lista de correos de los interesados (S)	35.35
	Redacción y envío de correo con el informe diario (S)	22.51
L6	Generación de tablas resumen con gráficas (S)	164.29
	Generación de tablas resumen (S)	87.99
	Redacción de informe con incidencias (S)	711.69
	Verificación lista de correos de los interesados (S)	32.87
	Redacción y envío de correo con el informe diario (S)	19.94
L7	Generación de tablas resumen con gráficas (S)	165.15
	Generación de tablas resumen (S)	65.44
	Redacción de informe con incidencias (S)	563.55
	Verificación lista de correos de los interesados (S)	32.56
	Redacción y envío de correo con el informe diario (S)	22.40
L8	Generación de tablas resumen con gráficas (S)	188.00
	Generación de tablas resumen (S)	65.01
	Redacción de informe con incidencias (S)	783.79
	Verificación lista de correos de los interesados (S)	30.45
	Redacción y envío de correo con el informe diario (S)	19.36
TOTAL		8291.00

Anexo 13: Tiempos del proceso de Generación - Difusión de Informe Diario de Producción PostTest

LOTE	TIEMPOS POR ACTIVIDAD	DURACIÓN (S)
L1	Generación de tablas resumen (S)	5.008913092
	Generación de gráficas de tablas resumen (S)	3.534015143
	Redacción de informe con incidencias (S)	10.26460146
	Verificación lista de correos de los interesados (S)	2.524483923
	Redacción y envío de correo con el informe diario (S)	9.84336463
L2	Generación de tablas resumen con gráficas (S)	4.834568576
	Generación de tablas resumen (S)	3.511757578
	Redacción de informe con incidencias (S)	9.458191157
	Verificación lista de correos de los interesados (S)	3.076690345
	Redacción y envío de correo con el informe diario (S)	9.777600836
L3	Generación de tablas resumen con gráficas (S)	4.414901296
	Generación de tablas resumen (S)	3.484359771
	Redacción de informe con incidencias (S)	9.77879766
	Verificación lista de correos de los interesados (S)	2.916029432
	Redacción y envío de correo con el informe diario (S)	9.399880476
L4	Generación de tablas resumen con gráficas (S)	4.570756089
	Generación de tablas resumen (S)	3.889496575
	Redacción de informe con incidencias (S)	10.06221896
	Verificación lista de correos de los interesados (S)	2.741259709
	Redacción y envío de correo con el informe diario (S)	8.732432292
L5	Generación de tablas resumen con gráficas (S)	5.077281176
	Generación de tablas resumen (S)	3.699448704
	Redacción de informe con incidencias (S)	11.04643035
	Verificación lista de correos de los interesados (S)	3.086870802
	Redacción y envío de correo con el informe diario (S)	9.532420933
L6	Generación de tablas resumen con gráficas (S)	4.870812705
	Generación de tablas resumen (S)	3.589201601
	Redacción de informe con incidencias (S)	9.565906476
	Verificación lista de correos de los interesados (S)	3.075895032
	Redacción y envío de correo con el informe diario (S)	8.649917341
L7	Generación de tablas resumen con gráficas (S)	4.365734703
	Generación de tablas resumen (S)	4.019041458
	Redacción de informe con incidencias (S)	9.576685774
	Verificación lista de correos de los interesados (S)	3.090924979
	Redacción y envío de correo con el informe diario (S)	9.39496687
L8	Generación de tablas resumen con gráficas (S)	4.445389059
	Generación de tablas resumen (S)	3.81727479
	Redacción de informe con incidencias (S)	9.958380441
	Verificación lista de correos de los interesados (S)	2.989658469
	Redacción y envío de correo con el informe diario (S)	9.578940297
TOTAL		245.255501