



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

Diseño y construcción de lockers automáticos para la recepción de paquetes utilizando RPA (robotic process automation) para agilizar el proceso de envío de encomiendas a WOLF.

Inca Cujilema, Galo Fabricio y Teneda Espín, Alvaro José.

Departamento de Ciencias de la Energía y Mecánica.

Carrera de Ingeniería Mecatrónica.

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

Ing. Mendoza Chipantasi, Darío José.

10 de Agosto del 2022.

Latacunga.

Reporte de verificación de contenidos

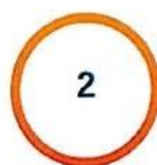
The logo for COPYLEAKS, featuring the word "COPYLEAKS" in white capital letters on a dark blue background. The letter "O" is stylized with a blue circular outline.

Tesis Inca Teneda.pdf

Scanned on: 13:26 August 10, 2022 UTC



Overall Similarity Score



Results Found



Total Words in Text

Identical Words	43
Words with Minor Changes	1
Paraphrased Words	12
Omitted Words	1647

A handwritten signature in blue ink, enclosed in a blue oval. The signature appears to read "Ing. Dario José Mendoza Chipantasi".

Ing. Mendoza Chipantasi, Dario José

C. C.: 0603110834



Departamento de Ciencias de la Energía y Mecánica

Carrera de Ingeniería Mecatrónica

Certificación

Certifico que el trabajo de titulación: **"Diseño y construcción de lockers automáticos para la recepción de paquetes utilizando RPA (robotic process automation) para agilizar el proceso de envío de encomiendas a WOLF"** fue realizado por los señores **Inca Cujilema, Galo Fabricio y Teneda Espín, Alvaro José**; el mismo que cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, además fue revisado y analizado en su totalidad por la herramienta de prevención y/o verificación de similitud de contenidos; razón por la cual me permito acreditar y autorizar para que se lo sustente públicamente.

Latacunga, 10 de agosto del 2022

Ing. Mendoza Chípantasi, Dario José

C. C.: 0603110834



Departamento de Ciencias de la Energía y Mecánica

Carrera de Ingeniería Mecatrónica

Responsabilidad de Autoría

Nosotros, **Inca Cujilema, Galo Fabricio y Teneda Espín, Alvaro José**, con cédulas de ciudadanía n° 0604464537 y 1804844866, declaramos que el contenido, ideas y criterios del trabajo de titulación: **“Diseño y construcción de lockers automáticos para la recepción de paquetes utilizando RPA (robotic process automation) para agilizar el proceso de envío de encomiendas a WOLF”** es de nuestra autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Latacunga, 10 de agosto del 2022

Inca Cujilema, Galo Fabricio

C.C.: 0604464537

Teneda Espín, Alvaro José

C.C.: 1804844866



Departamento de Ciencias de la Energía y Mecánica

Carrera de Ingeniería Mecatrónica

Autorización de Publicación

Nosotros, **Inca Cujilema, Galo Fabricio y Teneda Espín, Alvaro José**, con cédula/cédulas de ciudadanía n° 0604464537 y 1804844866, autorizamos a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: **“Diseño y construcción de lockers automáticos para la recepción de paquetes utilizando RPA (robotic process automation) para agilizar el proceso de envío de encomiendas a WOLF”** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi/nuestra responsabilidad.

Latacunga, 10 de agosto del 2022

Inca Cujilema, Galo Fabricio

C.C.: 0604464537

Teneda Espín, Alvaro José

C.C.: 1804844866

Dedicatoria

El presente trabajo es dedicado a mi Madre Ana y a mi padre Galo por el esfuerzo y confianza depositada. Quienes me acompañaron en este camino de arduo trabajo, siendo además mi sostén para seguir adelante. A mi tía Lola que ha sido otro pilar fundamental en mi formación profesional. A mis hermanos Danny y Madeley que son mi fuente de inspiración para cada día ser mejor.

A todos los miembros de mi familia que contribuyeron para poder culminar esta etapa muy importante de mi vida.

A mis amigos que me acompañaron durante esta hermosa etapa de mi vida y fueron un pilar fundamental para mi formación personal.

Inca Cujilema, Galo Fabricio

Dedicatoria

El presente escrito se lo dedico a todas las personas que me han acompañado a lo largo de mi formación académica, así como aquellas que han formado parte de mi formación personal.

A mi madre, Paty, por haberme apoyado incondicionalmente, e inculcado que “nada viene fácil en la vida; todo amerita esfuerzo y sacrificio”.

A mi padre, William, por ser un ejemplo de superación personal y académica; demostrándome que se puede perder todo, menos el amor a lo que uno hace.

A mi hermana y mi sobrina que las quiero demasiado, les dedico este trabajo como recordatorio de que la inteligencia no se define por cuanto sabes, sino como aplicas el conocimiento que adquieres para solucionar problemas en tu vida.

Teneda Espín, Alvaro José

Agradecimiento

En primer lugar, quiero agradecer a Dios por haberme dado la vida y permitirme cumplir con este sueño en mi vida.

Quiero agradecer a mi familia por el apoyo y la confianza depositada en mi persona, por ser el pilar fundamental de mi vida y darme todo su afecto.

A Corporación WOLF S.A., por brindarme la oportunidad de desarrollar mis habilidades en el campo de la ingeniería y ser una guía educativa y moral.

Un agradecimiento a mis amigos y compañero por haber hecho de esta etapa una experiencia única y muy especial para mí.

Inca Cujilema, Galo Fabricio

Agradecimiento

Agradezco a mis padres por darme la oportunidad de cursar una carrera universitaria y apoyarme en todo lo que ha estado a su alcance para que la culmine.

Agradezco a cada uno de los integrantes de corporación Wolf por darnos la confianza y la oportunidad de poder desarrollar este proyecto, así como el apoyo para poder solucionar problemas durante la elaboración del mismo.

Agradezco a mi compañero por su arduo trabajo y compromiso a lo largo del desarrollo del proyecto.

Teneda Espín, Alvaro José

ÍNDICE DE CONTENIDOS

Carátula	1
Reporte de verificación de contenidos	2
Certificación	3
Responsabilidad de autoría	4
Autorización de publicación	5
Dedicatoria	6
Dedicatoria	7
Agradecimiento.....	8
Agradecimiento.....	9
Índice de contenidos	10
índice de figuras	15
índice de tablas.....	20
índice de ecuaciones.....	22
Resumen	23
Abstract.....	24
Capítulo I: Introducción.....	25
Antecedentes.....	25
Planteamiento del problema.....	28
Justificación e importancia	29
Objetivos.....	30
<i>Objetivo general</i>	30
<i>Objetivos específicos</i>	30
Hipótesis.....	31
Variables de la investigación.....	31
<i>Variables independientes</i>	31
<i>Variables dependientes</i>	31
Capítulo II: Fundamentación teórica	32

Automatización de procesos por robótica (RPA).....	32
Centro de servicios compartidos (CSC)	34
Máquina para recepción de encomiendas.....	35
Pasarelas de pago.....	36
HMI (Interfaz humano máquina)	36
<i>Dar control al usuario</i>	37
<i>Reducir la carga de memoria del usuario</i>	37
<i>Consistencia de tareas e interacciones</i>	37
Protocolos de comunicación IoT	37
<i>HTTP (REST/JSON)</i>	38
<i>MQTT (Message Queuing Telemetry Transport)</i>	39
<i>CoAP (Constrained Application Protocol)</i>	39
Tarjetas de adquisición de datos	39
<i>Raspberry PI 4</i>	39
<i>Microcontroladores ATMEL AVR</i>	40
Pirámide de automatización	41
<i>Nivel de proceso (Red de campo)</i>	41
<i>Nivel de control (Red de control)</i>	42
<i>Nivel de supervisión (SCADA, HMI)</i>	42
Balanzas electrónicas y celdas de carga.....	42
Sistema de seguridad electromecánica.....	42
Modularidad.....	43
Balanzas electrónicas y celdas de carga.....	45
Sensores y actuadores	46
<i>Sensor Hall</i>	46
<i>Cerraduras electromagnéticas</i>	47
<i>Cerraduras tipo solenoide</i>	48
<i>Acondicionador de señal</i>	48

Protocolos de comunicación.....	49
<i>Protocolo RS-485</i>	49
<i>Conexión con RJ-45</i>	51
Capítulo III: Diseño del sistema	53
Diseño del sistema bajo RPA	53
Diseño modular	57
<i>Unión modular</i>	58
<i>Comunicación modular</i>	60
Diseño mecánico.....	64
<i>Diseño del sistema de pesaje</i>	64
<i>Cálculo estructural del sistema de pesaje</i>	73
<i>Simulaciones del sistema de pesaje</i>	77
Diseño electrónico	80
<i>Selección de componentes para la seguridad electrónica</i>	80
<i>Selección de microcontrolador</i>	82
<i>Diseño de pistas</i>	83
Diseño software	88
<i>Esquema de conexión</i>	88
<i>Arquitectura Backend-Frontend</i>	90
<i>Comunicación Raspberry con servidor remoto</i>	92
<i>Comunicación Raspberry - maestro</i>	94
Diseño del HMI	96
<i>Arquitectura de comunicación</i>	96
Diagrama general del sistema	97
Capítulo IV: Construcción del locker.....	99
Implementación del sistema de pesaje.....	99
<i>Percha interna</i>	99
<i>Lámina de apoyo</i>	100

	13
<i>Base rectangular</i>	101
Parte electrónica	101
<i>Expansión modular</i>	101
<i>Seguridad</i>	105
<i>Implementación de las galgas</i>	106
<i>Elaboración de las placas maestro y esclavo</i>	108
Implementación de la comunicación con el servidor	111
<i>Construcción de la estructura Backend-Frontend</i>	113
<i>Construcción de la estructura Backend / comunicación serial</i>	114
<i>Administración de los periféricos</i>	117
<i>Diseño de la estructura Backend-servidor remoto</i>	118
<i>Modo administrador</i>	123
<i>Modo de usuario</i>	126
Construcción del HMI para el usuario	128
Capítulo V: Pruebas y análisis de resultados	136
Pruebas de seguridad mecánica.....	137
<i>Evaluación de la cerradura pequeña</i>	141
<i>Comprobación de hipótesis de seguridad</i>	143
Pruebas de comunicación maestro-esclavo	145
<i>Comprobación de hipótesis de la comunicación serial</i>	148
Pruebas del sistema de pesaje.....	150
<i>Comprobación de hipótesis de la medición del sistema de pesaje</i>	161
Implementación de RPA en el proceso de envío de encomiendas	164
<i>Validación de la hipótesis del sistema</i>	168
Capítulo VI: Conclusiones y recomendaciones.....	171
Conclusiones.....	171
Recomendaciones.....	176
Bibliografía.....	179

Anexos 184

ÍNDICE DE FIGURAS

Figura 1. <i>Roles de RPA</i>	33
Figura 2. <i>Ejemplo de funcionamiento de un robot en CSC</i>	35
Figura 3. <i>Pirámide de la automatización</i>	41
Figura 4. <i>Arquitectura modular</i>	44
Figura 5. <i>Interruptores de efecto Hall</i>	47
Figura 6. <i>Cerradura electromagnética</i>	47
Figura 7. <i>Cerradura tipo solenoide</i>	48
Figura 8. <i>Módulo HX711</i>	49
Figura 9. <i>Topología RS485 y RS422</i>	50
Figura 10. <i>RJ-45 pin de salida</i>	52
Figura 11. <i>Esquema de color RJ-45 de cable directo</i>	52
Figura 12. <i>Diagrama del alcance del servido local bajo RPA y CSC</i>	56
Figura 13. <i>Modularidad de lockers externa</i>	58
Figura 14. <i>Unión mecánica</i>	59
Figura 15. <i>Unión electrónica</i>	59
Figura 16. <i>Expansión mecánica</i>	60
Figura 17. <i>Conexión maestro-esclavo</i>	62
Figura 18. <i>Circuito de conexión para MAX 485 con delimitador</i>	62
Figura 19. <i>Conexión para lectura de pines del Dip switch</i>	64
Figura 20. <i>Esquema modular electrónico</i>	64
Figura 21. <i>Diseño del subsistema de pesaje</i>	66
Figura 22. <i>Platina de perfil en L</i>	70
Figura 23. <i>Platina de perfil rectangular</i>	71
Figura 24. <i>Diseño de percha interna</i>	71
Figura 25. <i>Diseño de lámina de apoyo</i>	74
Figura 26. <i>Criterios de selección de balanzas</i>	76
Figura 27. <i>Simulación de la deformación máxima en la percha</i>	77

Figura 28. <i>Simulación de esfuerzo máxima en la percha</i>	77
Figura 29. <i>Simulación de la deformación con cargas puntuales</i>	78
Figura 30. <i>Simulación de la deformación por esfuerzo en la placa</i>	78
Figura 31. <i>Simulación con cargas puntuales por Von Mises</i>	79
Figura 32. <i>Simulación del esfuerzo en la placa por Von Mises</i>	79
Figura 33. <i>Parámetros de cálculo para pistas de control</i>	85
Figura 34. <i>Parámetros de cálculo para pistas de potencia</i>	86
Figura 35. <i>Disposición de la placa por bloques funcionales recomendada</i>	87
Figura 36. <i>Diseño de placa maestro</i>	88
Figura 37. <i>Diseño de placa esclavo</i>	88
Figura 38. <i>Esquema de comunicación del sistema</i>	89
Figura 39. <i>Esquema de funcionamiento del sistema Backend-Frontend</i>	92
Figura 40. <i>Diagrama general de proceso del usuario</i>	96
Figura 41. <i>Esquema de conexión para la comunicación</i>	97
Figura 42. <i>Pirámide de automatización para el servicio de envío de paquetes</i>	97
Figura 43. <i>Diagrama general de conexión para la comunicación</i>	98
Figura 44. <i>Subsistema de pesaje</i>	99
Figura 45. <i>Percha interna</i>	100
Figura 46. <i>Lámina de apoyo</i>	100
Figura 47. <i>Base rectangular</i>	101
Figura 48. <i>Conexión prueba 1 maestro-1 esclavo</i>	102
Figura 49. <i>Conexión prueba 1 maestro-2 esclavos</i>	103
Figura 50. <i>Comunicación y respuesta del esclavo</i>	104
Figura 51. <i>Designación de pines para puerto Rj45</i>	104
Figura 52. <i>Conexión para cerradura magnética</i>	105
Figura 53. <i>Placa de calibración para galgas</i>	106
Figura 54. <i>Conexión de galgas con cable de señal color verde</i>	107
Figura 55. <i>Conexión de galgas con cable de señal color blanco</i>	107

Figura 56. <i>Conexión de 4 galgas de un cuarto de puente</i>	108
Figura 57. <i>Construcción de placa maestro</i>	109
Figura 58. <i>Construcción de placa esclavo</i>	109
Figura 59. <i>Primera versión placas esclavo y maestro</i>	110
Figura 60. <i>Distribución de elementos por áreas según EMC</i>	110
Figura 61. <i>Construcción de placa con componentes SMD</i>	111
Figura 62. <i>Comparativa de placas esclavo</i>	111
Figura 63. <i>Conexión de periféricos a la Raspberry Pi 4</i>	112
Figura 64. <i>Diagrama de comunicación por peticiones</i>	114
Figura 65. <i>HMI para la comunicación por tramas Backend-maestro/esclavo</i>	116
Figura 66. <i>Respuesta de la petición que llega a la Raspberry por el puerto serial</i>	116
Figura 67. <i>Diagrama de comunicación por peticiones para el puerto serial</i>	117
Figura 68. <i>Arquitectura de comunicación MQTT</i>	119
Figura 69. <i>Diagrama de flujo de conexión del cliente MQTT</i>	120
Figura 70. <i>Diagrama de secuencia para el envío de datos por MQTT</i>	122
Figura 71. <i>Diagrama de flujo de conexión del cliente HTTP</i>	123
Figura 72. <i>Servicios inicializados en el servidor Backend</i>	124
Figura 73. <i>Conexión verificada en el servidor por MQTT y HTTP respectivamente</i> ..	124
Figura 74. <i>Conexión verificada en el broker por HTTP</i>	125
Figura 75. <i>Comandos de respuesta en el puerto serial en el Backend NestJS</i>	125
Figura 76. <i>Registro de los cambios de estado del locker en el servidor remoto</i>	126
Figura 77. <i>Autenticación de código de verificación</i>	127
Figura 78. <i>Comando de respuesta en el servidor remoto por MQTT con token</i>	128
Figura 79. <i>Página principal de HMI</i>	129
Figura 80. <i>Selección entre recoger y enviar paquetes</i>	130
Figura 81. <i>Direcciones de las páginas del HMI</i>	130
Figura 82. <i>Asignación de las URL</i>	131
Figura 83. <i>Página de selección entre código y escáner QR</i>	131

Figura 84. <i>Diagrama de secuencia para el ingreso de código de verificación.....</i>	132
Figura 85. <i>Pantallas de alerta para la verificación del código</i>	133
Figura 86. <i>Pantalla de información para el nuevo tarifado.....</i>	134
Figura 87. <i>Diagrama de secuencia para la verificación del pago.....</i>	134
Figura 88. <i>Prototipo a escala de prueba.....</i>	136
Figura 89. <i>Áreas de estudio para la seguridad mecánica del locker.....</i>	137
Figura 90. <i>Diseño mecánico para puerta.....</i>	140
Figura 91. <i>Doble de la percha en la zona frontal</i>	141
Figura 92. <i>Prueba de cerradura pequeña.....</i>	141
Figura 93. <i>Prueba chi-cuadrada para asociación: Dispositivos por temperatura</i>	144
Figura 94. <i>Tarjeta de informe para Dispositivos por temperatura</i>	145
Figura 95. <i>Pruebas de comunicación por tramas en maestro-esclavo</i>	148
Figura 96. <i>Prueba chi cuadrada para asociación: caracteres</i>	149
Figura 97. <i>Tarjeta de informe para caracteres enviados y recibidos.....</i>	150
Figura 98. <i>Esquema del producto y su agrupación por elementos</i>	151
Figura 99. <i>Disposición geométrica</i>	151
Figura 100. <i>Pruebas de funcionamiento de sensado.....</i>	152
Figura 101. <i>Error locker 3 (Abajo)</i>	156
Figura 102. <i>Error locker 2 (Medio).....</i>	156
Figura 103. <i>Error locker 1 (Arriba).....</i>	157
Figura 104. <i>Error corregido locker 3 (Abajo)</i>	159
Figura 105. <i>Error corregido locker 2 (Medio).....</i>	160
Figura 106. <i>Error corregido locker 1 (Arriba)</i>	161
Figura 107. <i>Prueba chi-cuadrada por asociación: Pruebas por casillero</i>	162
Figura 108. <i>Tarjeta de informe para pruebas de los casilleros</i>	163
Figura 109. <i>Proceso de envío de paquetes con la adición de RPA</i>	164
Figura 110. <i>Pruebas en el ensamblaje total del locker principal</i>	165
Figura 111. <i>Prueba de chi-cuadrada para asociación: Funcionalidad</i>	169

Figura 112. *Tarjeta de informe para asociación: Funcionalidad* 170

ÍNDICE DE TABLAS

Tabla 1. <i>Tipos de estructura para HTTP</i>	38
Tabla 2. <i>Parámetros de estado para HTTP</i>	38
Tabla 3. <i>Arquitectura modular</i>	40
Tabla 4. <i>Tipos de configuración de puente de Wheatstone</i>	46
Tabla 5. <i>Tabla de criterio de automatización para RPA</i>	53
Tabla 6. <i>Tabla de automatización de procesos mediante RPA</i>	54
Tabla 7. <i>Criterios de selección</i>	56
Tabla 8. <i>Tipo de unión</i>	58
Tabla 9. <i>Protocolos de comunicación</i>	60
Tabla 10. <i>Selección de sensor para el sistema de pesaje</i>	65
Tabla 11. <i>Criterios de selección de material</i>	66
Tabla 12. <i>Fuerzas aplicadas en la placa base</i>	74
Tabla 13. <i>Especificaciones de las cerraduras magnéticas</i>	80
Tabla 14. <i>Especificaciones del elemento de control</i>	81
Tabla 15. <i>Especificaciones del sensor final de carrera</i>	82
Tabla 16. <i>Especificaciones del microcontrolador</i>	83
Tabla 17. <i>Parámetros de ruteado</i>	86
Tabla 18. <i>Parámetros de selección del framework de Frontend</i>	90
Tabla 19. <i>Criterios de selección del lenguaje de programación para el sistema</i>	91
Tabla 20. <i>Los parámetros de selección para el protocolo</i>	92
Tabla 21. <i>Criterios de selección de Raspberry con placas maestro/esclavo</i>	94
Tabla 22. <i>Parámetros de conexión maestro-1 esclavo</i>	101
Tabla 23. <i>Parámetros de conexión 1 maestro- 2 esclavos</i>	102
Tabla 24. <i>Pines del puerto Rj485</i>	105
Tabla 25. <i>Sistema HMI</i>	112
Tabla 26. <i>Estados de las variables el sistema</i>	115
Tabla 27. <i>GPIO utilizados del teclado matricial</i>	117

Tabla 28. <i>Comandos para configurar la pantalla</i>	118
Tabla 29. <i>Control de comando de respuesta</i>	121
Tabla 30. <i>Datos de los destornilladores para las pruebas de seguridad</i>	137
Tabla 31. <i>Promedio de prueba de cerraduras</i>	138
Tabla 32. <i>Respuesta a lo largo del tiempo de seguridad electrónica</i>	142
Tabla 33. <i>Tabla de datos para el chi-cuadrado</i>	143
Tabla 34. <i>Tabla de conteos esperados y observados para dispositivos</i>	144
Tabla 35. <i>Pruebas de comunicación</i>	146
Tabla 36. <i>Prueba 5 de retorno de pesos de todos los casilleros</i>	147
Tabla 37. <i>Tabla de datos para el chi-cuadrado</i>	149
Tabla 38. <i>Tabla de conteos esperados y observados para tramas</i>	150
Tabla 39. <i>Pruebas de pesaje el locker</i>	153
Tabla 40. <i>Errores máximos permitidos</i>	155
Tabla 41. <i>Pruebas de pesaje el locker con el factor de corrección</i>	157
Tabla 42. <i>Tabla de datos para el chi-cuadrado</i>	162
Tabla 43. <i>Tabla de conteos esperados y observados para pruebas por casillero</i>	163
Tabla 44. <i>Datos del tiempo de uso de sistema de pesaje</i>	165
Tabla 45. <i>Datos recolectados del sistema en cada intento de envío del paquete</i>	167
Tabla 46. <i>Tabla de datos para el chi-cuadrado</i>	168
Tabla 47. <i>Informa de diagnóstico para asociación: tiempo, seguridad y sensado</i>	169

ÍNDICE DE ECUACIONES

Ecuación 1. <i>Ley de Ohm</i>	63
Ecuación 2. <i>Suma de resistencias en paralelo</i>	63
Ecuación 3. <i>Carga viva por piso</i>	67
Ecuación 4. <i>Carga viva total</i>	68
Ecuación 5. <i>Factor de carga</i>	68
Ecuación 6. <i>Factor de carga final</i>	69
Ecuación 7. <i>Área cooperante</i>	69
Ecuación 8. <i>Resistencia requerida</i>	69
Ecuación 9. <i>Área de la columna</i>	70
Ecuación 10. <i>Ecuación de la fuerza de peso</i>	73
Ecuación 11. <i>Ecuación del esfuerzo máximo en el centro de una lámina</i>	75
Ecuación 12. <i>Ecuación para el cálculo de área</i>	84
Ecuación 13. <i>Ecuación para el ancho de pista</i>	84

Resumen

En el presente proyecto de investigación se plantea el diseño y la construcción de un prototipo de casilleros modulares automáticos enfocados en la recepción de paquetes teniendo como base unas dimensiones de 1,8m de alto y unos 0,5 m de ancho. Este prototipo pretende agilizar el proceso usando tecnologías RPA aplicados en CSC, para ello se estableció un computador central que controla el proceso secuencial como: conectar al locker con el servidor remoto y registra su nombre, ubicación y por ende informa el número de casilleros disponibles. Previo al uso del sistema es necesario que el cliente registre los datos del paquete en la aplicación móvil y que se genere el código de envío. El computador central ejecuta el HMI y permite iniciar el proceso secuencial al cliente, donde debe elegir el método de validación de código, ya sea por el ingreso manual de 6 dígitos o con la lectura en el escáner Qr. Esta validación usa peticiones HTTP hacia el servidor remoto para otorgar los permisos de uso del casillero. Si el código es correcto, se envía el comando de acción a las placas maestros/esclavos usando el puerto serial. La secuencia de pesaje requiere que la puerta esté abierta, que se ingrese el paquete y cerrar el casillero para inicializar la calibración de la balanza y obtener el peso real del paquete. Estos valores de estados son enviados a manera de comandos hacia el puerto serial del computador central e inmediatamente se actualizan en el registro del servidor remoto utilizando MQTT para calcular la tarifa. Se verifica mediante el HMI que el cliente cerró la puerta y con HTTP se devuelven los datos actualizados del paquete con la nueva tarifa mostrada en pantalla donde el usuario puede aceptar o rechazar los nuevos valores.

Palabras clave: Automatización robótica de procesos (RPA), locker, Centros de servicios compartidos (CSC), tarifa de un servicio de paquetería.

Abstract

This research project proposes the design and construction of a prototype of automatic modular lockers focused on the reception of packages with dimensions of 1.8 m high and 0.5 m wide. This prototype aims to streamline the process using RPA technologies applied in CSC, for this a central computer was established that controls the sequential process such as: Connects the Locker with the remote server and records its name, location and thus reports the number of lockers available. Prior to the use of the system, it is necessary for the client to register the package data in the mobile application and for the shipping code to be generated. The central computer runs the HMI and allows the customer to initiate the sequential process, where he must choose the code validation method, either by manual entry of 6 digits or by reading the QR scanner. This validation uses HTTP requests to the remote server to grant permissions to use the locker. If the code is correct, the action command is sent to the Master/Slave boards using the serial port. The weighing sequence requires the door to be open, the package to be entered and the locker to be closed to initialize the scale calibration and obtain the actual weight of the package. These status values are sent as commands to the serial port of the host computer and immediately updated in the remote server register using MQTT to calculate the tariff. It is verified by the HMI that the client closed the door and with HTTP the updated package data is returned with the new rate displayed on the screen where the user can accept or reject the new values.

Keywords: Robotic process automation (RPA), locker, Shared service centers (CSC), rate of a parcel service.

Capítulo I

Introducción.

Antecedentes.

Con el paso de los años, se ha evidenciado la importancia de la mensajería como medio de comunicación, permitiendo el acceso a la información entre las personas. Así mismo, también ha evolucionado y se ha sistematizado el proceso logístico para mejorar los tiempos de entrega, a través de diferentes medios de transporte para la distribución de los servicios de mensajería y encomiendas.

La historia del servicio de mensajería se establece en el Antiguo Egipto, donde se evidenció que el sistema era de uso principal del faraón, remontándose a la base de la invención de la escritura donde se enviaban comunicados de un sitio a otro utilizando un intermediario que se transportaba en caballería o caminando.

Los incas recorían miles de kilómetros para transportar mensajes de forma oral entre los altos mandos y militares a lo largo de todo su territorio, se los denominaba "Chasquis". Una parte fundamental era su sistema de contabilidad y de escritura denominado "khipu"; y, su principal equipamiento era el qëpi (mochilla o morral) a la espalda donde llevaban objetos y encomiendas.

A partir de 1769 al establecerse la colonia en la real Audiencia de Quito, se creó un sistema para el servicio de correos con jornadas laborales. Su primer despacho, establecido el 7 de mayo de 1779, abarcó el territorio desde Riobamba hasta Perú volviéndose la marca postal más antigua conocida hasta el momento. (Tello, 2015)

Durante la presidencia del doctor Gabriel García Moreno, el 01 de enero de 1865 se puso en circulación la primera emisión postal ecuatoriana con un valor facial de medio Real y dos de 1 Real en colores azul, verde y amarillo respectivamente. Además, García Moreno trajo al país los primeros buzones, uno de los cuales sería instalado años más tarde por el presidente Ignacio de Veintimilla en el Teatro Sucre. Además, en 1894 se emitió el primer Mapa Nacional de Líneas Postales y Telegráficas

del Ecuador. (Correos del Ecuador a través del tiempo – Gob, 2016). Este fue el primer sistema de tarifa postal establecido en el país, basándose en una clasificación jerárquica de colores de acuerdo al servicio que se requiera, como: correspondencia ordinaria, encomiendas y telegramas.

En 1927 José Luis Tamayo inaugura el Palacio de Correos, con el nombre de Casa de Correos de Quito, dónde el servicio de correos en esta época se manejaba únicamente en las oficinas, para lo cual los clientes debían acercarse a la Administración de Correos a retirar su correspondencia, hasta el 07 de diciembre de 1928, fecha en la que el cuerpo de carteros hace su primera entrega de correspondencia a domicilio (Correos del Ecuador a través del tiempo – Gob, 2016).

A partir de 1970 se incorporaron vehículos motorizados al servicio de encomiendas para facilitar la distribución de los paquetes a nivel nacional, dónde progresivamente se implementaron vehículos con mayor capacidad de almacenamiento con la adquisición de nuevas flotas de camiones, camionetas y motocicletas para la distribución de correspondencia y paquetería.

Siendo correos del Ecuador la primera empresa postal de Sudamérica que obtiene la certificación plata en “Seguridad Postal “anclada a las normas de seguridad S58 y S59 por los altos niveles de seguridad de instalaciones, procedimientos, centros de traslado postal y el manejo de correspondencia en los aeropuertos (Correos del Ecuador a través del tiempo – Gob, 2016). Actualmente cuenta con 322 puntos de servicio en todo el país.

Existen diferentes tipos de empresas, cada una enfocada a ciertos sectores, como por ejemplo el sector postal que posee un importante mercado. A nivel global en el 2017, los ingresos en términos nominales se incrementaron en 4% con respecto al año 2016. Actualmente, el servicio de encomiendas es muy utilizado en este sector, pues en el año 2018 se facturó 150.2 millones de dólares, donde el 64% de la facturación corresponde a paquetería y el 36% a documentos postales (Agencia de regulación y control postal, 2018).

En Ecuador, a finales de 2018, el sector postal contaba con 161 operadores postales, lo que representó un incremento del 61% de operadores postales habilitados con respecto a 2017, donde cerca de la mitad utilizaban algún tipo de software en sus sistemas logísticos para el control de envíos (Agencia de regulación y control postal, 2018).

Las oficinas de correspondencia requieren de un sistema ágil y sencillo para atender volúmenes grandes de paquetes. La solución viable es el uso o instalación de casilleros electrónicos que ayuden en la organización y registro de grandes volúmenes de paquetes.

Al ofrecer transparencia integral, los casilleros Inteligentes de correo y paquetería aumentan la conveniencia para sus empleados y significan un ahorro de tiempo y trabajo para el personal de la oficina de correspondencia. Esta solución mejora la seguridad y simplifica las cadenas de productividad. (Ricoh, 2019). Existen normativas que facilitan la construcción de este tipo de buzones, como se lo muestra en la página de correos del ecuador EP.

La automatización de procesos a través de robots, o por sus siglas en inglés (RPA), es el tema del momento de los profesionales de Centros de Servicios Compartidos (CSC), aunque muchos de los CSC aún están en proceso de exploración de lo que RPA (Robotic Process Automation) puede llegar a hacer. (De la Hoz, 2016). Dentro de las soluciones RPA se puede referenciar dos tipologías claras: Aquellos robots permiten gestionar procesos no definidos, mediante un aprendizaje probabilístico y tratando datos no estructurados. Por lo contrario, la otra variante se enfoca en procesos deterministas, empleando datos estructurados y procesos basados en normativa. Estos últimos son los más empleados para generar centros de servicios compartidos, ya que ejecutan procesos transaccionales que suelen ser repetitivos y basados en una serie de reglas. Su naturaleza los hace ideales para generar un mejor desempeño y mayor calidad en este tipo de tareas. (Deloitte, 2016).

Una amplia gama de empresas está considerando la automatización de los servicios a través de máquinas innovadoras que incluyen RPA para realizar tareas como validación de datos, generación de facturas, pago de reclamaciones y servicios, entre otros. La clave está en utilizar la digitalización de estos procesos para construir una estrategia robusta que facilite la toma de decisiones en tiempo real y entregue resultados que generen valor al cliente. (Aldana, 2017).

Planteamiento del problema

Existen proyectos orientados a la creación de este tipo de sistemas inteligentes aplicados en los servicios de envío y recepción de encomiendas pero que actualmente no están implementados por todas las empresas de Courier del país.

El principal inconveniente de los servicios de paquetería, es que se rigen por un proceso tradicional en el que gran parte se lo realiza manualmente, por lo tanto, existen áreas que retardan el proceso de transacciones, generando una necesidad entre consumidores. Partiendo de esta premisa se puede plantear que uno de los mayores obstáculos es el hecho de que el peso del paquete, el destino y el pago está sujeto al error humano y a su dinámica de trabajo.

El servicio de paquetería en el Ecuador está centralizado en cada ciudad, por lo tanto, el usuario requiere una inversión de tiempo y dinero para movilizarse y realizar el envío de su paquete, tomando en cuenta el horario de funcionamiento del establecimiento de paquetería.

Actualmente, los procesos de paquetería carecen de automatización al mejoramiento del ciclo de recepción, calidad, eficiencia y seguridad en la distribución de los productos. El servicio de paquetería tiene una implementación básica de tecnologías para control y supervisión, más no para recepción de paquetes, siendo el operario el que realiza la mayor parte del trabajo. De esta forma, la demanda de la transacción de consumidores es limitada a la eficiencia del operario; por ende, para generar un crecimiento significativo al sector comercial, es necesario la contratación de personal calificado que satisfaga la demanda.

La misión de la Universidad de las Fuerzas Armadas ESPE es potenciar el avance tecnológico y la investigación científica, aplicando estos conocimientos de manera apropiada para el beneficio de la sociedad, como en este caso, que existe procesos sin automatizar y que pueden ser solventados por los estudiantes de la carrera de ingeniería mecatrónica.

Justificación e importancia.

La automatización robótica de procesos promete un ahorro de costes, fiabilidad y eficiencia en procesos fáciles de descomponer en reglas sencillas y con un gran volumen de transacciones. También son claves en flujos de trabajo con grandes cargas documentales y propicios al error humano

En un principio, se busca implementar el servicio del proyecto, con horarios de atención reducidos a establecimientos con 12 horas de funcionamiento. En definitiva, este servicio se considera como una parte importante para el desarrollo del comercio, fortaleciendo la logística de la recepción y distribución de los productos, debido a que actualmente las jornadas de trabajo están pensadas en 8 horas y con esto se ganaría un 50% de atención.

El propósito del presente proyecto es el desarrollo de un sistema que permita la recepción de paquetes basándose en la combinación de las diferentes tecnologías anteriormente planteadas, como son RPA y CSC, implementándolas en la elaboración de casilleros modulares automáticos, enfocada en la optimización del proceso de recepción de paquetes, facilitando el servicio de pago para el servicio de encomiendas.

Actualmente, todos los procesos de paquetería son susceptibles al mejoramiento del ciclo de recepción, calidad, eficiencia y seguridad en la distribución de los productos. La aplicación de esto permite el manejo de diferentes tecnologías, las cuales facilitan el control y supervisión, con ello esta herramienta busca cubrir la demanda de la transacción para encomiendas; siendo así un gran apoyo en el sector comercial.

Con la creciente demanda en el servicio de paquetería, es viable implementar un sistema de expansión modular para lockers, permitiendo una flexibilidad con respecto al volumen habitual de los paquetes y en función a las dimensiones del lugar de instalación. De esta manera se fortalecería el área de recepción de paquetes basada en la logística y la trazabilidad.

Objetivos

Objetivo general

- Diseñar y construir un prototipo de casilleros modulares automatizados para la recepción de paquetes.

Objetivos específicos

- Diseñar y construir los casilleros modulares basándose en las necesidades de eficiencia y las diferentes normativas para la fabricación del prototipo.
- Evaluar que el sistema mecánico planteado para que los lockers funcionen correctamente, brinden seguridad y cumplan los requerimientos de la Empresa.
- Diseñar los circuitos de control con sus respectivos diagramas circuitales para PCB que satisfagan las necesidades del usuario y den seguridad al momento de depositar los paquetes.
- Elaborar la placa PCB para los circuitos diseñados y evaluar el correcto funcionamiento de manera física para cada uno de los casilleros al momento de recibir el paquete
- Diseñar la capa de comunicación que permita la recepción de datos entre el hardware y un servidor remoto.
- Evaluar el funcionamiento del diseño modular que facilite la expansión del servicio mediante la automatización del mismo.
- Facilitar el servicio de pago para envío de encomiendas mediante la implementación de una interfaz amigable.
- Realizar pruebas de funcionamiento del sistema completo y de su modularidad.

Hipótesis

¿El diseño y construcción de lockers automáticos para la recepción de paquetes utilizando RPA (robotic process automation) permitirá agilizar el proceso de envío de encomiendas a WOLF?

Variables de la investigación

Variables independientes

- Lockers automáticos para la recepción de paquetes

Variables dependientes

- Agilizar el proceso de envío de encomiendas utilizando RPA (robotic process automation)

Capítulo II

Fundamentación teórica

Automatización de procesos por robótica (RPA)

Actualmente, RPA (Robotic Process Automation) representa un modelo de automatización de procesos transaccionales, basados en reglas específicas. Sin embargo, no existe un marco de referencia común obligatorio para identificar un proceso que pueda ser automatizado.

Al decidir usar e implementar RPA, las empresas deben tomar en cuenta que RPA es óptimo para tareas de gran volumen de manera estandarizada donde no se necesita juicio subjetivo, creatividad o interpretación de datos. Según Aguirre-Rodríguez (2017). “Procesos de negocios tales como cuentas por pagar, cuentas por cobrar, facturación, viajes y gastos, activos fijos y administración de recursos humanos son buenos candidatos para la implementación de esta tecnología del software”

Estos procesos se pueden automatizarse en varios niveles, ya sea que todo el proceso puede sustituirse por un software optimizado o en su defecto solo un pequeño proceso que no necesite emitir un juicio de valor como es el caso del servicio de verificación de pagos o sistemas de pesaje automáticos.

En RPA, el robot no es físico debido a que se refiere a un software desarrollado para reemplazar tareas repetitivas. “Utiliza reglas lógicas pre-construidas para entregar resultados. Está conformado por macros con capacidad de realizar múltiples funciones a través de múltiples plataformas”. (Deloitte, 2017).

En este sentido, RPA se aplica a sistemas programables que van a estar en contacto directo con el usuario y le permite a este alcanzar los objetivos en menor tiempo y con una exactitud mayor, además de que a nivel de actualizaciones e innovación no tiene límites en los parámetros establecidos.

Según Becerra et al (2019) “La tecnología RPA es un software que replica las interacciones de un usuario humano, este trabaja a través de varias funciones,

aplicaciones y permite la automatización de actividades manuales, repetitivas, basadas en reglas que no requieren tomar decisiones/pensar”.

A nivel organizacional, las multinacionales más importantes han implementado estas tecnologías para mejorar su rendimiento y productividad, por lo que separa las actividades de bajo nivel que pueden ser reemplazadas de las actividades que requieren un monitoreo y toma de decisiones, logrando que sus trabajadores se dediquen a actividades de mayor importancia.

“BMW, en su planta de Spartanburg al Sur de California, automatizó labores físicas, permitiendo que sus trabajadores se encargaran de procesos que requerían análisis, toma de decisiones y control de calidad” (Dow Jones & Company Inc, 2018).

La implementación de RPA se ha expandido a numerosas áreas en la industria, permitiendo una mejora en la calidad del servicio y la productividad. Por ejemplo, “en salud tras la implementación de Bots se permitió al personal realizar actividades relacionadas con el trato de casos especiales y de alto cuidado de pacientes” (Volkenburgh, 2018).

Figura 1.

Roles de RPA



Nota. Tomado de “La era de la Automatización”: (Deloitte, 2016)

Centro de servicios compartidos (CSC)

Aplicar RPA en el mundo de los servicios compartidos resulta ser el punto de beneficio más alto para una industria pues, maximiza las ganancias y sobre todo monitorea y gestiona el proceso con las herramientas de CSS. Es aquí donde “la automatización de procesos soportados por software de robótica RPA en actividades administrativas derivan en mejoras sobre la ejecución de flujos de trabajo”. (Deloitte,2016).

Los centros de servicios compartidos deben contener y mantener sistemas de información de última tecnología, convirtiéndose en una herramienta fundamental que soporte la optimización de los procesos y la productividad. Asimismo, “necesitan procesos estandarizados para la aplicación rápida, eficiente y sin problemas y, con ello, eliminar procesos duplicados” (Marín, 2016). Es muy importante que los sistemas tengan una estructura clara para que la lógica de funcionamiento sea entendida por los desarrolladores y puedan dar un seguimiento o mantenimiento a los procesos automatizados.

RPA en CSC busca la forma de automatizar un sistema haciendo uso del software que se conoce como robot, “un robot CSC se define como un software de computadora que se ejecuta a base de programas que reemplazan tareas de humanos y permite ampliar su funcionalidad por medio de macros”. (Deloitte,2016).

Un robot en CSC no tiene características físicas, sin embargo, puede actuar como un medio de enlace con un mecanismo físico como un brazo robótico o un sistema de sensores y actuadores mientras la aplicación necesite de ello, pero su propósito fundamental se centra en servicios de tecnologías de la información.

La automatización de medio digitales permite mejorar sistemas utilizando un robot CSS.

Figura 2.

Ejemplo de funcionamiento de un robot en CSC



Nota. Tomado de “La era de la Automatización”: (Deloitte, 2016).

Como se muestra en la figura 2 un proceso automatizado genera optimizaciones y reduce costos que pueden ser aprovechados para una reducción de precios finales y competir en el mercado, estableciendo un valor agregado en los productos.

Máquina para recepción de encomiendas

Los casilleros inteligentes de paquetería presentan una mejora significativa en cuanto a logística, seguridad, y entrega de encomiendas. “Las entregas y recogidas de correspondencia se monitorean automáticamente, manteniendo una estricta cadena de custodia. Se trata de una solución de entrega de paquetería que transformará la oficina de correspondencia.” (Ricoh, 2019).

La transferencia de mercancía es más segura y no necesita de tener contacto con otra persona, además de que puede colocar el casillero en cualquier parte de la ciudad, lo que permite al usuario utilizar el casillero que mejor le convenga hablando desde el punto de vista logístico.

Pasarelas de pago

Las pasarelas de pagos funcionan como un enlace entre el usuario y su cuenta bancaria, el cobro se realiza mediante transferencias con un nivel de seguridad muy alto, garantizando que los pagos se realicen de forma correcta gracias a protocolos de seguridad desarrollados específicamente para el intercambio de datos entre un servidor y el usuario.

Esta estrategia evita que la información sea interceptada por terceras personas, comprometiendo datos sensibles. Al establecer una infraestructura de clave pública con sus procedimientos y mecanismos se fomenta la seguridad entre las entidades participantes cuando se realizan transacciones bancarias online. (Fonseca et al, 2013)

RPA en procesos PAC (Pedido de cobro) evita pérdidas millonarias en el cobro de facturas, según Deloitte México “Alrededor del 80% de la cartera vencida está relacionada con facturas que contienen errores de datos” (Deloitte, 2017). Algunas empresas contratan personal que se dedica a compensar estas fallas, sin embargo, CSC puede suplir este punto, según la empresa Automation Anywhere “el empleo de robótica puede reducir hasta un 60% el costo de procesamiento por factura” (Deloitte, 2017). Estos procesos emulan las actividades de un trabajador, pero además puede mejorar el proceso usando herramientas de optimización y administración de datos que agiliza el flujo de trabajo.

HMI (Interfaz humano máquina)

HMI significa “Human Machine Interface”, y representa el enlace gráfico entre la máquina y el usuario, mediante la implementación de paneles de control, información o registro. “Las señales entrantes pertenecen a tarjetas de entrada/salida en el computador, PLC’s (Controladores lógicos programables), RTU (Unidades remotas de I/O) o DRIVE’s (Variadores de velocidad de motores). (Salichs, 2012).

Un HMI permite la visualización de datos en tiempo real haciendo uso de contenido gráfico para el usuario, este puede implementarse a un nivel muy sencillo o de forma profesional en sistemas muy complejos.

Según Rodríguez (2013), el diseño de interfaces se basa en 3 reglas básicas como se describe a continuación.

Dar control al usuario

El diseñador debe dar al usuario la posibilidad de realizar su trabajo y que se adapte a sus exigencias

- Permitir al usuario utilizar teclado o mouse.
- Permitir al usuario interrumpir y reanudar la tarea.
- Utilizar mensajes y textos descriptivos.
- Reducir la carga de memoria del usuario.

Reducir la carga de memoria del usuario

El usuario no debe memorizar y recordar información.

- Proporcionar funciones, deshacer, rehacer y acciones por defecto
- Utilizar metáforas del mundo real (sistema telefónico, agenda)
- Presentar al usuario solo la información que necesita
- Hacer clara la presentación visual

Consistencia de tareas e interacciones

- Consistencia a la realización de tareas.
- Consistencia en las interacciones y las respuestas.
- Consistencia del aspecto estético.
- Permitir la libre exploración de la interfaz, sin miedo a consecuencias negativas.

Protocolos de comunicación IoT

La Plataforma IoT (Internet of things) tiene el potencial de automatizar y aumentar la productividad de modelos de negocios. La conectividad de sus servicios

se da mediante protocolos de comunicación que se rigen bajo normas y reglas para que dos entidades puedan intercambiar información, entre estos destacan los más importantes.

HTTP (REST/JSON)

Sus siglas Hypertext Transfer Protocol, es un protocolo que “se basa en cliente/servidor con incontables herramientas de código abierto que usan HTTP, y que todo lenguaje de codificación tiene bibliotecas HTTP, es muy accesible”. (Semle, 2016)

Este protocolo gira en torno a REST (Representational State Transfer), es un modelo donde los clientes pueden acceder a recursos del servidor a través de peticiones, en este sentido, HTTP brinda el transporte, pero no define la estructura del mensaje, por ello la presentación puede expresarse en formatos como JSON (JavaScript Object Notatios), HTML (HyperText Markup Language), JavaScript, XML (Extensible Markup Language).

Tabla 1.

Tipos de estructura para HTTP

XML	JSON
<item>	
<value>50</value>	
<context>	“value”: 50, “context”: {“status”:
<status>abierto </status>	“abierto”}
</context>	}
</item>	

Los parámetros de peticiones HTTP responden a:

Tabla 2.

Parámetros de estado para HTTP

Estado	Descripción
200 OK	La petición ha tenido éxito.

Estado	Descripción
400 Bad Request	La petición no es válida
401 Unauthorized	Error en las acciones, autenticación es posible
404 Not found	El servidor no encuentra el recurso solicitado
503 No server error	Los servicios no están disponibles/listos

MQTT (Message Queuing Telemetry Transport)

Es un protocolo con diseño publicar/suscribir destinado a usarse en sistema SCADA (Supervisory Control and Data Acquisition), se caracteriza por ser abierto, ligero y simple, es uno de los más confiables para sistemas remotos que actúa sobre el protocolo de transmisión (TCP).

La aplicación ligera y sencilla del protocolo MQTT, de acuerdo al trabajo M2M Protocols for Constrained Environments in the Context of IoT: A Comparison of Approaches, presenta una mejor conformidad a los patrones de comunicación (calidad, seguridad y escalabilidad) de acuerdo al escenario de aplicación (Frigieri et al., 2015).

CoAP (Constrained Application Protocol)

CoAP fue creado para ser compatible con HTTP con la carga mínima posible. “CoAP es similar a HTTP, pero usa UDP/multicast en lugar de TCP.” (AADECA, 2016). CoAP es utilizado cuando protocolos como HTTP es muy demandante en recursos, especialmente si se trata de ancho de banda, sin embargo, sus limitaciones como el desarrollo en comparación a otros protocolos y que el servidor no necesariamente realizara la confirmación de la petición, funciona correctamente en aplicaciones simples y limitadas.

Tarjetas de adquisición de datos

Raspberry PI 4

Es un ordenador de bajo costo con un sistema operativo de código abierto Raspbian, fue desarrollado en el Reino Unido con fines educativos en ciencia de la computación. En su arquitectura consta de puertos USB (Universal Serial Bus),

conectores HDMI, salida de video, puerto TCP/IP, slot para a tarjeta SD y con un procesador de 700 Mhz.

Tabla 3.

Arquitectura modular

Características	Raspberry PI 4
Procesador	ARM Cortex-A72
Frecuencia de reloj	1,5 GHz
GPU	Video Core VI (con soporte para OpenGL ES 3.x)
Memoria	1 GB / 2 GB / 4 GB LPDDR4 SDRAM
Conectividad	Bluetooth 5.0, Wi-Fi 802.11ac, Gigabit Ethernet
Puertos	GPIO 40 pines 2 x micro HDMI 2 x USB 2.0 2 x USB 3.0 CSI (cámara Raspberry Pi) DSI (pantalla táctil) micro SD conector de audio Jack USB-C (alimentación)

Nota. recuperado de la página de Xacata, Raspberry Pi 4 es oficial (Xacata,2019)

Microcontroladores ATMEL AVR

Los AVR (Advaced Virtual RISC) son una familia de microcontroladores de Atmel que fueron creados para ejecutar código en C con arquitectura Harvard con características como:

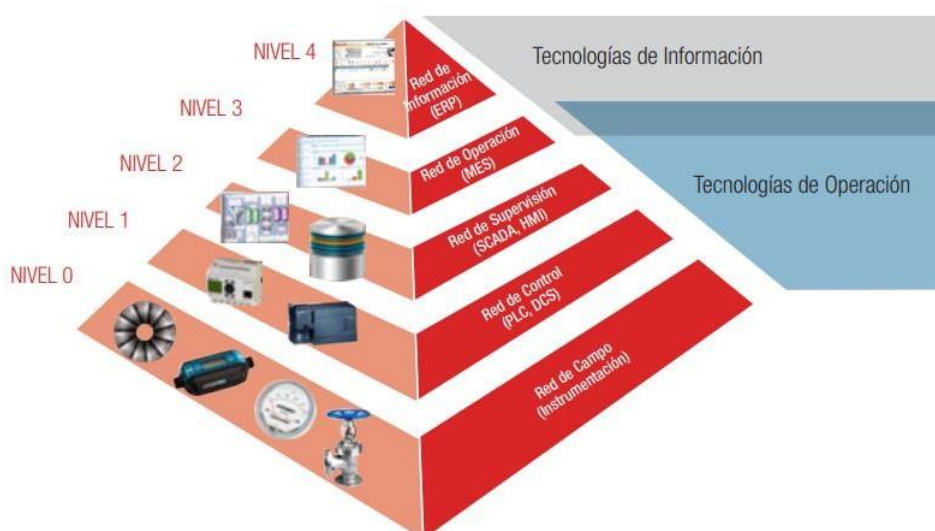
- Los registros 0 al 15 tienen diferentes capacidades de direccionamiento que los registros 16 al 31.
- Los registros de I/O 0 al 31 tienen distintas características que las posiciones 32 al 63.
- La instrucción CLR afecta los 'flag', mientras que la instrucción SER no lo hace, a pesar de que parecen ser instrucciones complementarias (dejar todos los bits en 1 y dejar todos los bits en 0 respectivamente). (Marone,2012)

Pirámide de automatización

En la industria se define que el éxito de una empresa va a depender de la fluidez y continuidad de la información en toda la cadena de valor. Para llegar a cumplir este objetivo, se plantea la pirámide de automatización como una “integración gráfica de los equipos y computadoras que forman parte del proceso en todos los niveles de fabricación, con la finalidad de optimizar la calidad, producción, automatización de operaciones y la gestión eficiente en las diferentes áreas de la empresa.” (ATCE,2022)

Figura 3.

Pirámide de la automatización



Nota. Tomado de “Niveles que forman la pirámide de automatización” (p. 1) (CCI-es, 2022).

La pirámide de automatización tiene cinco niveles de clasificación, pero no siempre es así, pues algunos niveles van a eliminarse si el proyecto no dispone de las tecnologías. En el proyecto de desarrollo de prototipos por lo general solo se trabaja en el área de tecnologías de operación que son:

Nivel de proceso (Red de campo)

Se caracteriza por estar en contacto directo con el entorno para adquirir datos mediante sensores y ejecutar acciones en equipos y máquinas.

Nivel de control (Red de control)

En este nivel se agrupan los controladores del sistema, ya sean microcontroladores, PLCs, DCS, etc., que permitan utilizar los datos del nivel de proceso y ejecutar arquitecturas de control en los actuadores

Nivel de supervisión (SCADA, HMI)

Los equipos en este nivel están programados para controlar la secuencia de los procesos, para administrar los datos de los niveles inferiores y presentar todo el sistema mediante arquitecturas gráficas.

Balanzas electrónicas y celdas de carga.

Las balanzas electrónicas son más precisas en comparación a las balanzas con principio mecánico, esto se debe a la velocidad de obtención y procesamiento de datos gracias a sistemas computarizados. Por otro lado, de importancia creciente son las celdas de carga, porque se eliminan en ellas los pivotes y piezas móviles y producen una corriente eléctrica que pueden utilizarse para fines de control y registros directos de datos. Requieren muy poco espacio, su colocación se adapta a la forma del recipiente, siendo posible usar desde una hasta 4 celdas. El mantenimiento de las celdas de carga, indicadores y componentes en general es prácticamente nulo. No hay partes en movimiento, ni desgaste. (Hernández et al, 2018).

Sistema de seguridad electromecánica.

La normativa NECT tiene por objeto fijar las condiciones mínimas de seguridad que deben cumplir las instalaciones eléctricas en bajo voltaje, con el fin de salvaguardar a las personas que las operan o hacen uso de ellas, proteger los equipos y preservar el ambiente en que han sido construidas. (NEC, 2018)

Su cumplimiento, junto a un adecuado mantenimiento, garantiza una instalación básicamente libre de riesgos; sin embargo, no garantiza necesariamente la eficiencia, buen servicio, flexibilidad y facilidad de ampliación de las instalaciones. (NEC, 2013)

Modularidad

Según Tseng. M. (2018) “Las ventajas del diseño modular son flexibilidad del diseño, el aumento y la reducción de costos. Debido a la agrupación de los componentes de cada módulo, el diseñador puede modificar fácilmente cada módulo en lugar de cambiar todo el diseño.”

La necesidad de expansión e implementación sencilla de un sistema con la finalidad de solventar las necesidades del usuario o los requerimientos de una empresa, se pueden solucionar al desarrollar un modelo de fácil reconfiguración.

Una de las características de las máquinas o herramienta reconfigurables es la modularidad, tanto desde el punto de vista estructural como desde el punto de vista del controlador. La modularidad permite una rápida y efectiva integración durante el cambio y/o modificación estructural de la MHR de una primera capacidad de producción a una segunda capacidad de producción, para fabricar un volumen determinado de un producto o familia de productos. (Pérez et al, 2005).

Pese a que la modularidad es una de las características más importantes para poder catalogar a la máquina como reconfigurable, también hay que tomar en cuenta otros aspectos como lo menciona Stone.

Las características que determinan la facilidad de reconfiguración son las siguientes: modularidad, integrabilidad, convertibilidad y diagnosticabilidad (Stone, 1997). La modularidad ha sido estudiada ampliamente desde el punto de vista funcional y constructivo (Fixon, 2001).

Ulrich al momento de hablar de modularidad, expresa que es fundamental tomar en cuenta la arquitectura que beneficiará al producto que se plantea desarrollar, tanto a largo plazo como a corto plazo.

El tipo de arquitectura que se decida implementar afecta el cambio del producto, variedad de productos, estandarización de componentes, rendimiento de producto, capacidad de manufactura y gestión del desarrollo del producto; de igual manera, las arquitecturas modulares se las plantea como aquellas que cada parte

física activa un conjunto específico de elementos funcionales e interaccionan con los otros trozos. (Ulrich K. &Eppinger S., 2013, p 203).

Los tipos de arquitecturas más empleadas para el diseño modular de un producto son:

- Arquitectura modular de ranura
- Arquitectura modular de bus
- Arquitectura modular seccional

Ulrich K. et al, (2013) explican que: “Las arquitecturas modulares de bus y modulares seccionales son particularmente útiles para situaciones en las que el producto general debe variar mucho en configuración, pero cuyos trozos puedan interactuar en formas estándar con el resto del producto” (p.187).

Para el diseño del producto se podría implementar cualquiera de las dos arquitecturas, tomando en cuenta que una de las más sencillas es la arquitectura “modular de bus”, debido a que implementa un tronco común al que otros periféricos se conectan por medio del mismo tipo de interfaz.

Figura 4.

Arquitectura modular



Nota. Tomado de “Diseño y desarrollo de productos”, (p. 187), por (Ulrich K. &Eppinger S., 2013)

Los pasos recomendados por Ulrich (2013) para poder establecer la arquitectura del producto se basa en:

- Crear un esquema del producto.
- Agrupar los elementos del esquema.

- Crear una disposición geométrica aproximada.
- Identificar las interacciones fundamentales e incidentales.

Balanzas electrónicas y celdas de carga.

Debido a que es importante obtener el peso del objeto depositado al interior del locker, la implementación de una balanza interna es la base para empezar el diseño estructural del sistema.

En el escrito “Diseño y Construcción de un Sistema Automático de Pesaje y Control Estadístico Mediante Básculas Electrónicas Inteligentes”, se hace referencia a que las balanzas electrónicas, son más rápidas y por lo general más precisas que las mecánicas. También pueden incorporarse a sistemas computarizados, lo que las hace más útiles y eficaces que las balanzas mecánicas en la mayoría de las aplicaciones. (Hernández et al, 2018).

Al momento de incorporar sistemas computarizados, se puede recopilar de mejor manera los datos y ocuparlos de acuerdo a la necesidad que requiera el operario, a diferencia de que si se empleara sistemas mecánicos de pesaje.

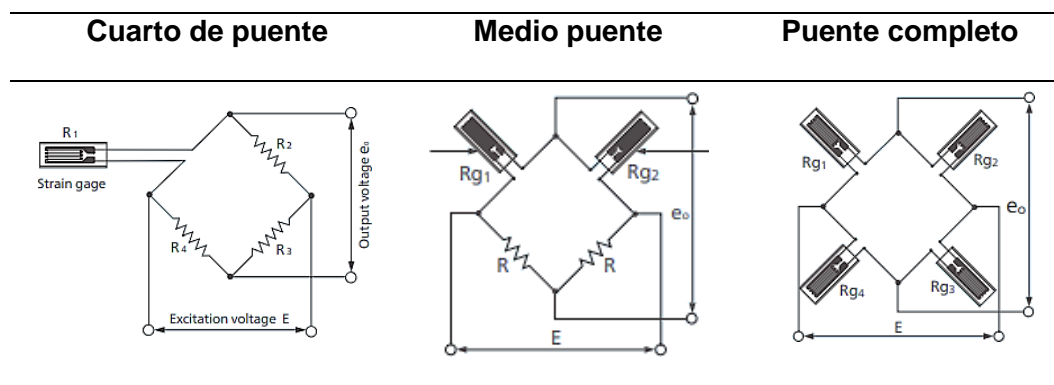
De igual manera, Hernández, S. (2018) manifiesta que “Requieren muy poco espacio, su colocación se adapta a la forma del recipiente, siendo posible utilizar desde una, hasta 4 celdas. El mantenimiento de las celdas de carga, indicadores y componentes en general es prácticamente nulo. No hay partes en movimiento, ni desgaste. “(p 15).

Siendo una ventaja considerable el hecho de que no requiere mantenimiento constante del sistema para que opere óptimamente, al igual que el espacio que necesita para su instalación es reducido.

Las galgas extensiométricas se pueden colocar en una, dos o cuatro ramas del puente de Wheatstone, dónde en el caso de emplearse una o dos galgas, se completará el resto del puente con resistencias de precisión.

Tabla 4.

Tipos de configuración de puente de Wheatstone



Nota. Tomado de technical Strain gages, (p. 4), por (kyowa-ei, 2022)

La mayor desventaja de completar el puente con resistencias es el hecho de que no se efectúa una auto compensación del mismo al momento de verse afectado por valores externos del ambiente como la temperatura; puesto que esta solo afecta al valor óhmico de la galga y por ende desequilibra el sistema.

El emplear cuatro galgas mejora la sensibilidad y la estabilidad del puente ante perturbaciones de factores externos al momento de medir, debido a que se forma el puente de Wheatstone solo con galgas.

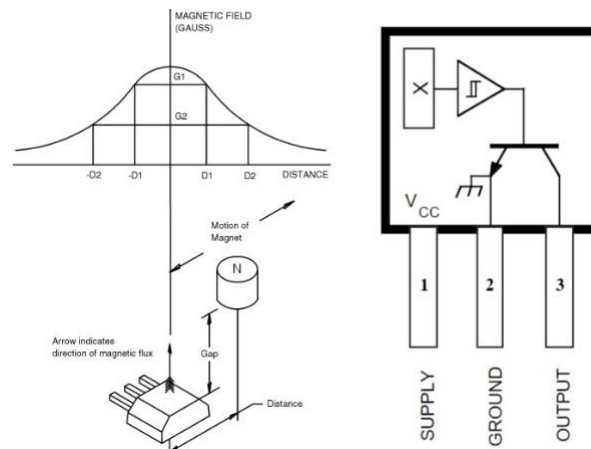
Sensores y actuadores.

Sensor Hall

El sensor Hall al estar elaborado por una fina lamina conductora, se caracteriza por reaccionar al flujo de corriente. Cuando se somete a un campo magnético, responde con un voltaje de salida proporcional a la fuerza del campo magnético; Al combinarlo con la electrónica asociada, forma un sensor de efecto Hall.

(Honeyweell,2022)

Los interruptores de efecto Hall son circuitos integrados monolíticos que mayormente se emplea para detección de campos magnéticos, aunque también se lo puede ocupar como componente principal para dispositivos de detección de corriente, temperatura, presión, posición, etc.

Figura 5.*Interruptores de efecto Hall*

Nota. Recuperado del datasheet 3141THRU3144

Cerraduras electromagnéticas

Una cerradura electromagnética al momento de activarse crea un campo magnético que atrae una placa hacia su estructura con la fuerza suficiente para asegurar la puerta. El principio detrás de una cerradura magnética es el uso de electromagnetismo para bloquear una puerta cuando está energizada.

(HVRMAG,2021)

Figura 6.*Cerradura electromagnética*

Cerraduras tipo solenoide

El principio de funcionamiento de la cerradura se basa en un émbolo y un bobinado; al momento de enviar corriente a través del bobinado se crea un campo electromagnético, permitiendo la retracción del émbolo y la liberación del seguro.

A diferencia de las cerraduras electromagnéticas, por lo general se encuentran en condiciones normalmente cerradas, la implementación de cerradura por solenoide en una puerta permite el bloqueo y cierre automático en caso de desconexión de cables o cortes de energía.

Figura 7.

Cerradura tipo solenoide



Acondicionador de señal

Debido a que la señal entregada por el puente de Wheatstone necesita acondicionamiento para poder trabajar con los datos sensados, se planteó el uso del módulo HX711.

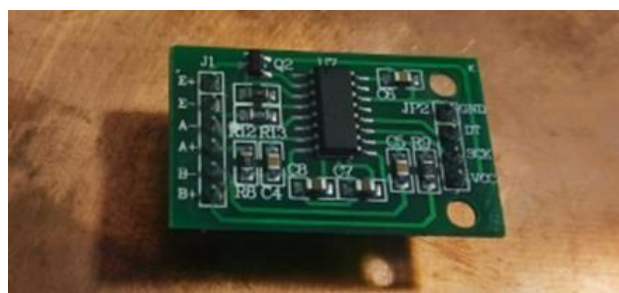
El multiplexor de entrada selecciona la entrada diferencial del canal A o B al amplificador de ganancia programable (PGA) de bajo ruido. (Digikey Electronics, 2022)

El módulo viene incorporado con un amplificador y solo requiere la conexión de las señales del puente, para posteriormente mediante software poder empezar con la calibración del sistema de pesaje.

De acuerdo al pdf del datasheet proporcionado por Digikey Electronics (2022) “El canal A se puede programar con una ganancia de 128 o 64, correspondiente a un voltaje de entrada diferencial de escala completa de $\pm 20\text{mV}$ o $\pm 40\text{mV}$ respectivamente, cuando un suministro de 5V está conectado al pin de la fuente de alimentación analógica AVDD.” (p 1)

Figura 8.

Módulo HX711



Protocolos de comunicación

Protocolo RS-485

El protocolo RS-485 fue desarrollado para poder expandir las capacidades físicas del RS-232; Weis, O. (2021) explica que “RS-485 (actualmente conocido como EIA/TIA-485) es una interfaz estándar de la capa física de comunicación, un método de transmisión de señales, el 1er nivel del modelo OSI (Interconexión de Sistemas Abiertos).”

Para este protocolo principalmente se emplea una conexión de 2 o 3 hilos, dependiendo la aplicación que se pretende desarrollar; siempre tomando en cuenta que se debe establecer:

- Un hilo de datos
- Un hilo de datos invertidos
- Un hilo de tierra

Con la finalidad de que los transmisores y los receptores, mediante un cable de par trenzado, intercambien los datos.

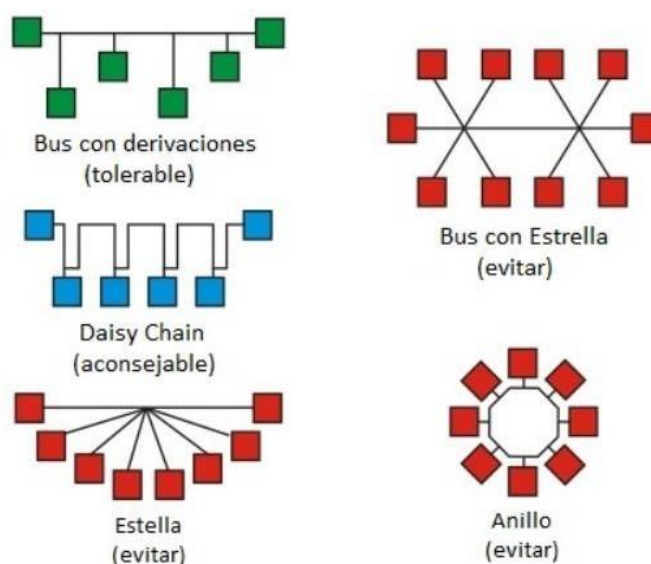
Una de las principales características y virtudes que posee este tipo de comunicación es el “multipunto”. Donde la línea de comunicación RS-485 puede funcionar con varios receptores y transceptores conectados. Es decir, se puede conectar un transmisor y varios receptores simultáneamente a la línea de comunicación. Todos los otros transmisores que precisen conectarse deberán esperar hasta que la línea de comunicación esté disponible para la transmisión de datos. (Weis, O., 2021)

Los tipos de topología para RS-485 al igual que para RS-422 no influyen en su rendimiento mientras la velocidad y las distancias sean relativamente bajas. Sin embargo, cuando los efectos de las líneas de transmisión empiezan a aparecer, existe apenas una topología simple para manejarlos. (NOVUS Automation, 2022)

Las topologías más empleadas, con diferentes niveles de recomendación al momento de implementar, para RS485 se muestran a continuación:

Figura 9.

Topología RS485 y RS422



Nota. Tomado de Conceptos Fundamentales De RS485 Y RS422, (p. 3), por (NOVUS Automation, 2022)

NOVUS (2022) considera la topología de bus como tolerable para su implementación, sin embargo, explica que: “Cuando se utiliza un bus con derivaciones, se recomienda que la longitud de las derivaciones que conectan cada dispositivo a la línea de comunicación principal sea lo más corto posible (más corto que la longitud del bus principal).” (p.3)

Al emplear este protocolo, es necesario conmutar estados del emisor, así como del receptor para evitar conflicto de datos, así como explica Weis, O. (2021):

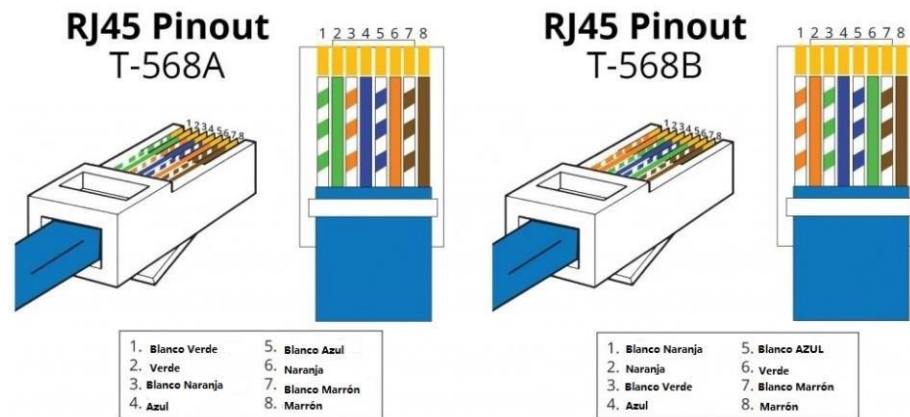
El servidor puede conmutar su controlador RS-485 al modo de transmisión por sí mismo, mientras que los otros controladores RS-485 (esclavos) trabajan en el modo de recepción. Para que un esclavo responda al maestro a través de la línea de comunicación, el 'maestro' debe enviarle un comando especial, que da al dispositivo en cuestión el derecho de cambiar su controlador al modo de transmisión durante un tiempo determinado.

Conexión con RJ-45.

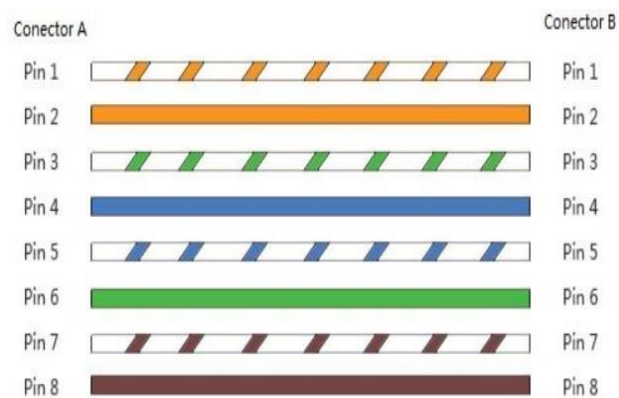
Debido a que la capa de comunicación posee un enlace físico compuesto por los cables y otros elementos de hardware, como conectores; es pertinente emplear un protocolo para el diseño de los mismos, así como explica Worton (2021):

La utilización de un cable directo o cruzado dependerá del tipo de conexión que se necesite. Para normalizar la disposición de cables, se utilizan dos estándares, el T568A y T568B, los cuales proporcionan esquemas de cableado para la terminación de los cables de red en enchufes, así como enchufes RJ45 de ocho posiciones.

La selección de protocolo también dependerá si se emplea conexión directa o cruzada en los terminales para la transmisión de datos.

Figura 10.*RJ-45 pin de salida*

Nota. Tomado de T568A y T568B: dos estándares de cable de red RJ45 (Worton, 2021)

Figura 11.*Esquema de color RJ-45 de cable directo*

Nota. Tomado de T568A y T568B: dos estándares de cable de red RJ-45 (Worton, 2021)

Capítulo III

Diseño del sistema

Diseño del sistema bajo RPA

RPA propone varios criterios para identificar un proceso que puede ser automatizado, los cuales evalúan la oportunidad de automatización en procesos repetitivos, para ello se planifican rutas secuenciales y tiempos de aplicación. La siguiente tabla muestra la evaluación por criterios de un sistema de recepción de paquetes.

Tabla 5.

Tabla de criterio de automatización para RPA

Criterio	Características	Valoración
Múltiples sistemas utilizados	El robot CSC puede acceder al subsistema de pesaje, al subsistema esclavo-maestro, a la pasarela de pagos y a la secuencia de HMI.	Si
Transacciones de alto volumen / valor	Un sistema de paquetería tradicional requiere de una o dos personas tomando en cuenta que se pueden generar errores en la facturación.	Si
Propenso a errores y re-trabajo	El proceso manual tiene una frecuencia y complejidad media, sin embargo, resulta en un gran número de errores.	Si
Alta predictibilidad	El proceso está definido de forma secuencial que es visible para el cliente mediante un HMI y lleva un registro de estado de los casilleros.	Si
Excepciones limitadas	El proceso se centra en automatizar la toma de datos del paquete y asegurarlo de forma inmediata.	Si

Criterio	Características	Valoración
Carga de trabajo manual significativa	El proceso tradicional es manual en la toma de datos del cliente, en su registro en una base de datos y en adquirir los datos del paquete. Para que el proceso se agilice se necesita aumentar la mano de obra.	Si

El servicio de paquetería puede ser optimizado con respecto a registro de datos del cliente y del paquete, además de automatizar el sistema de cobro gracias a una pasarela de pago. De esta manera, se define un proceso secuencial de toma de datos del paquete y facilita al cliente interactuar con el sistema gracias al uso de una interfaz intuitiva.

Tabla 6.

Tabla de automatización de procesos mediante RPA

Proceso	Antes	Después	Variable a controlar
Sistema de pesaje	Se requiere de una persona para la toma de datos del paquete	El sistema sensa automáticamente el paquete del usuario.	Peso
Pasarela de pagos	Una persona registra el paquete en una base de datos y realiza el cobro del envío.	El sistema muestra el valor de envío y debita directamente de la tarjeta.	Tarifa
Sistema de identificación del usuario	El personal llena los datos del usuario y del paquete en un formulario cada vez que ocupa el servicio	El servicio requiere validar los datos de envío mediante un código único.	Validación de datos

Proceso	Antes	Después	Variable a controlar
Seguridad	Se almacenan los paquetes en la zona de despacho.	El locker asegura los paquetes en su estructura de forma individual.	Sensor Hall y estado de la cerradura
Interacción	Se requiere de personal capacitado para brindar el servicio de paquetería	Se reduce el error humano al usar un sistema de envío a base de una interfaz intuitiva.	Tiempo de uso del HMI

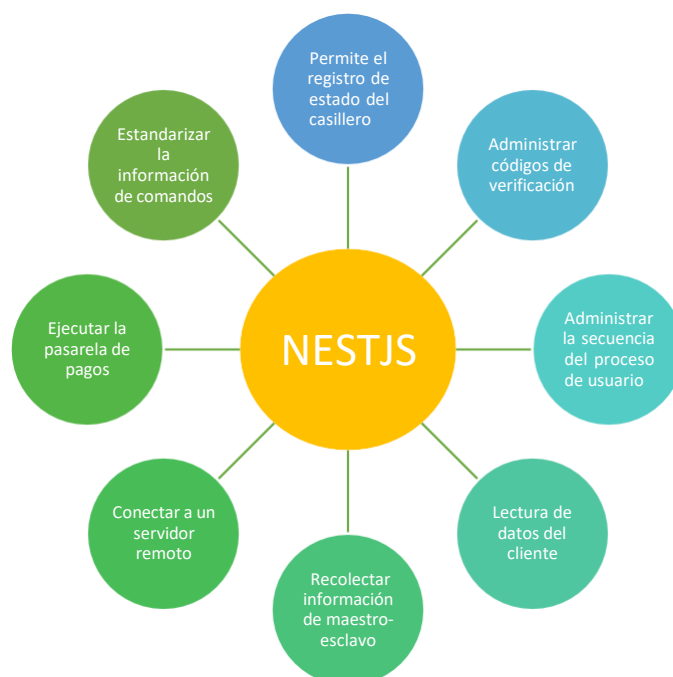
Al identificar los puntos que el sistema de paquetería puede automatizarse, se define que el servidor local de NestJS como Backend es un robot CSC, el alcance se explica en la figura 12:

El sistema es el conjunto de los diseños explicados anteriormente que trabajan de forma secuencial, en el centro se encuentra NestJS como un robot CSC que cumple con la funcionalidad de RPA, administrando los datos de la comunicación con el sistema maestro-esclavo que se envían entre el locker y el servidor remoto. Esta trama organiza la información de los estados del casillero y se registra su cambio en una base de datos por MQTT, mientras que el protocolo HTTP administra los códigos de verificación, ejecuta la pasarela de pagos y realiza la secuencia del proceso.

Debido a que el proceso de comunicación va a ser automatizado, necesariamente se debe controlar la apertura y cierre de los lockers, leer los datos del sistema de pesaje y llevar un registro de las acciones de las torres.

Figura 12.

Diagrama del alcance del servido local bajo RPA y CSC



Para el control de los sensores de pesaje y los sensores adicionales, se establece una comunicación con el controlador que administra los valores medidos y a su vez ejecuta comandos de control.

Tabla 7.

Criterios de selección

Criterios de selección	Peso	Optimización de la recepción del paquete usando galgas de un cuarto de puente con PIC (Programmable Integrated Circuit) y Raspberry		Optimización de la recepción del paquete usando galgas de precisión con PC como servidor		Optimización de la recepción del paquete usando una balanza de resorte con Raspberry	
		Calificación	Evaluación ponderada	Calificación	Evaluación ponderada	Calificación	Evaluación ponderada
Rango de medición amplio (galgas).	20%	5	1	3	0,6	2	0,4
Precisión en medición del peso	15%	4	0,6	5	0,75	3	0,45
Facilidad de uso	10%	2	0,2	3	0,3	4	0,4
Aspecto económico accesible	15%	5	0,75	1	0,15	3	0,45

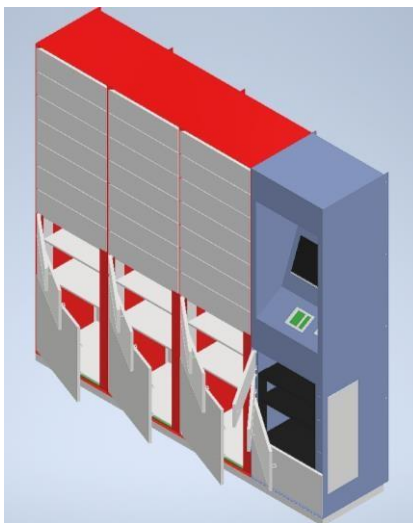
Criterios de selección	Peso	Optimización de la recepción del paquete usando galgas de un cuarto de puente con PIC (Programmable Integrated Circuit) y Raspberry		Optimización de la recepción del paquete usando galgas de precisión con PC como servidor		Optimización de la recepción del paquete usando una balanza de resorte con Raspberry	
		Calificación	Evaluación ponderada	Calificación	Evaluación ponderada	Calificación	Evaluación ponderada
Tiempo de computo	10%	4	0,4	3	0,3	2	0,2
Facilidad de control	15%	4	0,6	4	0,6	3	0,45
Flexibilidad de diseño para HMI	15%	5	0,75	3	0,45	3	0,45
Total, puntos		4,3		3,15		2,35	
Lugar		1		2		3	
¿Continuar?		Si		Revisar		No	

Para este punto se eligió un sistema basado en galgas extensiométricas (medio puente o precisión) controladas por PIC como unidad de control y recepción de datos, que a su vez estará administrada por una Raspberry.

Diseño modular

El objetivo de una arquitectura “modular de bus”, es implementar un tronco común al que otros periféricos se conectan por medio del mismo tipo de interfaz.

Para el desarrollo del sistema se tomó en cuenta que existirá un maestro (principal) y se le podrá anexas múltiples esclavos (secundarios); la cantidad de esclavos que se le anexe al sistema será directamente proporcional a la cantidad de lockers que requiera la empresa para solventar la necesidad de recepción de paquetes.

Figura 13.*Modularidad de lockers externa*

Nota. Expansión modular física que corresponde a una torre maestra (azul) y las torres esclavas (rojos)

Unión modular

La modularidad debe ser fácil de implementar, ofreciendo seguridad para cada torre, como para todo el sistema; por lo cual se planteó 2 tipos de uniones entre lockers:

Tabla 8.*Tipo de unión*

Tipo de unión	Accesibilidad	Costo	Facilidad de uso	Fuente de energía	Seguridad
Mecánica	Fácil	Baja	Alto	No	media
Electrónica	Difícil	Media	Alto	si	Alta

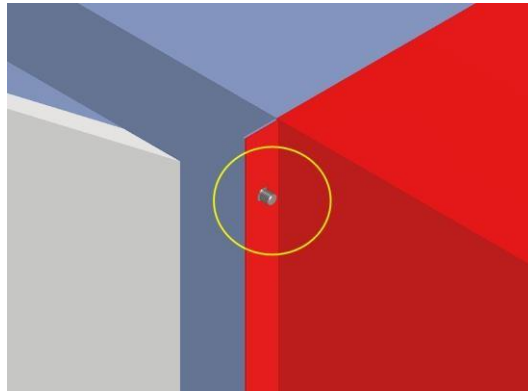
Nota. Especificaciones a tomar en cuenta para la modularidad mecánica.

Ya que en caso de que existiese un corte de luz, una seguridad electrónica no funcionaría y debido a que la seguridad mecánica es de fácil acceso; se optó por emplear las 2 para que se complementen entre sí y de esta manera poder obtener una modularidad altamente funcional.

Se dispondrá de una franja metálica trasera en los lockers para unirlos mecánicamente mediante pernos hexagonales.

Figura 14.

Unión mecánica

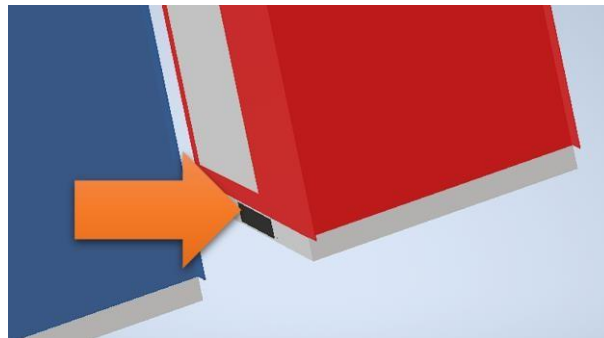


Nota. Ubicación de los orificios para la expansión modular

Y en la parte inferior se implementará una cerradura magnética que solo se podrá controlar mediante comandos de administrador.

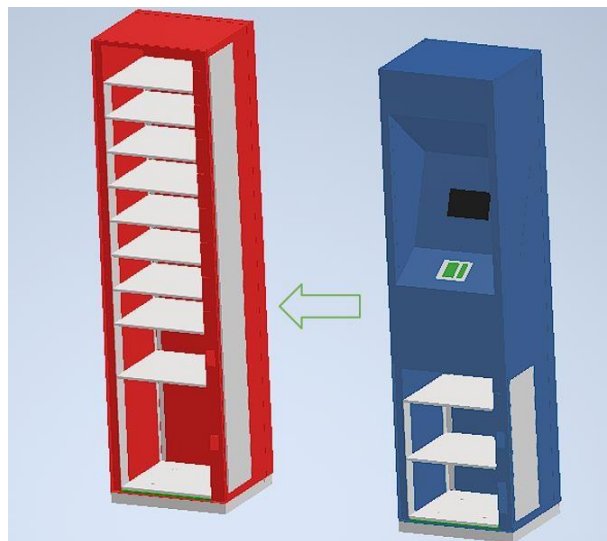
Figura 15.

Unión electrónica



Nota. Ubicación de la cerradura en la base rectangular

La expansión modular se lo realizará hacia la izquierda para que la puerta de mantenimiento de los lockers no quede expuesta al usuario.

Figura 16.*Expansión mecánica*

Nota. Prototipo de diseño esclavo- maestro

De esta manera se podrá incrementar las torres de manera segura y fácil. El diseño modular mecánico seleccionado condicionará al sistema electrónico, por lo cual la comunicación a emplearse se basará en una comunicación de BUS.

Comunicación modular

Dentro de las normativas físicas de comunicación en serie, las más empleadas son:

- RS-232
- RS-422
- RS-485

Los parámetros de selección para el protocolo fue la siguiente:

Tabla 9.*Protocolos de comunicación*

Protocolo	Peso	RS-232	RS-422	RS-485
Modo de trabajo diferencial	20%	No cumple	Si cumple	Si cumple

Protocolo	Peso	RS-232	RS-422	RS-485
Número de emisores	10%	1	1	32
Número de receptores	25%	1	Hasta 32	Hasta 32
Topología admisible	25%	Punto a unto	Multipunto y anillo	Bus, multipunto, entre otras
Transmisión	20%	Half dúplex, full dúplex	Full dúplex	Half dúplex
Total	100%	45%	80%	100%

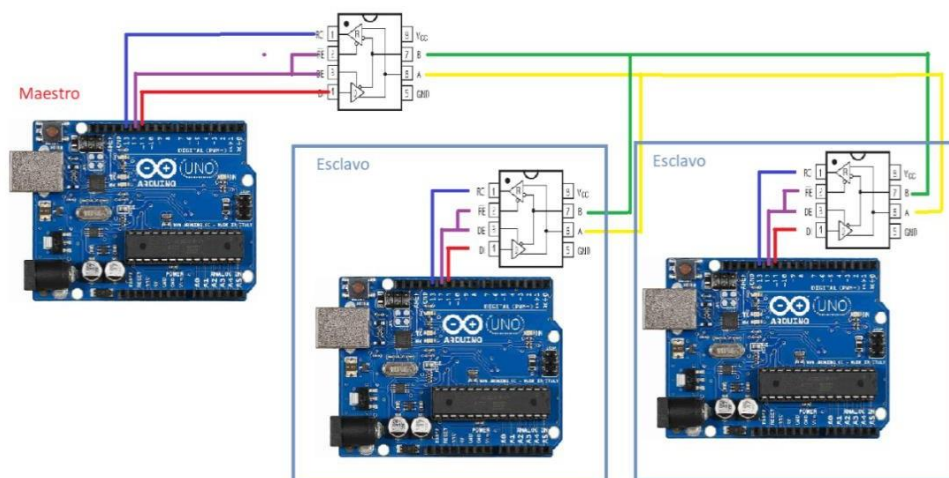
Nota. Selección del protocolo que más se adapte al proyecto

Debido a que el protocolo que más cumple las necesidades para el proyecto es el RS 485 con un porcentaje de 100% de aceptación a los criterios de selección, siendo el número de receptores y la topología admisible, las cualidades más relevantes.

Para generar la comunicación entre maestros y esclavos se empleó el encapsulado "MAX 485"; necesita de 1 puerto emisor del mensaje, 1 puerto para recibir el mensaje y 1 puerto para la gate que permitirá la comunicación como se muestra en la figura siguiente:

Figura 17.

Conexión maestro-esclavo



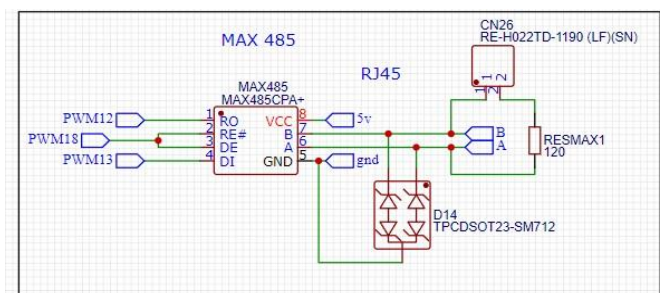
Nota. Representación de un maestro con dos esclavos empleando MAX 485

Se puede expandir a muchos más esclavos, tomando en cuenta que se deberá establecer la conexión con una resistencia entre A y B en el último esclavo para poder generar el final de comunicación.

Para el esquemático se optó por establecer dos pines machos que se los puenteará con un jumper al momento de establecer el último esclavo y conectará la resistencia de 120 ohm recomendada para el circuito

Figura 18.

Circuito de conexión para MAX 485 con delimitador



La placa "maestro" emitirá el mensaje a todas las placas esclavo, pero solo una debe responder, para lo cual se consideró necesario generar un ID a cada uno de los esclavos, conservando la modularidad al mismo tiempo.

Siendo la implementación de un Dip switch uno de los métodos para poder identificar a cada esclavo con un ID diferente.

Debido a que el voltaje de entrada para el Dip switch es de 5V DC suministrado por la Placa, se procede a calcular la resistencia mínima para cada uno de los puertos

$$V = I * R$$

Donde:

V = Voltaje

R = Resistencia

Rt = Resistencia

I = Intensidad

Ecuación 1.

Ley de Ohm

Nota. tomado de "Circuitos eléctricos", por Nilson, J. & Riedel, S.

$$\frac{5}{0.1} = Rt$$

$$Rt = \frac{1}{\frac{1}{R} + \frac{1}{R} + \frac{1}{R} + \frac{1}{R}}$$

Ecuación 2.

Suma de resistencias en paralelo

Nota. Tomado de "Circuitos eléctricos", por Nilson, J. & Riedel, S.

$$Rt = \frac{R}{4}$$

$$\frac{4 * 5}{0.1} = R$$

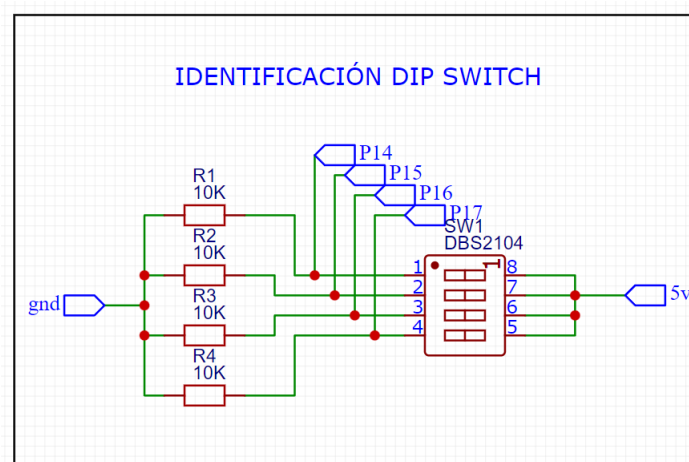
$$200 = R$$

La resistencia mínima que se debería ocupar es de 200 ohms para el divisor de voltaje que requiere la entrada digital al momento de leer si existe un pull up o un pull

down; en este caso la resistencia recomendada para la lectura es de 10K ohm, por lo que cumple con la condición >200 ohm.

Figura 19.

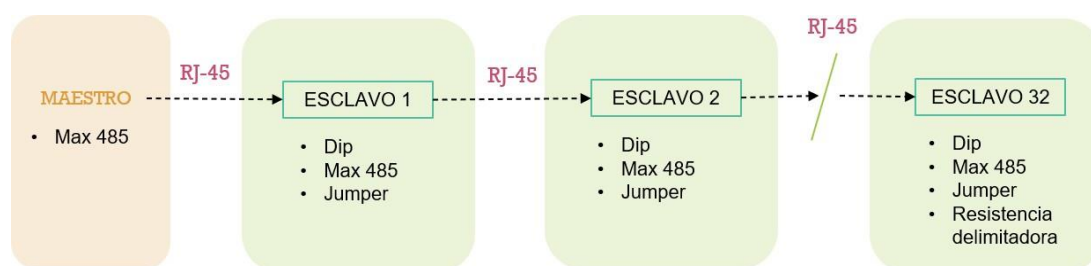
Conexión para lectura de pines del Dip switch



Debido a que la conexión entre placas no requiere mucha distancia y debe ser modular, se optó por implementar puertos RJ45 que permitan un sistema “plug and play” mediante la comunicación por cable UTP RJ-45 basado en el código de color T568 B de cable directo, ya que es la más usada.

Figura 20.

Esquema modular electrónico



Diseño mecánico

Diseño del sistema de pesaje

En la actualidad el sistema de pesaje se puede realizar con varias galgas a la vez, con la finalidad de admitir diferentes cargas nominales, lo que amplía de forma significativa sus aplicaciones en el ámbito alimenticio, de paquetería, clasificación, etc.

El modelo base que se tomó para el diseño, debe cumplir con los requerimientos del sistema de pesaje, que son:

- Área de pesaje mediana o grande
- Medición por gramos
- Funcionamiento electrónico
- Encerado automático

Para la implementación de este sistema de pesaje es necesario seleccionar el tipo de sensor a ocupar; por lo que se toman en cuenta el peso máximo que se desea sensor, la precisión que se tolera y la facilidad de su mantenimiento, por lo tanto, se plantea:

Tabla 10.

Selección de sensor para el sistema de pesaje

Tipo de galga extensiométrica	Peso	Precableadas	Precisión	Calidad
Rango de medición	30%	50 kg	5 kg	30 kg
Precisión	20%	Media	Alta	Alta
Costo	10%	Baja	Alta	Alta
Facilidad de uso	20%	Medio	Alto	Alto
Robustez	20%	Alta	Medio	Alta
Total	100%	80%	20%	70%

Se eligió las galgas extensiométricas precableadas debido a que cumplen con un 80% de los requerimientos de diseño, considerando el rango de medición y la facilidad de uso como los puntos más fuertes de selección.

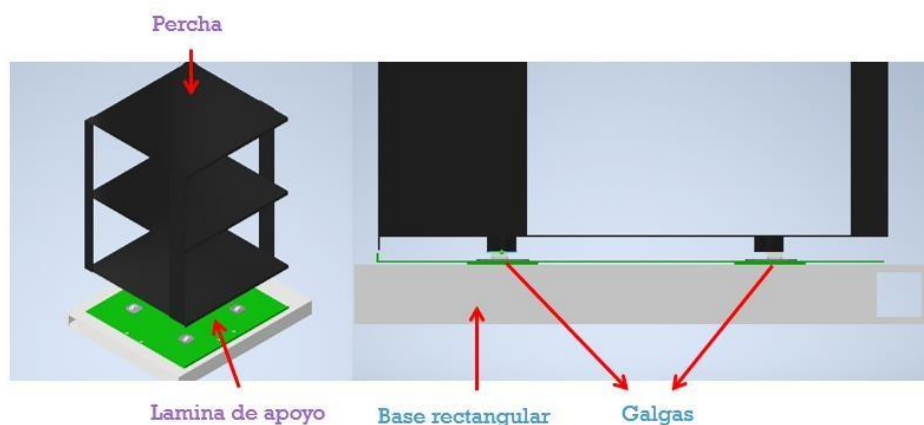
Este tipo de galgas no tienen pivotes, piezas móviles y producen una corriente eléctrica que pueden utilizarse para fines de control y registros directos de datos.

Debido a que se emplea galgas de un cuarto de puente, la implementación óptima es

por pares para que se complementen entre sí y permitan más rango de medición, como se muestra en la tabla 4.

Figura 21.

Diseño del subsistema de pesaje



Nota. El diseño propuesto comprende a la percha, lámina de apoyo, base rectangular y galgas extensiométricas.

Para poder obtener el peso de cada paquete depositado en la torre, se diseñó un sistema a manera de una percha, ubicada encima de los sensores, colocados en una base rectangular como se muestra en la figura 21.

Selección de material

La selección del material que se empleará en la estructura fue a base de los siguientes criterios:

Tabla 11.

Criterios de selección de material.

Material	Peso	Plancha laminada en caliente	Planchas laminadas en frío	Planchas anti abrasivas	Planchas galvanizadas	Planchas inoxidables
Esfuerzo máximo	30%	270 Mpa	310 Mpa	160 MPa	310 MPa	310 Mpa
Pintado de superficies	10%	Difícil	Fácil	Medio	Medio	Difícil

Material	Peso	Plancha laminada en caliente	Planchas laminadas en frío	Planchas anti abrasivas	Planchas galvanizadas	Planchas inoxidables
Resistencia a la oxidación	25%	Alta	Bajo	Alto	Alta	Alta
Deslizante	15%	Rugosa	Liso	Rugoso	Liso	Liso
Costo beneficio	20%	Bajo	Bajo	Medio	Alta	Medio
Total	100%	25%	55%	25%	90%	70%

Se eligió la plancha galvanizada con un 90% de aprobación debido a su alta resistencia a la oxidación y su alto valor en costo beneficio, a pesar de que no tiene un pintado de superficies sencillo, se compensa con su capa deslizante que es resistente a la abrasión y a daños mecánicos, además el material dura 30 años sin mantenimiento en condiciones normales.

Cálculos

Dimensionamiento de las columnas para la carga

$$\# \text{ pisos} = 3$$

$$d_{\text{pisos}} = 220 \text{ mm}$$

$$\text{carga en cada piso } (C_{cp}) = 2.5 \text{ kg}$$

$$\text{Area de la plancha por piso } (A_p) = 358 \times 340 \text{ mm}$$

Carga viva por piso (Cv):

$$C_v = \frac{C_{cp}}{A_p}$$

Ecuación 3.

Carga viva por piso

Nota. Tomado de Norma Ecuatoriana de la Construcción (2013). Capítulo 15

$$C_v = \frac{2.5Kg}{0.12172m^2}$$

$$C_v = 20 \frac{Kg}{m^2}$$

Carga viva total (Cvt):

$$C_{vt} = 20 \frac{Kg}{m^2} * 3$$

Ecuación 4.

Carga viva total

Nota. Tomado de Norma Ecuatoriana de la Construcción (2013). Capítulo 15

$$C_{vt} = 60 \frac{Kg}{m^2}$$

Factor de carga de acuerdo a la NEC

$$C_u = 1.2C_m + 1.6C_{vt}$$

Ecuación 5.

Factor de carga

Nota. Tomado de Norma Ecuatoriana de la Construcción (2013). Capítulo 15

Donde:

Cu = Factor de carga

Cm = Carga muerta

Cvt = Carga viva total

Se considera despreciable la carga muerta debido a que el peso de las planchas es pequeño

$$C_u = 1.6 * (60 \frac{Kg}{m^2})$$

$$C_u = 96 \frac{Kg}{m^2}$$

Ocupando un factor de seguridad de $N = 3$ se obtiene el factor de carga final

(Cuc) por plancha

$$Cuc = Cu * N$$

Ecuación 6.

Factor de carga final

Nota. Tomado de Norma Ecuatoriana de la Construcción (2013). Capítulo 15

$$Cuc = 96 \frac{Kg}{m^2} * 3$$

$$Cuc = 288 \frac{Kg}{m^2}$$

El área cooperante (At) es el espacio que va a soportar las columnas, que es la misma área de la plancha por piso (Ap), esto se debe a que las columnas (platinas) poseen un área muy pequeña.

$$Ap = At$$

Ecuación 7.

Área cooperante

Nota. Tomado de Norma Ecuatoriana de la Construcción (2013). Capítulo 15

$$At = 0.2172m^2$$

Resistencia requerida (Pu):

$$Pu = Cuc * At * \# \text{ pisos}$$

Ecuación 8.

Resistencia requerida

Nota. Tomado de Norma Ecuatoriana de la Construcción (2013). Capítulo 15

$$Pu = 288 \frac{Kg}{m^2} * 0.2172m^2 * 3$$

$$Pu = 187.66kg$$

Área de la columna (Ag)

$$Ag = \frac{Pu}{Sy}$$

Ecuación 9.

Área de la columna

Nota. Tomado de Norma Ecuatoriana de la Construcción (2013). Capítulo 15

Material empleado ASTM A36 con resistencia a la fluencia (S_y) de:

$$S_y = 2528.9 \frac{kg}{cm^2}$$

$$A_g = \frac{187.66kg}{2528.9 \frac{kg}{cm^2}}$$

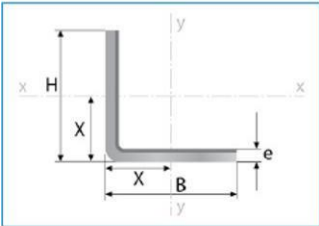
$$A_g = 0.0742cm^2$$

Área recomendada $> 0.0742cm^2$

Platinas existentes de acuerdo al catálogo de Ambatol.

Figura 22.

Platina de perfil en L.



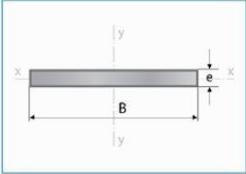
Designaciones	Esp.	Peso	Propiedades Estáticas									
			Eje x-x			Eje y-y			Distancia del eje y-y superior			
			Momento de inercia	Módulo de resistencia	Radio de giro	Momento de inercia	Módulo de resistencia	Radio de giro				
H	B	e	P	P	I	W	i	I	W	i	X=Y	
mm	mm	mm	kg/m	kg/6m	cm ⁴	cm ³	cm	cm ⁴	cm ³	cm	cm	
	20	20	1,50	0,44	2,61	0,21	0,15	0,47	0,21	0,15	0,47	0,54
			2,00	0,58	3,46	0,28	0,20	0,62	0,28	0,20	0,62	0,59
			3,00	0,83	4,96	0,38	0,28	0,60	0,38	0,28	0,60	0,64
	25	25	1,50	0,55	3,32	0,42	0,24	0,59	0,42	0,24	0,59	0,67
			2,00	0,73	4,38	0,56	0,32	0,78	0,56	0,32	0,78	0,72
			3,00	1,06	6,36	0,79	0,45	0,76	0,79	0,45	0,76	0,77
	30	30	1,50	0,67	4,03	0,75	0,35	0,71	0,75	0,35	0,71	0,79
			2,00	0,89	5,34	1,00	0,46	0,94	1,00	0,46	0,94	0,84
			3,00	1,30	7,78	1,41	0,67	0,92	1,41	0,67	0,92	0,89
	40	40	1,50	0,91	5,44	1,83	0,63	0,95	1,83	0,63	0,95	1,04
			2,00	1,21	7,23	2,44	0,84	1,26	2,44	0,84	1,26	1,09
			3,00	1,77	10,61	3,50	1,22	1,25	3,50	1,22	1,25	1,14
			4,00	2,31	13,83	4,46	1,58	1,23	4,46	1,58	1,23	1,19

Largo Normal:
6 metros
Recubrimiento:
Negro o Galvanizado
Calidad de Acero:
ASTM A36 / ASTM A572 Gr 50 /
ASTM A653
Norma de Fabricación:
NTE INEN 1623
Espesores:
Desde 1,50 a 6,00 mm
Observaciones:
Otras dimensiones y largos,
previa consulta

$$A_1 = 1.41cm^2$$

Figura 23.

Platina de perfil rectangular.

	Dimensiones	Espesor	Peso
	B	e	P
	Pulg.	mm	kg/6m
1/2 x 1/8	3,00	1,70	
1/2 x 3/16	4,00	2,26	
1/2 x 1/4	6,00	3,39	
3/4 x 1/8	3,00	2,68	
3/4 x 3/16	4,00	3,58	
3/4 x 1/4	6,00	5,37	
1 x 1/8	3,00	3,53	
1 x 3/16	4,00	4,71	
1 x 1/4	6,00	7,07	
1 x 3/8	9,50	10,59	
1 x 1/2	12,00	14,13	
1 1/4 x 1/8	3,00	4,24	
1 1/4 x 3/16	4,00	5,65	
1 1/4 x 1/4	6,00	8,47	
1 1/4 x 3/8	9,50	12,71	
1 1/2 x 1/8	3,00	5,37	
1 1/2 x 3/16	4,00	7,16	
1 1/2 x 1/4	6,00	10,74	
1 1/2 x 3/8	8,00	16,11	
1 1/2 x 1/2	12,00	21,48	
2 x 1/8	3,00	7,07	
2 x 3/16	4,00	9,50	

Largo normal: 6 metros
Recubrimiento: Negro
Espesores: Desde 3,00 a 12,00 mm
Calidad de Acero: ASTM A36 / ASTM A 572 Gr 50
Norma de Fabricación: INEN 2222 / ISO 1035 / INEN 2215 / DIN 1017
Observaciones: Otras dimensiones y largos previa consulta.

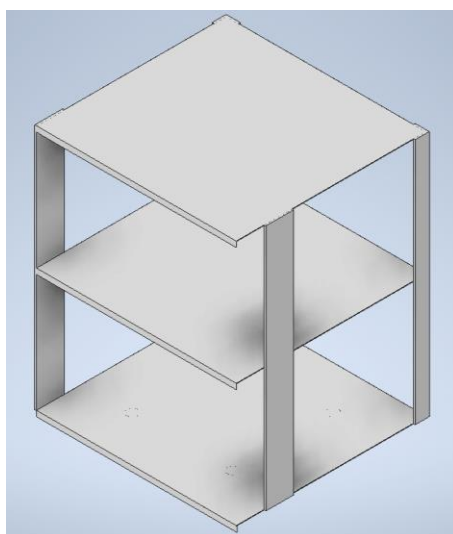
Aplicaciones
 - Somier camas.
 - Cerrajería en general (puertas, ventanas, rejas).
 - Estructuras en general.
 - Muebles metálicos.

$$A2 = 1.5 \text{ cm}^2$$

Se puede emplear cualquiera de los dos perfiles, pues ambos están disponibles en Ambatol y además superan el área mínima recomendada.

Figura 24.

Diseño de percha interna



Nota. Se diseñó mediante software la percha de 3 pisos empleando los perfiles seleccionados

Comprobación del factor de seguridad empleando la platina rectangular

Debido a que el primer prototipo va a ser de 3 pisos, es necesario comprobar si los materiales seleccionados soportan hasta por lo menos 8 pisos.

Área de la columna: Aplicando la ecuación 9:

$$A_g = \frac{P_u}{S_y}$$

Empleando un área de

$$A_g = 1.5 \text{ cm}^2$$

$$A_g * S_y = P_u$$

$$P_u = 2528.9 \frac{\text{kg}}{\text{cm}^2} * 1.5 \text{ cm}^2$$

Se calcula la resistencia requerida:

$$P_u = 3793.35 \text{ kg}$$

Planteando 8 pisos con una distancia igual entre ellos y con la misma área de la placa se aplica la ecuación 8:

$$C_{uc} = \frac{P_u}{A_t * \# \text{ pisos}}$$

$$C_{uc} = \frac{3793.35 \text{ kg}}{0.2172 \text{ m}^2 * 8}$$

$$C_{uc} = 2183.1 \frac{\text{Kg}}{\text{m}^2}$$

Tomando en cuenta que la carga viva por piso máximo va a ser de

$$C_v = 20 \frac{\text{Kg}}{\text{m}^2}$$

Y debido a que se planteó 8 pisos, se calcula el factor de carga

$$C_u = 160 \frac{\text{Kg}}{\text{m}^2} * 1.6$$

$$C_u = 256 \frac{\text{Kg}}{\text{m}^2}$$

El factor de seguridad debe ser mayor o igual al planteado, se usa la ecuación

6.

$$Cuc = Cu * N$$

$$N = \frac{2183.1 \frac{Kg}{m^2}}{256 \frac{Kg}{m^2}}$$

$$N = 8.52$$

El cálculo del área de la platina considera un factor de seguridad de 3, sin embargo, al momento de recalcular el factor de seguridad en función al área de la platina adquirida acorde al catálogo comercial se obtiene un valor superior a 8. Por lo tanto, se comprueba que las platinas seleccionadas pueden emplearse en una estructura de hasta 8 pisos.

Cálculo estructural del sistema de pesaje

El diseño planteado para la base del soporte se lo realizó con la finalidad de almacenar 4 galgas, están dispuestas de manera uniforme para un mejor soporte y distribución del peso total, tanto de la percha, así como de los objetos depositados en cada uno de los pisos.

Para el cálculo se analizó el primer prototipo de percha con 3 pisos, en cada uno se estima un paquete de 2 kg dando como resultado final 6 kg, además, se suma el peso de la percha estimado en 2 kg. El peso total de la percha con paquetes es de 8 kg, se divide para los 4 puntos de apoyo obteniendo:

$$\text{Masa en cada pata} = 2kg$$

Se calcula el peso de cada pata en función de la masa planteada:

$$w = 2kg * 9.81 \frac{m}{s^2}$$

Ecuación 10.

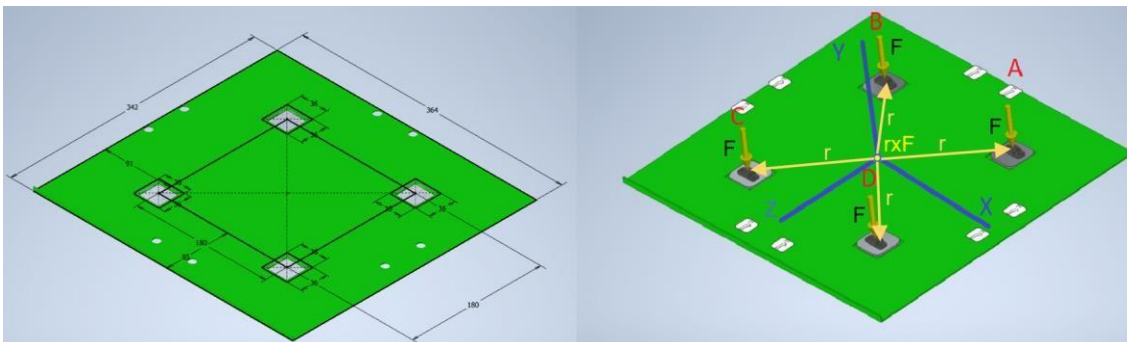
Ecuación de la fuerza de peso

Nota. Tomado de "Diseño de elementos de máquina ", por Robert, L. Mott

$$w = 19.6N$$

Figura 25.

Diseño de lámina de apoyo



Nota. La lámina se encontrará ubicada en la base y contendrá las galgas.

Debido a que las cuatro galgas se encuentran distribuidas en el interior de la placa de manera uniforme, la distribución de los vectores posición (r) está dado por los elementos A, B, C, D, en los ejes X y Z, en los mismos puntos, el vector fuerza (F) está aplicado hacia la lámina de apoyo en el eje Y. Este análisis ayuda en el cálculo del momento de torsión respecto al centro de la placa (rxF), para posteriormente analizar la simulación y verificar si la estructura satisface las necesidades de diseño.

Tabla 12.

Fuerzas aplicadas en la placa base

F	r [mm]	F [N]	rxF [kg.mm]
A	$90i-90k$	$-19.6j$	$-1764i-1764k$
B	$-90i-90k$	$-19.6j$	$-1764i+1764k$
C	$-90i+90k$	$-19.6j$	$1764i-1764k$
D	$90i+90k$	$-19.6j$	$1764i+1764k$
Total (Σ)		$-78.4j$	$0i+0k$

Nota. El momento de torsión con respecto al centro de la lámina de apoyo es nulo.

Tomado de "Mecánica vectorial para Ingenieros", por Beer, F. & Johnston, E.

Calculando el área de la lámina de apoyo

$$A = 0.364 \times 0.342m$$

$$A = 0.1245m^2$$

Calculando el esfuerzo máximo en el centro de la lámina de apoyo

$$\sigma_0 = \frac{F_0}{A}$$

Ecuación 11.

Ecuación del esfuerzo máximo en el centro de una lámina

Nota. Tomado de "Diseño de elementos de máquina ", por Robert, L. Mott

$$\sigma_0 = \frac{-78.4N}{0.1245m^2}$$

$$\sigma_0 = -632.25Pa$$

Una vez obtenido el esfuerzo máximo en el centro de la lámina de apoyo, mediante software se somete a un análisis estructural para comprobar la fiabilidad del diseño y poder visualizar de mejor manera las posibles deformaciones en toda la placa.

Según la normativa NTE INEN-OIML R 76-1 (2013) Fuente, OIML R 76-1:2006, específicamente en el punto 3.5, se define que los errores máximos permitidos dependen del rango de funcionamiento de la balanza. El prototipo tiene una precisión medida en gramos ($d= 1 \text{ g}$) y al no emplear un instrumento indicador auxiliar, la división para la escala de verificación puede definirse como $e=d$ en la siguiente tabla:

Figura 26.

Criterios de selección de balanzas.

Clase de exactitud	División de escala de verificación, e	Número de divisiones de escala de verificación, $n = \text{Max}/e$		Capacidad mínima Min (Límite inferior)
		mínimo	máximo	
Especial (I)	$0.001 \text{ g} \leq e^*$	50 000**	-	100 e
Alta (II)	$0.001 \text{ g} \leq e \leq 0.05 \text{ g}$ $0.1 \text{ g} \leq e$	100 5 000	100 000 100 000	20 e 50 e
Media III	$0.1 \text{ g} \leq e \leq 2 \text{ g}$	100	10 000	20 e
	$5 \text{ g} \leq e$	500	10 000	20 e
Ordinaria (III)	$5 \text{ g} \leq e$	100	1 000	10 e

* Normalmente, no es posible ensayar y verificar un instrumento con $e < 1 \text{ mg}$, debido a la incertidumbre de las cargas de ensayo.

** Véase la excepción de 3.4.4.

Nota. “d” es la unidad de medida más significativa y “e” es la división de escala de verificación, *Recuperado* de la normativa NTE INEN-OIML R 76-1 (2013) Fuente, OIML R 76-1:2006 (p 3).

El proceso de pesaje funciona en un rango de pruebas por gramo. Por lo tanto, se puede catalogar como una balanza de clase III y según el punto 3.5.2. Bajo la configuración de 4 galgas, se rige bajo la normativa de INSTRUMENTOS DE PESAR NO AUTOMÁTICOS. PARTE 1: REQUISITOS METROLÓGICOS Y TÉCNICOS. Las unidades de masa para las cargas de calibración pueden ser expresadas en gramos (g) según el punto 2.1. de la normativa NTE INEN-OIML R 76-1 (2013) Fuente, OIML R 76-1:2006.

Para el pesaje, se ejecuta una sola vez el programa de tara y se define el valor cero de medición, es decir que se encera tomando en cuenta el peso de la percha y de los paquetes que esta contenga. Según el punto 4.14.4 de NTE INEN-OIML R 76-1 (2013) Fuente, OIML R 76-1:2006 para dispositivos de preselección de tara: “no será posible hacer funcionar un dispositivo de preselección de tara si está en uso de un dispositivo de tara”, por lo tanto, el proceso de tara toma en cuenta a la percha una sola vez para el cálculo del factor de tara usando un peso conocido y posteriormente encera el sistema cada vez que se registra un peso.

Para el proceso de encerar el sistema se aplica la normativa del punto 4.5.6 según NTE INEN-OIML R 76-1 (2013) Fuente, OIML R 76-1:2006.- “el dispositivo de ajuste automático en cero funcionará cuando el equilibrio sea estable o permanezca estable por debajo de cero durante 5 segundos”.

Simulaciones del sistema de pesaje

Figura 27.

Simulación de la deformación máxima en la percha

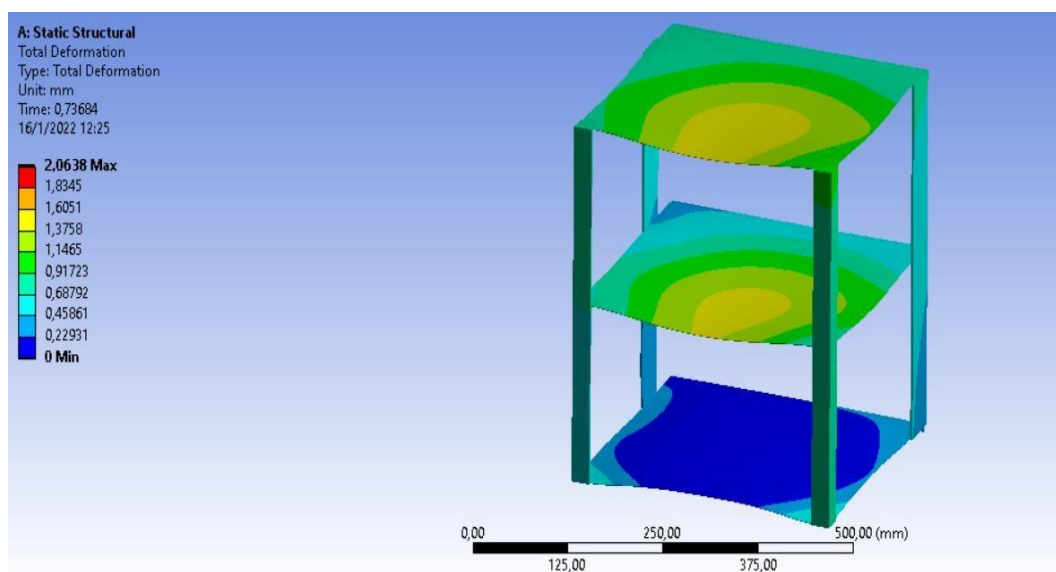
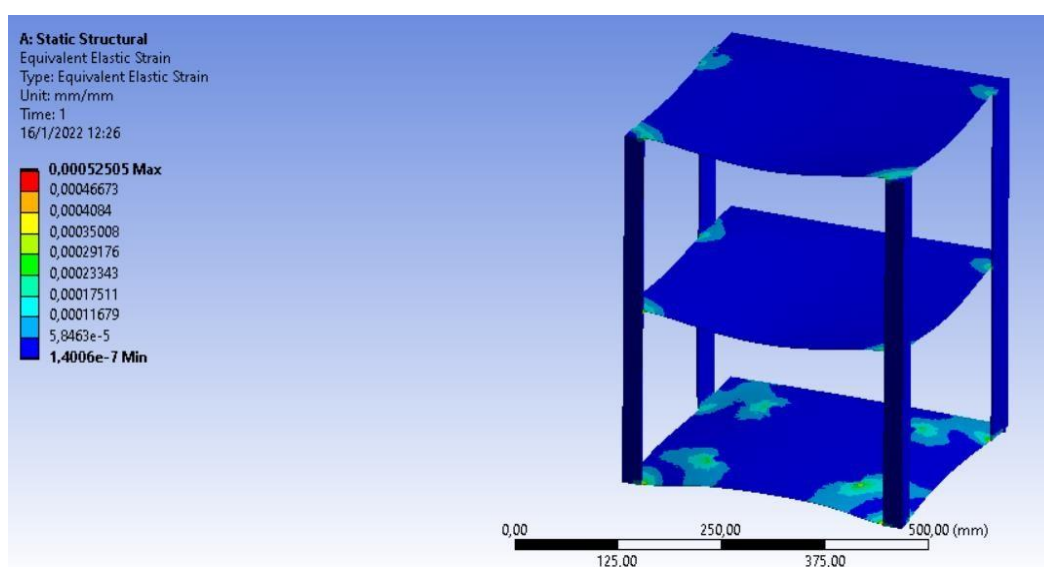


Figura 28.

Simulación de esfuerzo máxima en la percha



Mediante el análisis estructural de la percha por el método de elementos finitos, se obtuvieron los resultados de equivalente estático y deformación total de la estructura, ratificando el correcto diseño para una percha de tres pisos, teniendo como resultado una deformación aceptable para la recepción de paquetes.

Figura 29.

Simulación de la deformación con cargas puntuales

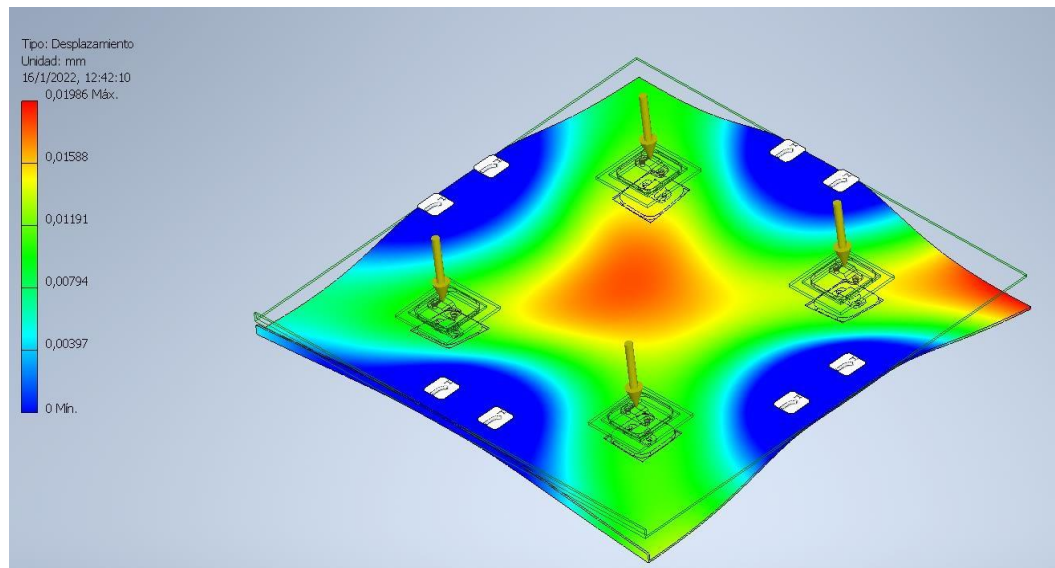
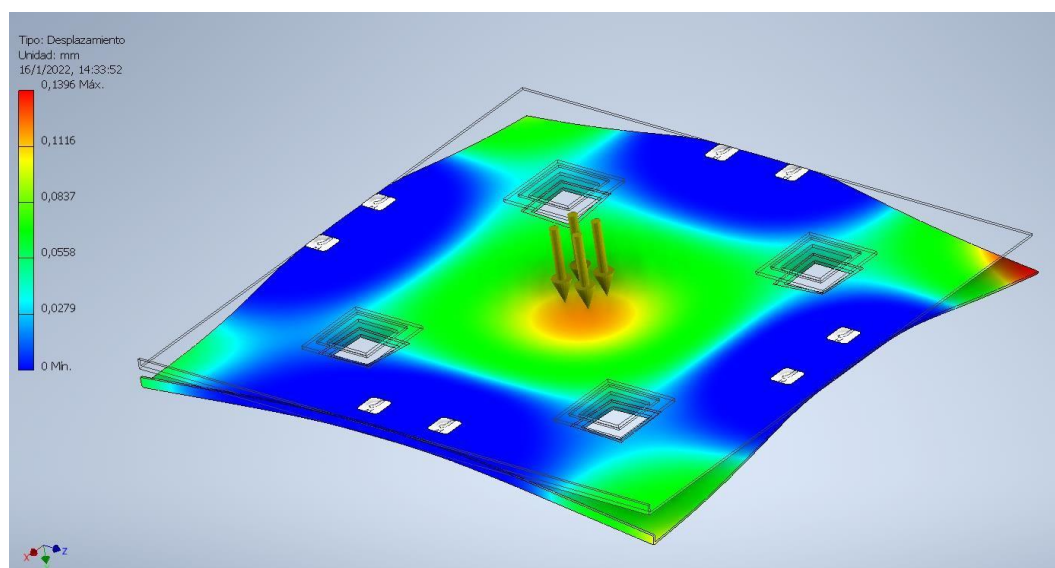


Figura 30.

Simulación de la deformación por esfuerzo en la placa



El análisis de deformación muestra como resultado que, pese a que la sumatoria de fuerzas en el centro de la lámina es nula, se genera un esfuerzo que provoca una deformación máxima de 0.1396 mm, calificándolo como aceptable para soportar la estructura de la percha y los paquetes.

Figura 31.

Simulación con cargas puntuales por Von Mises

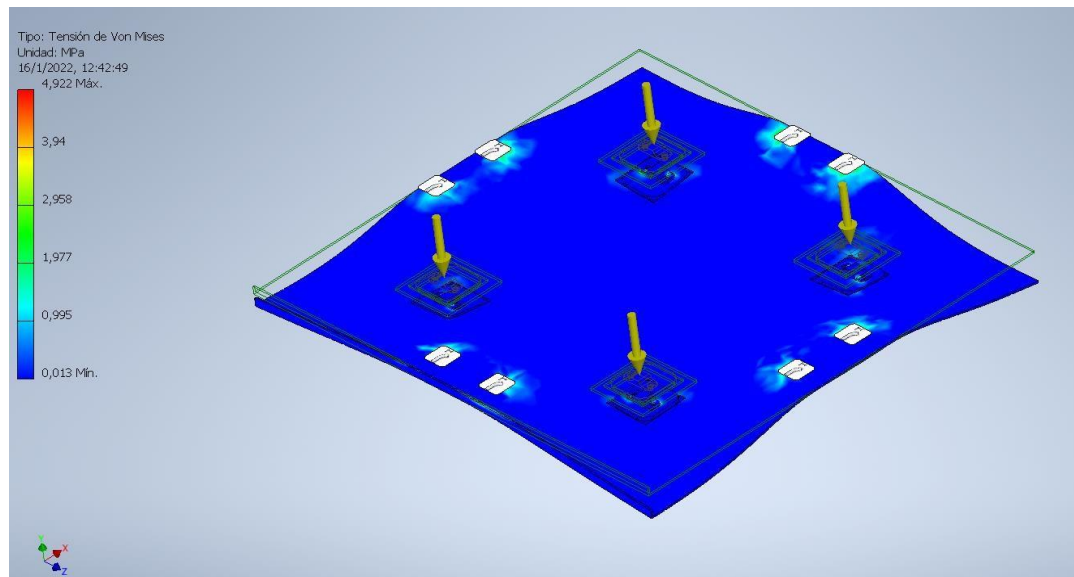
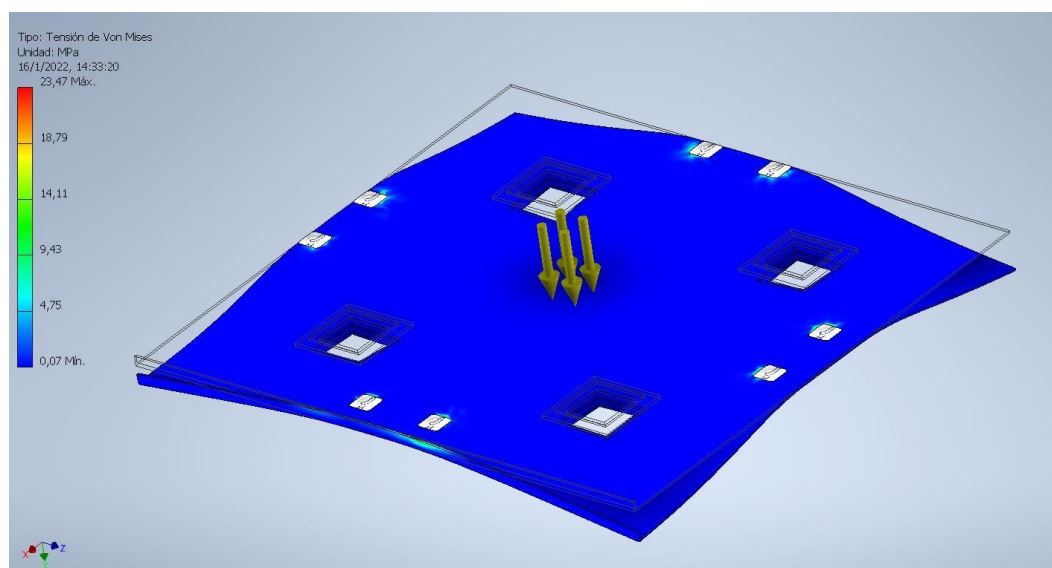


Figura 32.

Simulación del esfuerzo en la placa por Von Mises



Mediante el análisis por simulación empleando el método de Von Mises, se comprobó que la tensión producida por las cargas de la percha y de los paquetes llega a un valor máximo de 23,47 Mpa, el acero galvanizado tiene como límite elástico de 310 Mpa, siendo superior al valor máximo que genera la carga depositada sobre la lámina.

En las simulaciones realizadas se puede verificar que los materiales empleados para el diseño estructural son lo suficientemente resistentes para soportar las cargas máximas admisibles.

Diseño electrónico

Debido a que el proceso va a ser automatizado, necesariamente se debe controlar la apertura y cierre de los lockers para poder corroborar la recepción del paquete, siendo la seguridad uno de los puntos que más se debe tomar en cuenta para el diseño.

Selección de componentes para la seguridad electrónica

A comparación con las cerraduras mecánicas, las cerraduras magnéticas son más resistentes a golpes u otro tipo de daño, por lo que no se pueden forzar para abrirlas y no disponen de estructuras mecánicas que puedan romperse para vulnerar la seguridad, existiendo en el mercado diferentes tamaños de cerraduras dependiendo de la necesidad.

Tabla 13.

Especificaciones de las cerraduras magnéticas

Características	Peso	Tipo de cerraduras magnéticas		
		Pequeña (Solenoides)	mediana	grande
Fuerza de cierre	20%	20 kg	60 kg	180 kg
Facilidad de uso	20%	Alto	Alto	Alto
Robustez	20%	Medio	Alta	Alta
Retorno mecánico	5%	si	no	no

Características	Peso	Tipo de cerraduras magnéticas		
		Pequeña (Solenoides)	mediana	grande
Consumo (A) a 12V	15%	0.3	0.4	0.8
Posición inicial	20%	Normalmente cerrado	Normalmente abierto	Normalmente abierto
Total	100%	75%	65%	65%

El criterio clave que selecciona del 75% a las cerraduras de solenoide, es la posición inicial, pues en caso de existir un corte de luz, las puertas con cerradura pequeña permanecerán con seguridad, mientras que las puertas con cerraduras medianas y grandes requerirán de una fuente de energía extra con un consumo de amperaje alto para mantener asegurado los paquetes. El diseño poseerá mayor cantidad de cerraduras para paquetes pequeños.

Debido a que las cerraduras funcionan a 12 V se consideró fundamental el desacople de la parte de control con la de potencia, para lo cual las opciones consideradas para su implementación fueron:

Tabla 14.

Especificaciones del elemento de control

Características	Peso	Elemento de control	
		Relé	Optotransistor
Tamaño	15%	grande	pequeño
Principio de funcionamiento	25%	Mecánico	Óptico
Facilidad de uso	25%	Alto	Alto
Robustez	25%	Alta	Media
Señal admisible	10%	Pulso digital	Pulso digital o PWM
Total	100%	65%	75%

El optotransistor logra un 75% gracias a que su principio de funcionamiento que no es mecánico y por cuestiones de diseño resulta más fácil de implementar en pequeños espacios gracias a tu tamaño, incluso al combinarlo con un mosfet, adicionalmente se puede transmitir un tren de pulsos para controlar el voltaje suministrado a cada cerradura.

Para poder determinar el estado de la puerta se planteó emplear un pulsador o un sensor de efecto Hall como final de carrera, obteniendo la siguiente tabla.

Tabla 15.

Especificaciones del sensor final de carrera

Características	Peso	Sensor	
		Pulsador	Sensor Hall
Tamaño	20%	Pequeño	Pequeño
Principio de funcionamiento	20%	Mecánico	Magnético
Robustez	25%	media	Alta
Visibilidad para el usuario	20%	si	no
Señal de retorno	5%	Digital	Digital
Durabilidad	10%	media	Alta
Total	100%	35%	100%

Debido a que conviene que los componentes no sean visibles para el usuario, se ocupó un sensor de efecto Hall o sonda Hall, que gracias a su principio de funcionamiento entrega una señal digital para determinar si el imán ubicado en la puerta se encuentra próximo (cerrado) o lejano (abierto) y genera seguridad al sistema por su robustez.

Selección de microcontrolador

Debido a que solo van a tener acceso personal de mantenimiento a las placas, para poder cargar las actualizaciones en la programación se optó por la conexión mediante ICSP.

Tabla 16.*Especificaciones del microcontrolador*

Parámetros	Peso	Microcontrolador	
		Atmega328p	Atmega 2560
Tamaño	10%	mediano	Pequeño
Presentación	10%	Encapsulado	SMD
Robustez	10%	Alta	Alta
ICSP	20%	Si	Si
Complejidad para implementación	15%	Fácil	Media
Pines digitales	15%	14	54
Pines analógicos	15%	12	31
Memoria RAM	5%	2kB	8kB
Total	100%	45%	85%

Se estableció para una primera versión al microcontrolador Atmega328P por la presentación en encapsulado y por la complejidad del diseño en la placa al momento de routear. Posteriormente se aplica el Atmega2560 que posee mayor capacidad computacional y su presentación en SMD permite diseñar placas más compactas.

Diseño de pistas

Los cálculos del ancho de banda para los circuitos de control se diseñaron bajo la normativa EMC que define el ancho de pista según el amperaje de la línea y según su aplicación (control, comunicación, potencia).

Cálculos para el ancho de pista para los circuitos de control

El ancho de la traza se calcula para los valores de:

$$A = 2 \text{ Amperios}$$

$$\text{Grosor} = 10 \text{ onzas/pies}^2$$

$$\text{Aumento de temperatura} = 10^\circ\text{C}$$

Primero, se calcula el área:

$$\text{Área}[\text{mils}^2] = \left(\frac{\text{Corriente}[\text{Amps}]}{k * (\text{Temp_Rise}[\text{grados C}]^b)} \right)^{\frac{1}{c}}$$

Ecuación 12.

Ecuación para el cálculo de área

Nota. Tomado de "Reglas de diseño y estándares industriales" por Blanes, J.

Para capas internas IPC-2221: k = 0,024, b = 0,44, c = 0,725

Para capas externas IPC-2221: k = 0,048, b = 0,44, c = 0,725

donde k, b y c son constantes resultantes del ajuste de curvas a las curvas IPC-2221 y para este caso solo se van a calcular para capas internas:

Reemplazando con los requerimientos:

$$\text{Área}[\text{mils}^2] = \left(\frac{2[\text{Amps}]}{0,024 * (10[^\circ\text{C}])^{0,44}} \right)^{\frac{1}{0,725}}$$

$$\text{Área} = 126,087[\text{mils}^2]$$

Luego, se calcula el ancho:

$$\text{Ancho}[\text{mils}] = \frac{\text{Área}[\text{mils}^2]}{\text{Grosor}[\text{oz}] * 1.378[\text{mils}/\text{oz}]}$$

Ecuación 13.

Ecuación para el ancho de pista

Nota. Tomado de "Reglas de diseño y estándares industriales" por Blanes, J.

$$\text{Ancho}[\text{mils}] = \frac{126,087[\text{mils}^2]}{10[\text{oz}] * 1.378[\text{mils}/\text{oz}]}$$

$$\text{Ancho}[\text{mils}] = 9.15 \text{ mils}$$

Su equivalente a mm es:

$$\text{Ancho}[\text{mils}] = 0.23241 \text{ mm}$$

Resultado de la simulación

Figura 33.

Parámetros de cálculo para pistas de control

Entradas:		
Actual	2.2	Amperios
Grosor	10	onzas / pie ² ▾

Entradas opcionales:		
Aumento de la temperatura	10	Grado C ▾
Temperatura ambiente	25	Grado C ▾
Longitud de seguimiento	13	cm ▾

Resultados para capas internas:		
Ancho de trazo requerido	0.232	milímetro ▾
Resistencia	0.0283	ohmios
Caída de voltaje	0.0623	voltios
Pérdida de potencia	0.137	vatios

Nota. Se empleó la calculadora virtual para obtener los parámetros

Cálculos para el ancho de pista para los circuitos de potencia

El ancho de la traza se calcula para los valores de:

$$A = 6 \text{ Amperios}$$

$$Grosor = 10 \text{ onzas/pies}^2$$

$$\text{Aumento de temperatura} = 20 \text{ }^\circ\text{C}$$

Primero, se calcula el área usando la ecuación 12:

$$\text{Área}[\text{mils}^2] = \left(\frac{\text{Corriente}[\text{Amps}]}{k * (\text{Temp_Rise}[\text{grados C}])^b} \right)^{\frac{1}{c}}$$

Reemplazando con los requerimientos:

$$\text{Área}[\text{mils}^2] = \left(\frac{6[\text{Amps}]}{0,024 * (20[^\circ\text{C}])^{0,44}} \right)^{\frac{1}{0,725}}$$

$$\text{Área} = 329.54[\text{mils}^2]$$

Luego, se calcula el ancho usando la ecuación 13:

$$\text{Ancho}[\text{mils}] = \frac{\text{Área}[\text{mils}^2]}{\text{Grosor}[\text{oz}] * 1.378[\text{mils}/\text{oz}]}$$

$$\text{Ancho}[\text{mils}] = \frac{126,087[\text{mils}^2]}{10[\text{oz}] * 1.378[\text{mils}/\text{oz}]}$$

$$\text{Ancho}[\text{mils}] = 23.914 \text{ mils}$$

Su equivalente a mm es:

$$\text{Ancho}[\text{mils}] = 0.607 \text{ mm}$$

Resultado de la simulación

Figura 34.

Parámetros de cálculo para pistas de potencia

Entradas:		
Actual	6	Amperios
Grosor	10	onzas/pie^2 ▾
Entradas opcionales:		
Aumento de la temperatura	20	Grado C ▾
Temperatura ambiente	25	Grado C ▾
Longitud de seguimiento	13	cm ▾
Resultados para capas internas:		
Ancho de trazo requerido	0.607	milímetro ▾
Resistencia	0.0112	ohmios
Caída de voltaje	0.0672	voltios
Pérdida de potencia	0.403	vatios

Nota. Se empleó la calculadora virtual para obtener los parámetros

El diseño de la placa se ve limitada por la tecnología de fabricación de placas del país, esto se debe a que el ancho de la pista para el sistema resulta menor a 1 mm, por lo que se considera óptimo realizar el primer prototipo por el método del planchado a doble capa, se ocupa los siguientes valores al routear:

Tabla 17.

Parámetros de ruteado

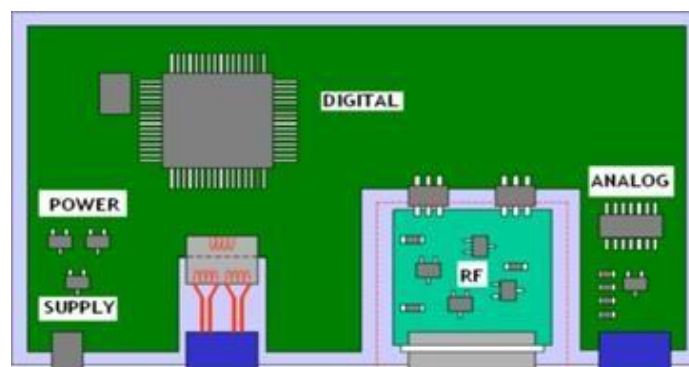
Parámetros	Routeado maestro	Routeado esclavo
Ancho de pista	1 mm	1 mm
Diámetro exterior pad	2mm	2mm

Parámetros	Routeado maestro	Routeado esclavo
Diámetro de orificio para componentes	0.7 mm	0.7 mm
Capas	2	2
Mallado superior	5v	5v
Mallado inferior	Gnd	Gnd
Espaciado entre pista y mallado	0.3 mm	0.3 mm
Orificios de montaje	3	4
Puerto USB	Si	No

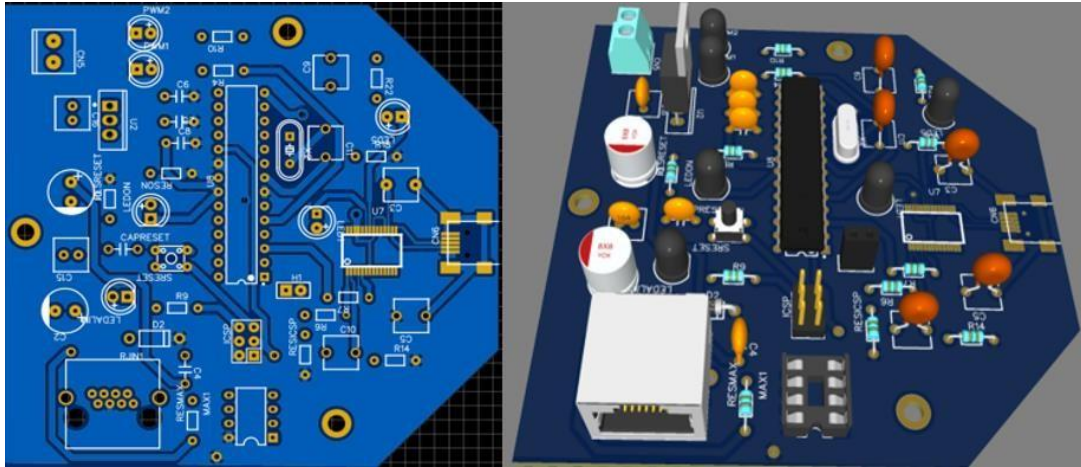
La plataforma Easyeda permite aplicar todos los parámetros de diseños de pistas especificados en la tabla, obteniendo los siguientes resultados basados en la normativa EMC que propone la disposición por sectores de los componentes en la placa.

Figura 35.

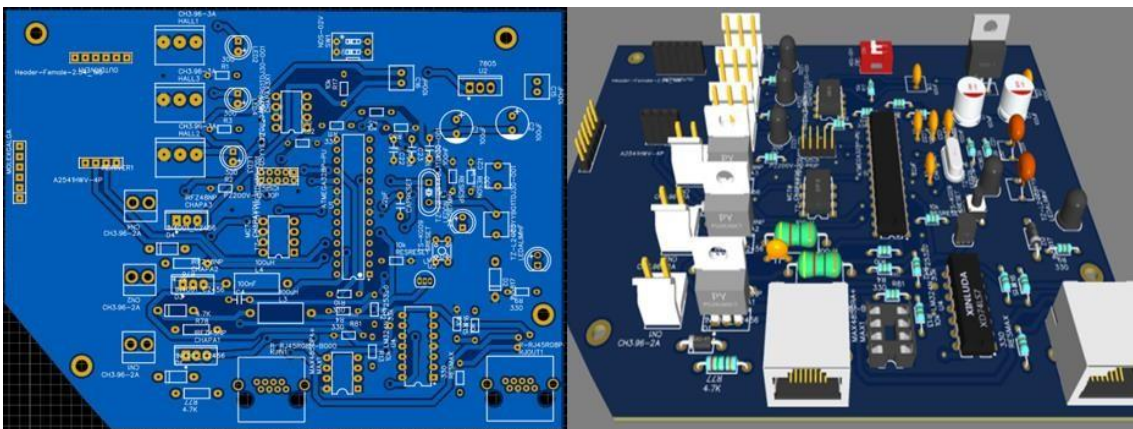
Disposición de la placa por bloques funcionales recomendada



Nota. Tomado de Getting EMC design right first time, (p. 25), por (Rangu M., 2019)

Figura 36.*Diseño de placa maestro*

Nota. Para la conexión USB se amplió las pistas del integrado y del puerto debido a que los componentes solo se disponen en SMD (Surface Mounting Device).

Figura 37.*Diseño de placa esclavo*

Nota. Por facilidad de suelda se optó en corregir manualmente el ruteado, minimizando el número de vías en la placa.

Diseño software

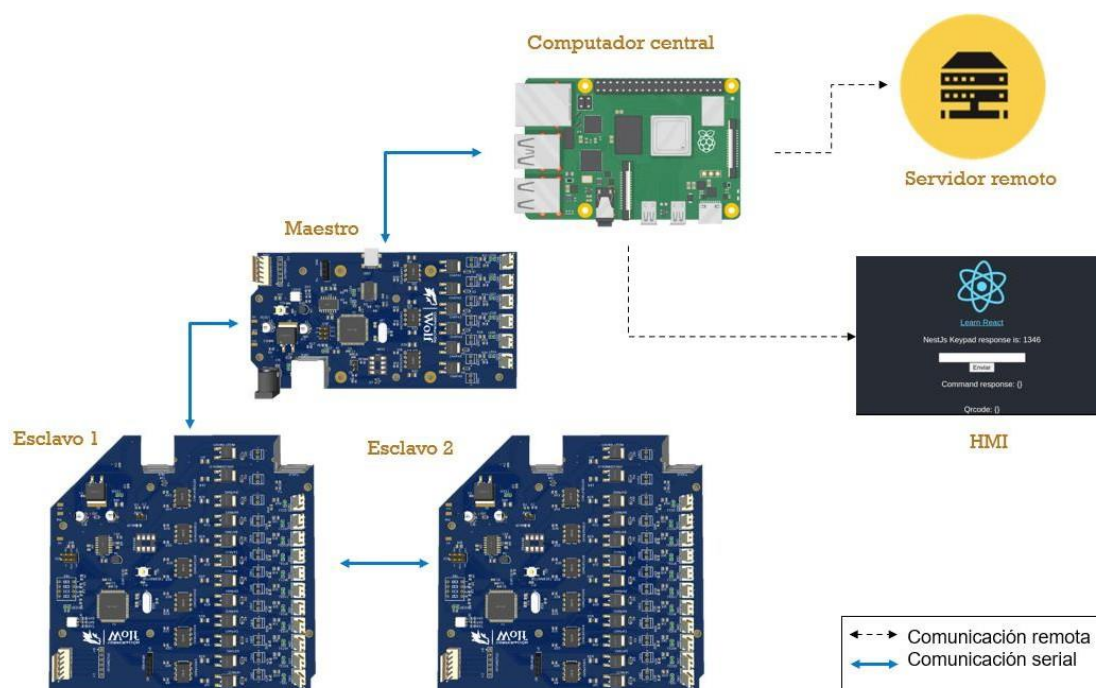
Esquema de conexión

Los requerimientos del proyecto para su diseño de software parte de la necesidad de controlar a la planta de forma remota. Por lo tanto, se requiere también de un computador central ubicado localmente en la planta, dotado de conectividad con el servidor remoto para disminuir la carga operacional del mismo.

Este computador central debe tener la capacidad de establecer una comunicación bidireccional con un sistema maestro-esclavo para administrar el uso de sensores y actuadores con la finalidad de cumplir los requerimientos del sistema de paquetería. Además, debe tener la capacidad de ejecutar una interfaz dirigida al cliente y a su vez tener conectividad con periféricos como teclado y lector QR.

Figura 38.

Esquema de comunicación del sistema



Se propone el esquema de conexión según la figura 39 para la transferencia de datos en modo usuario, que parte de la interacción del cliente con el HMI en el computador central y se dirige a manera de comandos hacia el sistema maestro esclavo, recibiendo valores de estado del casillero que posteriormente serán publicados en el servidor remoto.

Para el modo administrador, el servidor remoto publica valores de estado en el computador central y este debe ejecutarlas en el sistema maestro esclavo, no hay necesidad de implementar un HMI.

Arquitectura Backend-Frontend

En la arquitectura del sistema, la Raspberry funciona como un sitio o aplicación web y por consiguiente tiene servicios de Frontend y Backend.

Para Frontend es importante seleccionar framework que es el esquema o marco de trabajo para el desarrollo del HMI y el lenguaje de programación; entonces se definen los siguientes parámetros de selección.

Tabla 18.

Parámetros de selección del framework de Frontend

Aplicación	Peso	React	Angular
Desarrollo	10%	Web	Web-Movil
Autosuficiencia	10%	Requiere de librerías adicionales	No requiere de librerías adicionales
Curva de aprendizaje	20%	Estructura simple y tiene constantes actualizaciones	Sintaxis innecesaria y administración compleja
Desempeño	15%	Reduce la carga a los navegadores	Cuanto más vinculaciones tenga, es más lento el proceso
Lenguaje	20%	JavaScript ES6 + JSX script que se asemeja a HTML	Usa JavaScript o TypeScript
Estructura de aplicación	20%	No existe "una única estructura correcta"	La estructura es fija y compleja
Herramientas	5%	Compatible con Visual Studio, Atom, Sublime Text	Compatible con Visual Studio, Atom, Sublime Text
Total	100%	90%	65%

Se propone desarrollar un aplicativo con React, por su facilidad de aprendizaje en comparación a Angular, también por la flexibilidad de la estructura de la aplicación, de la curva de aprendizaje y su desempeño es aprovechado por la tarjeta Raspberry Pi4.

Tabla 19.

Criterios de selección del lenguaje de programación para el sistema

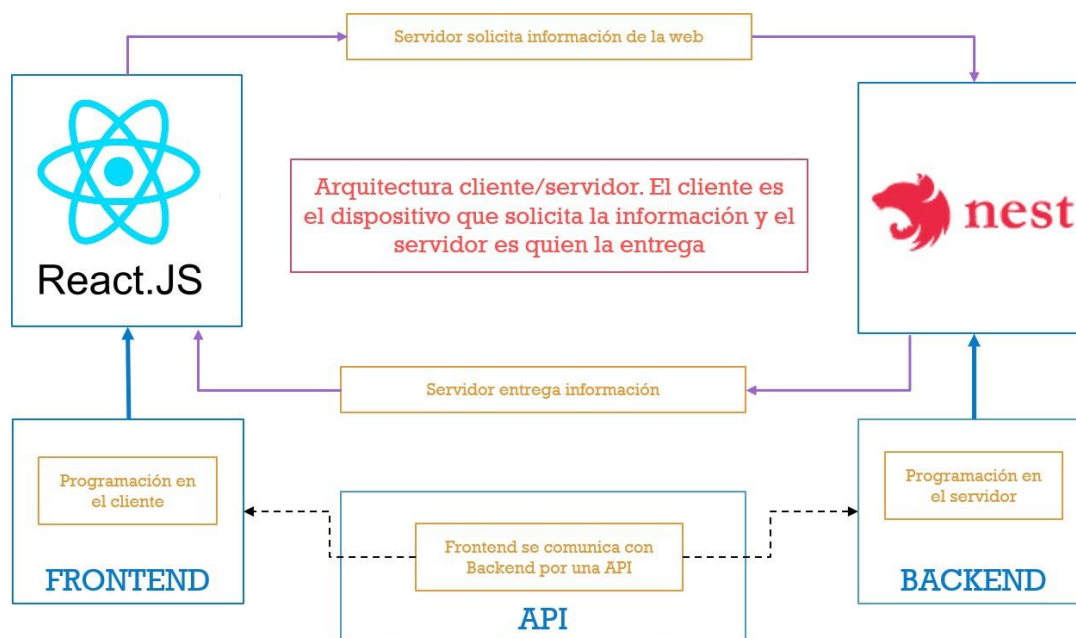
Lenguaje	Peso	JavaScript	TypeScript
Tipado	10%	Lenguaje ligero	Potente sistema de tipos, que incluye características genéricas y JS
Modularización	5%	Patrocinio de Microsoft	Admite bibliotecas JS y documentación API (Application Programming Interface)
Orientación a objetos	15%	Se puede, pero es más complejo	Sintaxis dedicada
Decoradores	15%	No soporta decoradores	Soporta decoradores
Interfaces	20%	No soporta	Soporta interfaces
Curva de Aprendizaje	15%	Bajo	Medio
Aplicación en proyectos	10%	Proyectos pequeños	Proyectos medianos y grandes
Uso de frameworks	10%	No es necesario	Es necesario
Total	100%	35%	85%

El lenguaje del proyecto se va a realizar en TypeScript con una aprobación de 85%, debido a que soporta interfaces y es ideal para proyectos medianos y grandes.

Además, la ayuda de frameworks y decoradores es un punto vital para el desarrollo de la interfaz y control de datos.

Figura 39.

Esquema de funcionamiento del sistema Backend-Frontend



Comunicación Raspberry con servidor remoto

La arquitectura para la Raspberry Pi 4 como la computadora central busca aplicar tecnologías IoT (Internet of Things).

En tecnologías IoT, los protocolos más usados son:

- MQTT (Message Queing Telemetry Transport)
- CoAP (Constrained Application Protocol)
- HTTP (Hypertext Transfer Protocol)

Tabla 20.

Los parámetros de selección para el protocolo

Protocolo	Peso	MQTT	CoAP	HTTP
Protocolo Base	10%	TCP	UDP	TCP

Protocolo	Peso	MQTT	CoAP	HTTP
Consumo de energía	15%	Mayor que CoAP	Menor que MQTT	Mayor que MQTT
Número de tipos de mensaje usados	10%	16	4	Indefinido (depende del servidor web)
Tamaño de la cabecera	5%	2 bytes	4 bytes	Indefinido
Fiabilidad	20%	3 niveles de calidad de servicio QoS 0: Entrega no garantizada QoS 1: Confirmación de entrega QoS 2: Doble confirmación de entrega	Mensajes confirmables Mensajes no confirmables ACKs Retransmisiones	Limitado al protocolo TCP
Implementación	15%	Fácil de implementar y complejo para añadir extensiones	Pocas librerías existentes y soporte	Fácil de implementar, no siempre está optimizado
Seguridad	15%	Puede utilizar TLS/SSL	DTLS o IPSec	TLS/SSL

Protocolo	Peso	MQTT	CoAP	HTTP
Otro	10%	Útil para conexiones en localizaciones remotas	Baja saturación Baja latencia	Útil para envío de paquetes grandes y capacidad de respuesta
Total	100%	85%	40%	60%

El protocolo que más cumple las necesidades para el proyecto es MQTT con un 85% de aprobación debido a la fiabilidad del servicio de envío y recepción de mensajes de tipo suscriptor-publicador, además usa el protocolo base TCP, lo que optimiza su funcionamiento para trabajar con recursos limitados y presenta una ventaja para localizaciones remotas.

Con 60% de aprobación, el protocolo HTTP tiene una estructura de números de tipo de mensaje que dependen del servidor, esto beneficia al tamaño de cabecera que permite usar grandes cantidades de datos y validar la información al servidor, por lo cual el sistema puede comprobar datos del cliente enviando peticiones desde la API.

Comunicación Raspberry - maestro

Tabla 21.

Criterios de selección de Raspberry con placas maestro/esclavo

Protocolo	Peso	Puerto serial	I2C
Designación de pines	5%	TxD: Transmisión de datos. Rx D: Recepción de datos.	SDA: Datos SCL: Reloj serial

Protocolo	Peso	Puerto serial	I2C
Velocidad de datos	15%	La comunicación máxima puede ser entre 230Kbbs a 460kbb	Hasta 3.4Mbbs algunas variantes pueden llegar a 1Mbbs
Distancia	10%	Menos de 15 metros	Está pensada para comunicaciones dentro de la placa
Número de maestros	5%	No hay maestros.	Varios
Tipo comunicación	15%	Asíncrona	Síncrona
Complejidad del hardware	20%	Baja	Media
Reloj	15%	Cada dispositivo usa el suyo interno.	Una señal de reloj común entre múltiples maestros y esclavos.
Direccionamiento de software	15%	La comunicación es entre dos dispositivos	Todos los maestros pueden comunicarse con todos los esclavos
Total	100%	85%	45%

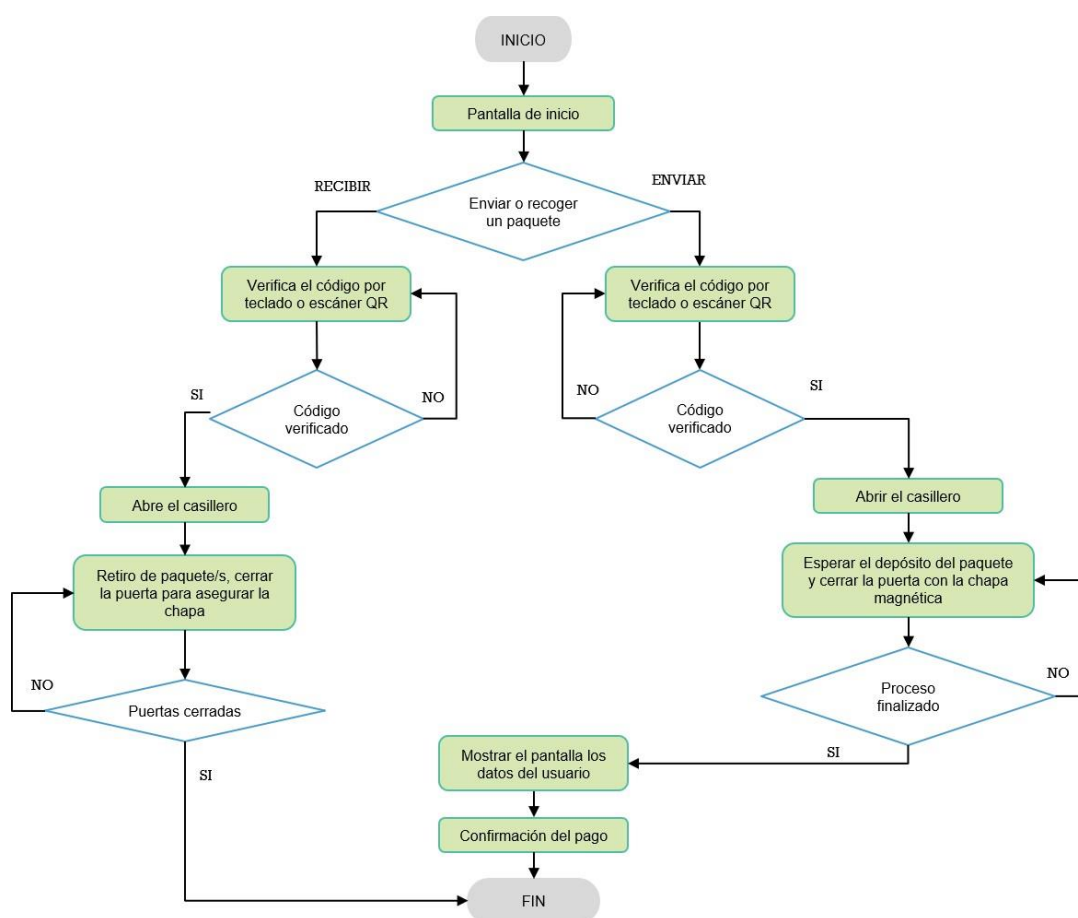
La arquitectura de comunicación que mejor se acopla al proyecto es el puerto serial, debido a que en cada estación solo habrá una Raspberry como computador local conectado a una unidad de control (maestro) que a su vez se comunica en serie a sus esclavos. En este sentido, la distancia y la complejidad de la comunicación optimizan el uso del espacio para la instalación, además la baja velocidad de datos se compensa con el protocolo de comunicación por comandos que requieren poca memoria.

Diseño del HMI

El sistema tiene un HMI que sirve para controlar el proceso de entrega y recepción de paquetes, este proceso responde a las peticiones del cliente y gestiona las acciones del maestro/esclavo. Dispone de una pantalla ubicada en el locker principal y la navegación entre ventanas está desarrollada en función del teclado matricial.

Figura 40.

Diagrama general de proceso del usuario



Arquitectura de comunicación

La arquitectura de comunicación se define como la combinación de un aplicativo Backend-Frontend con un enlace hacia el servidor remoto mediante el protocolo MQTT y también se conecta a las placas maestro/esclavo usando el puerto serial.

Figura 41.

Esquema de conexión para la comunicación

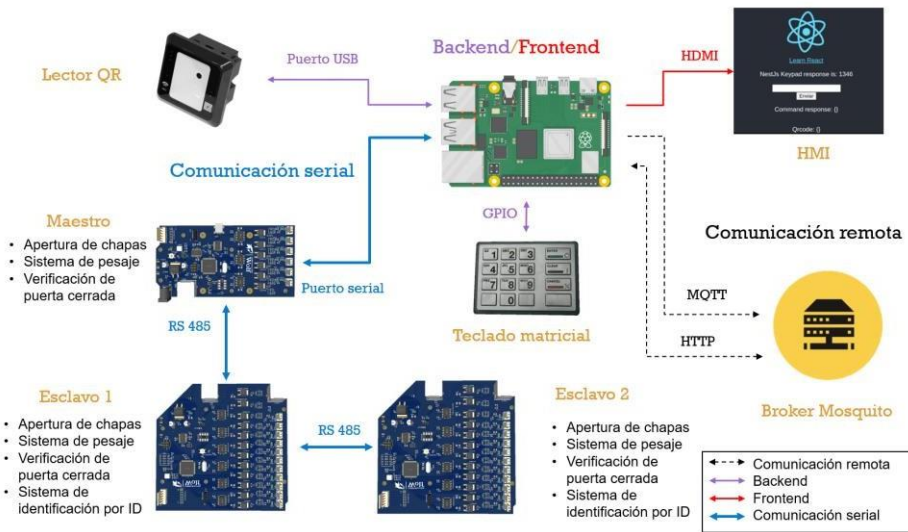
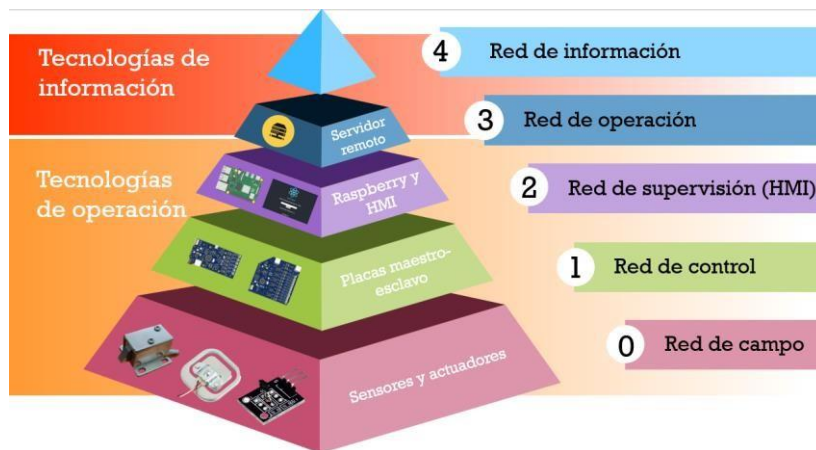


Diagrama general del sistema

El diagrama general del sistema parte del desarrollo de la pirámide de automatización, específicamente en las “Tecnologías de Operación”; los protocolos de comunicación permiten que el servidor remoto registre los estados de los casilleros gracias a la administración de la computadora central local. Se configura a los microcontroladores ATMEGA 2560 para que puedan leer todos los sensores y accionar a los actuadores bajo la arquitectura modular maestro/esclavo tipo bus mediante el protocolo RS-485.

Figura 42.

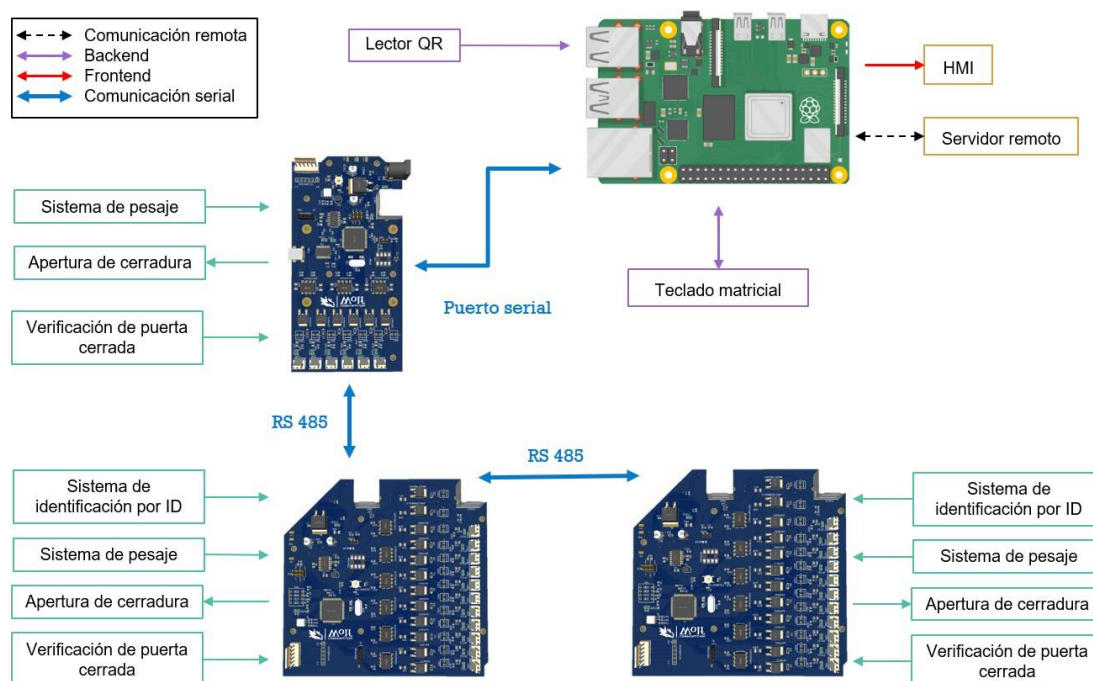
Pirámide de automatización para el servicio de envío de paquetes



Se programa a la Raspberry con el aplicativo Backend-Frontend como computadora central, debido a que se encarga del diseño HMI que forma parte del servicio Frontend, del mismo modo, la lectura del teclado matricial, la recepción de datos con el lector QR, está en el lado de Backend. Este último servicio establece la comunicación hacia el puerto serial con las tarjetas maestro/esclavo y se vincula con el servidor remoto.

Figura 43.

Diagrama general de conexión para la comunicación



El servidor remoto controlará a la Raspberry mediante el protocolo HTTP y recibirá los datos de la planta en tiempo real por MQTT, como se muestra en la figura 43. La Raspberry responderá a los requerimientos del servidor brindando datos al cliente por el HMI y solicitando información a las placas maestro esclavo para administrar el estado de los sensores y actuadores.

Capítulo IV

Construcción del locker

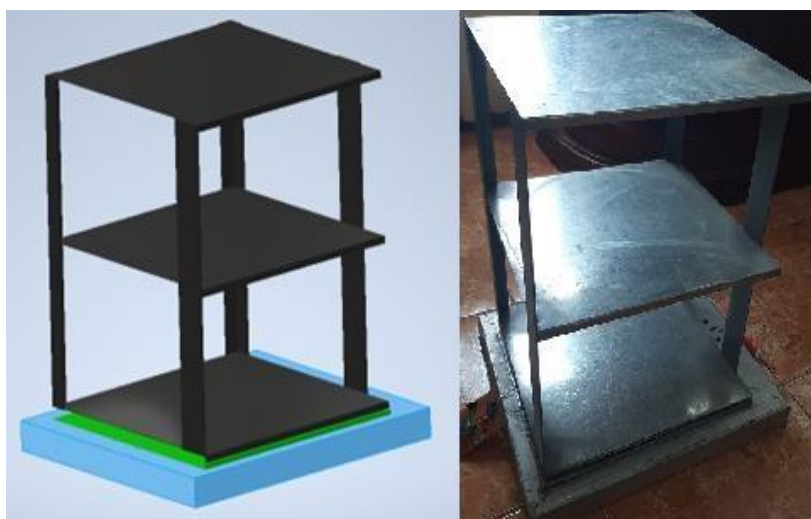
Implementación del sistema de pesaje

Para el primer prototipo se desarrolló la construcción de un locker a escala con la finalidad de probar y corregir todos los posibles errores que en el CAD no se pueden apreciar.

La unión de las 3 estructuras (percha interna, lamina de apoyo, base rectangular), forma el subsistema de sensado, fundamental para poder obtener un peso aproximado del paquete depositado en cada uno de los casilleros.

Figura 44.

Subsistema de pesaje



Nota. Para que el subsistema de sensado funcione óptimamente, debe estar posicionada la percha sobre las 4 galgas de manera correcta.

Percha interna

Debido a que la función de la percha interna es el de almacenar los paquetes y permitir el sensado de los mismos, para las columnas se emplearon dos platinas en L y dos platinas rectangulares; mientras que para cada bandeja se empleó tol galvanizado de 0.9 mm de espesor, con la finalidad de evitar que la percha se vuelva pesada y a la vez sea resistente cómo se lo evidenció en el capítulo anterior.

Figura 45.

Percha interna



Lámina de apoyo

Debido a que sobre esta lámina va a encontrarse la percha con los paquetes, se empleó tol galvanizado de 1.4 mm de espesor; los orificios cuadrados para colocar las galgas, están dispuestos de manera equidistantemente respecto al centro, sin que influyan en la deformación para el sensado.

De la misma manera se dispuso 8 orificios en los bordes para poder atornillar la lámina de apoyo al armazón.

Figura 46.

Lámina de apoyo

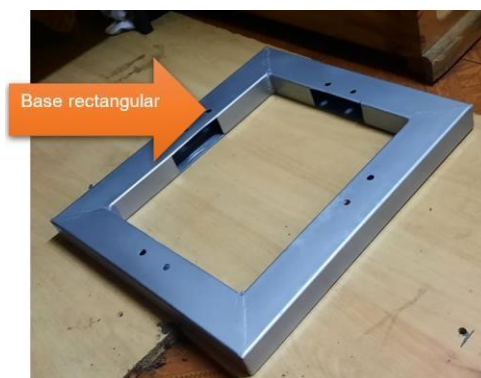


Base rectangular

Empleando tubos cuadrados de acero se construyó una base dónde internamente se pueda pasar el cableado para la expansión modular y al mismo tiempo permita atornillar la lámina de apoyo de tal manera que el centro permanezca libre.

Figura 47.

Base rectangular



Parte electrónica

Previo a la construcción de las placas electrónicas se realizó pruebas para cada uno de los subsistemas a controlar.

Expansión modular

Mediante software serial se estableció los pines que funcionarán como gate, TX y RX tanto para maestro como para esclavos, se empleó 4 puertos digitales para leer el ID que se genera al implementar un Dip switch. Las tarjetas ocupadas y los puertos designados son los siguientes:

Tabla 22.

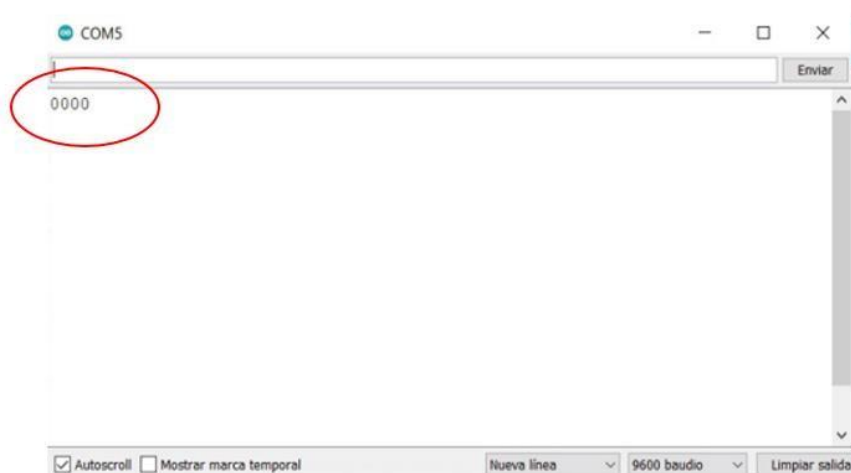
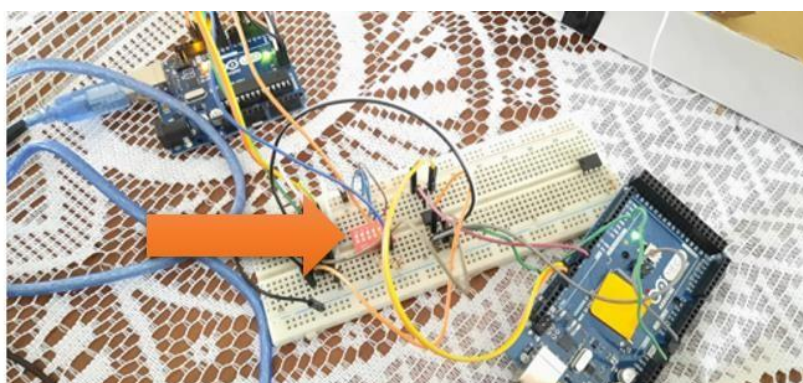
Parámetros de conexión maestro-1 esclavo

Parámetros	Maestro	Esclavo
Tarjeta	Arduino Mega	Arduino Uno
Pin RX	10	12
Pin TX	11	13

Parámetros	Maestro	Esclavo
Pin Gate	8	11
Pines Dip switch	N/A	10,9,8,4

Figura 48.

Conexión prueba 1 maestro-1 esclavo



El Arduino maestro recibe el ID del esclavo establecido en el Dip switch y lo muestra en el command window, en este caso el ID del primer locker es 0000.

Tabla 23.

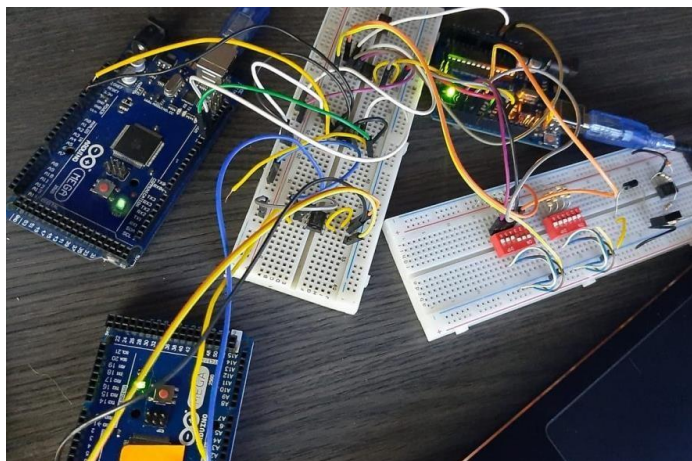
Parámetros de conexión 1 maestro- 2 esclavos

Parámetros	Maestro	Esclavo
Tarjeta	Arduino Mega	Arduino Uno/Mega
Pin RX	10	12
Pin TX	11	13

Parámetros	Maestro	Esclavo
Pin gate	8	8
Pines Dip switch	N/A	2,3,4,5

Figura 49.

Conexión prueba 1 maestro-2 esclavos

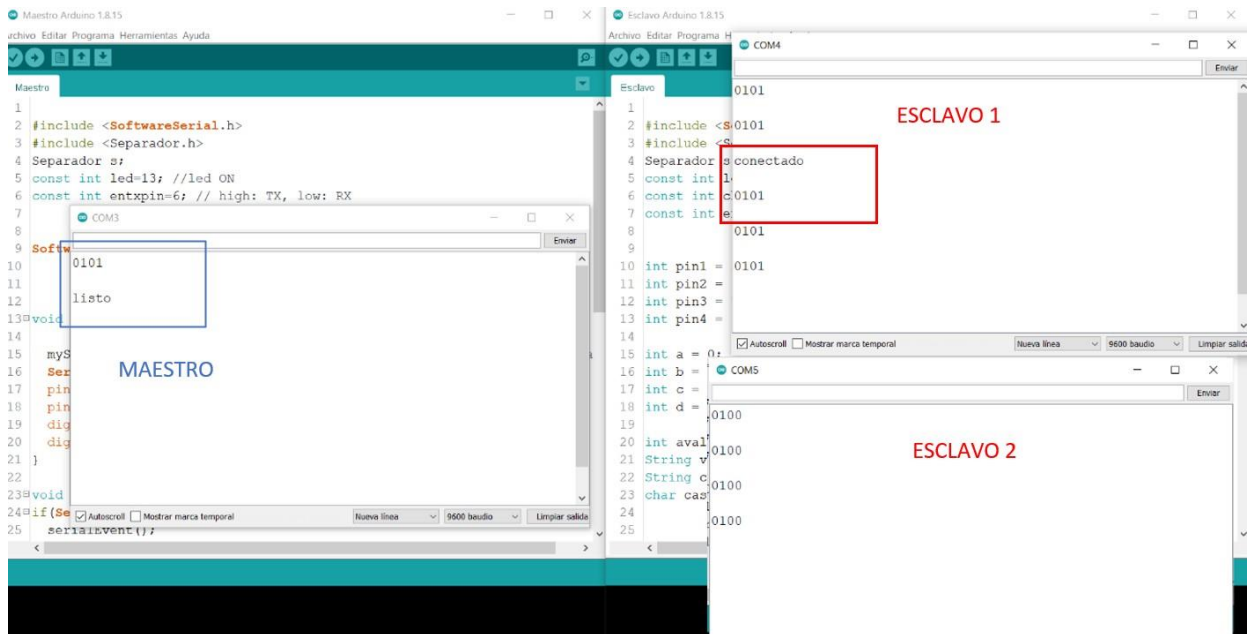


Con el Arduino maestro se energizan el sistema y en la ventana de comandos se ingresa el ID con el esclavo que se desea comunicar.

Al mismo tiempo en la ventana de comandos de los esclavos se imprime continuamente el ID que se les ha asignado (0101 - 0100) y al momento de recibir el comando imprime en pantalla “conectado” respondiendo al maestro; este imprime “listo” al final de la secuencia.

Figura 50.

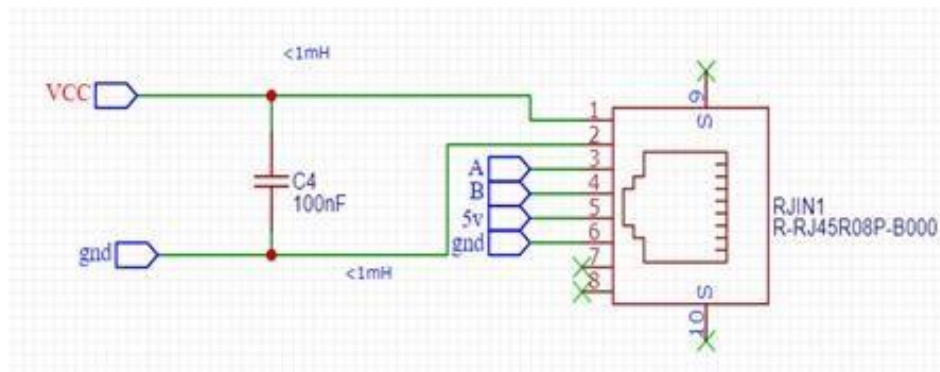
Comunicación y respuesta del esclavo



Debido a que el cable UTP para los puertos RJ45 dispone de 4 pares trenzados, los pines asignados fueron los siguientes:

Figura 51.

Designación de pines para puerto Rj45



Nota. Se asignó un solo puerto para el maestro (salida) y dos puertos para cada esclavo (Entrada y Salida).

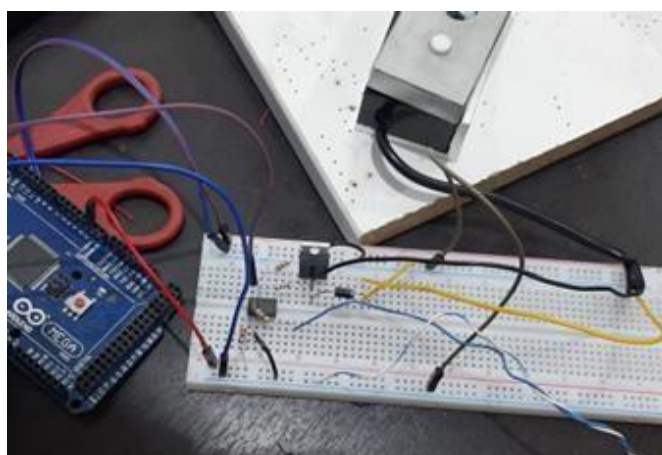
Como se explicó en el capítulo anterior, por cuestión de diseño al momento de suavizar y limpiar la señal de entrada, se dispone de un capacitor cerámico de 100nf.

Tabla 24.*Pines del puerto Rj485*

Pines	Simbología	Funcionalidad
1	VCC	Voltaje de fuente 12 V DC
2	Gnd	Tierra de fuente
3	A	Canal RX MAX 485
4	B	Canal TX MAX 485
5	5 V	Voltaje del regulador
6	Gnd	Tierra del regulador
7	N/A	N/A
8	N/A	N/A

Seguridad

Para la implementación de las cerraduras magnéticas se estableció el funcionamiento mediante PWM con la finalidad de controlar la fuerza de cierre mediante la regulación del ancho de pulso del voltaje.

Figura 52.*Conexión para cerradura magnética*

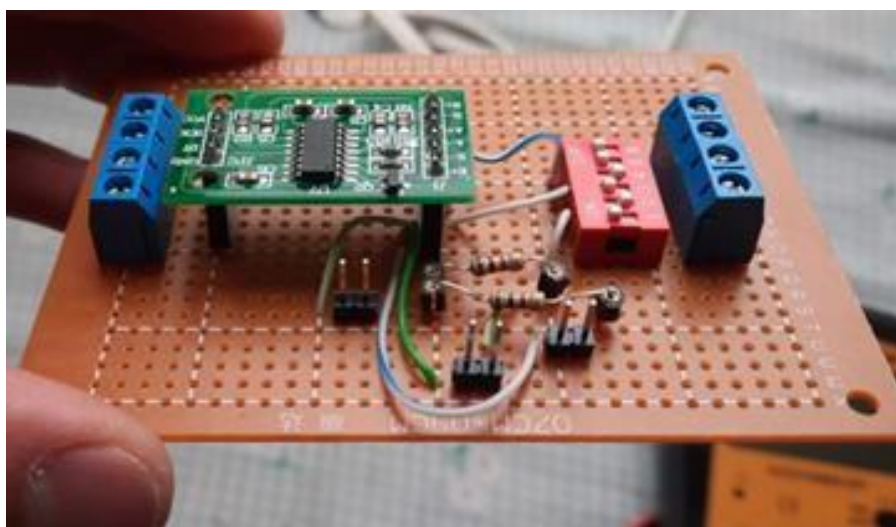
Nota. Con el mismo circuito se probó cada una de las cerraduras, notando un aumento de temperatura en las cerraduras NC (pequeñas) ya que estas requieren de su activación para liberar el seguro.

Implementación de las galgas

Previo a la construcción de la placa maestro y esclavo, se diseñó una placa para probar las galgas por pares, debido a que la resistencia con la que se complete el puente debe ser igual para obtener un diferencial aproximado y así establecer la correcta conexión con en el módulo Hx711:

Figura 53.

Placa de calibración para galgas



Nota. Se implementó un Dip switch para conmutar la conexión dependiendo la cantidad de galgas que se vayan a probar.

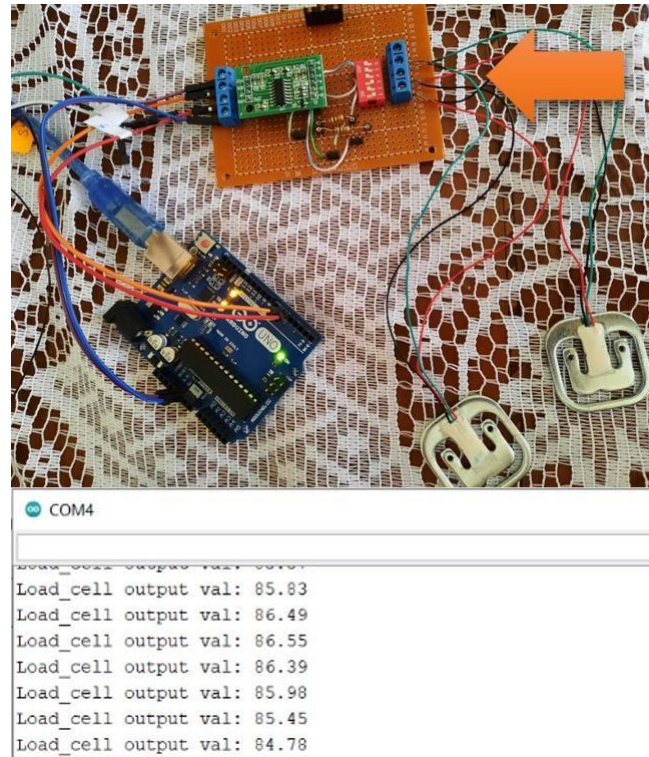
- **2 Galgas**

En las galgas empleadas el color del cable diferencial se clasifica en verde y blanco, por lo cual se agrupó de acuerdo al color, obteniendo un par con cable de señal verde y un par con cable de señal blanca.

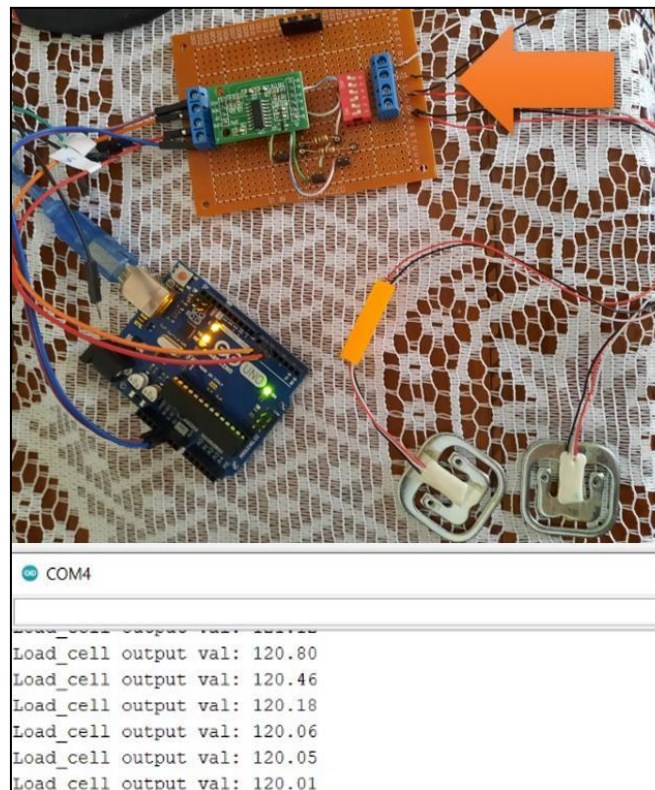
Después de realizar la conexión se cargó el programa en Arduino y se visualizó mediante el command window el funcionamiento de cada par imprimiendo en pantalla valores en gramos al aplicar presión en cualquiera de las galgas.

Figura 54.

Conexión de galgas con cable de señal color verde

**Figura 55.**

Conexión de galgas con cable de señal color blanco

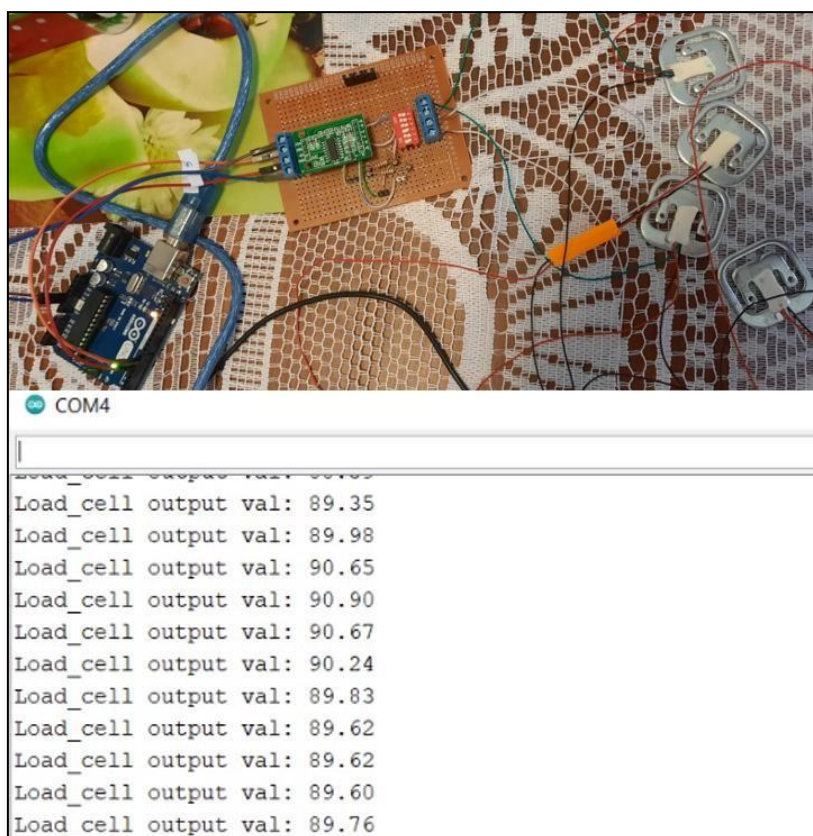


- **4 Galgas**

Una vez verificado el correcto funcionamiento de las galgas por pares se estableció la conexión de las 4 entré sí:

Figura 56.

Conexión de 4 galgas de un cuarto de puente



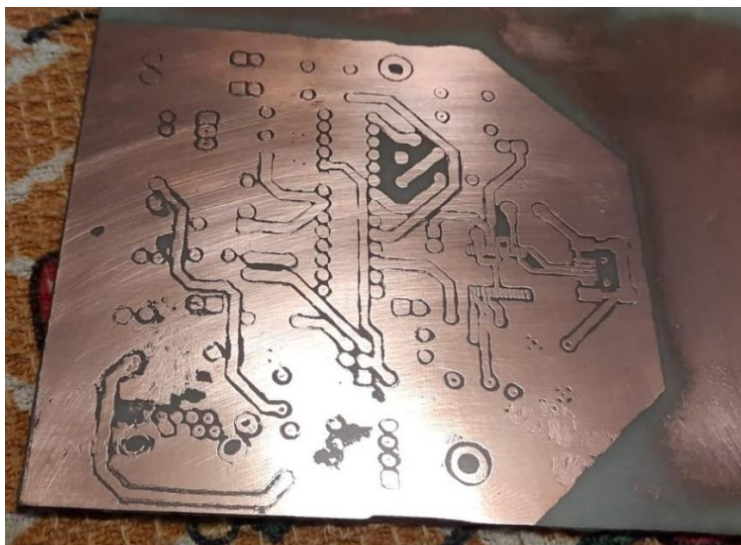
Nota. En la ventana de comandos se pudo comprobar el funcionamiento y la correcta conexión de las 4 galgas ya que se complementaban entre sí al momento de sensar.

Elaboración de las placas maestro y esclavo

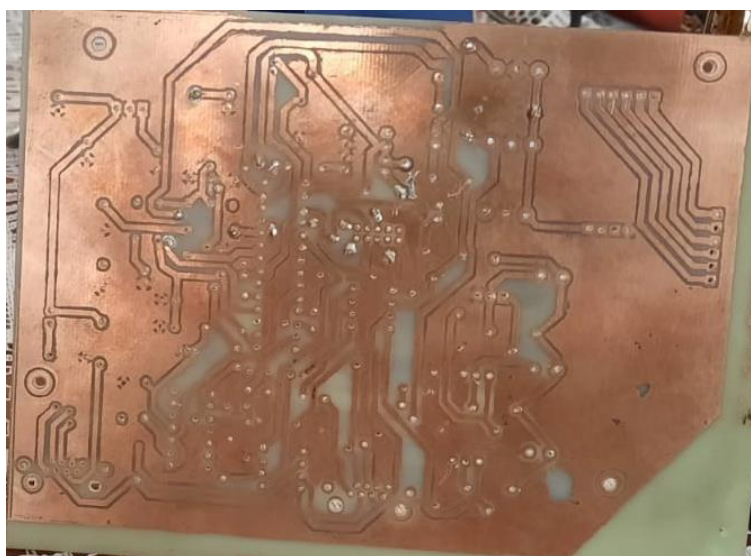
Una vez optimizada la disposición de los componentes que conformarán las placas y luego de haber comprobado el correcto ruteado y mallado en el área de cobre; se elaboró por el método del planchado a doble capa:

Figura 57.

Construcción de placa maestro

**Figura 58.**

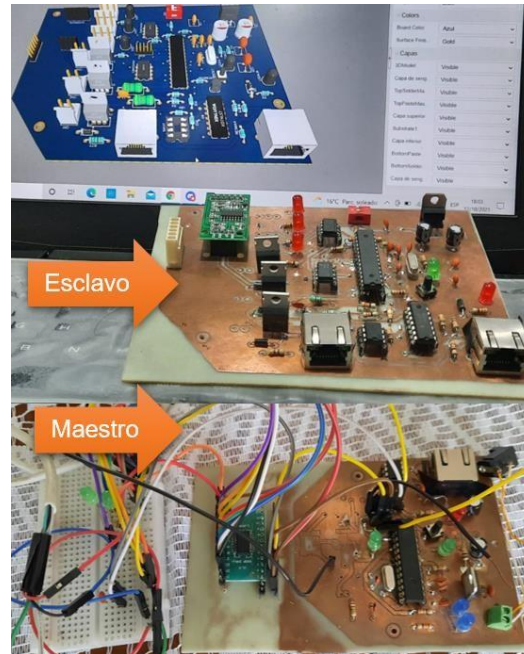
Construcción de placa esclavo



Nota. Una vez generada las pistas se realizó las perforaciones y se soldó los componentes para posteriormente probar la continuidad de punto a punto.

Figura 59.

Primera versión placas esclavo y maestro



Debido a que se requiere más salidas y entradas en la placa, posteriormente se diseñó y construyó placas con componentes en SMD que cumplan las mismas funciones pero que requieran menor espacio. En base a los cálculos de diseño para el ancho de pista y distribución de los componentes del capítulo 2.

Figura 60.

Distribución de elementos por áreas según EMC

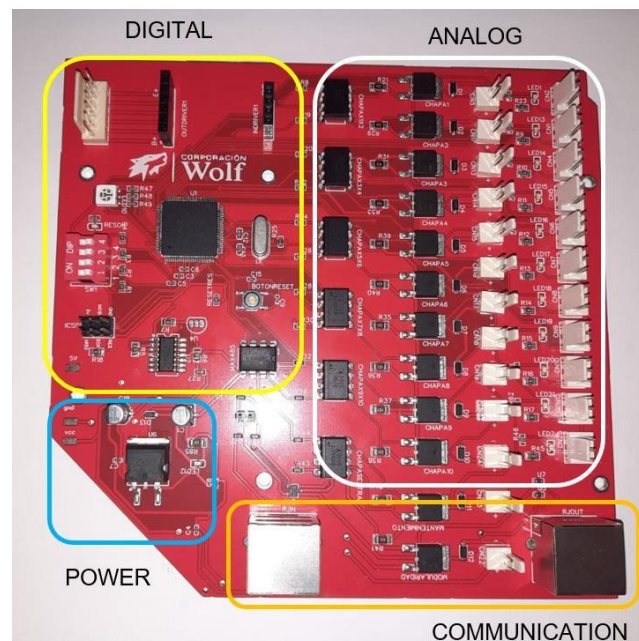


Figura 61.

Construcción de placa con componentes SMD

**Figura 62.**

Comparativa de placas esclavo



Nota. Para la versión final se establecieron 10 puertos de salida, mientras que para la primera se probó solo con 3.

Al construir la placa implementando conectores Molex, se estableció la modularidad en la estructura por la facilidad de poder conectar y reemplazar los periféricos (galgas, cerraduras, sensor Hall)

Implementación de la comunicación con el servidor

La arquitectura básica del sistema software se representa en la siguiente figura, en esta se aprecia como está conectada la pantalla, teclado y escáner QR

como periféricos para el HMI. Este último está programado en la Raspberry Pi 4 a manera de servidor-aplicación o Backend-Frontend.

Figura 63.

Conexión de periféricos a la Raspberry Pi 4



Nota. El Arduino representa la conexión al maestro.

Las funciones que requiere el sistema en modo administrador y en modo usuario se detallan en la siguiente tabla:

Tabla 25.

Sistema HMI.

Módulo	Ejecución
Aplicación web administrador	<ul style="list-style-type: none"> • Conectividad al servidor remoto MQTT • Conectividad al servidor remoto HTTP • Monitoreo de casilleros por comandos <ul style="list-style-type: none"> • Expansión modular
Aplicación web usuario	<ul style="list-style-type: none"> • Validación de códigos de usuario <ul style="list-style-type: none"> • Visualización del proceso • Monitoreo de casilleros por comandos <ul style="list-style-type: none"> • Pasarela de pagos

Módulo	Ejecución
Funcionalidad del prototipo HMI	<ul style="list-style-type: none"> • Lectura de código QR • Lectura del teclado matricial • Conexión al servidor remoto • Proceso: Apertura de casillero • Proceso: Cierre de casillero

Nota. La variación entre modo usuario y administrador es controlada por el servidor remoto, por lo tanto, el usuario no es capaz de acceder a la programación del administrador mediante el HMI.

Construcción de la estructura Backend-Frontend.

Para el montaje del sistema se usan las plantillas base. Para Backend, NestJS permite contar con controladores encargados de escuchar las peticiones de la aplicación y preparar las respuestas, esto lo hace mediante rutas llamadas endpoints que usan decoradores para definir parámetros y estructuras para la petición. Del mismo modo, NestJS tiene servicios que se encargan de controlar la lógica de funcionamiento y pueden aplicarse a todos los controladores para importar un servicio.

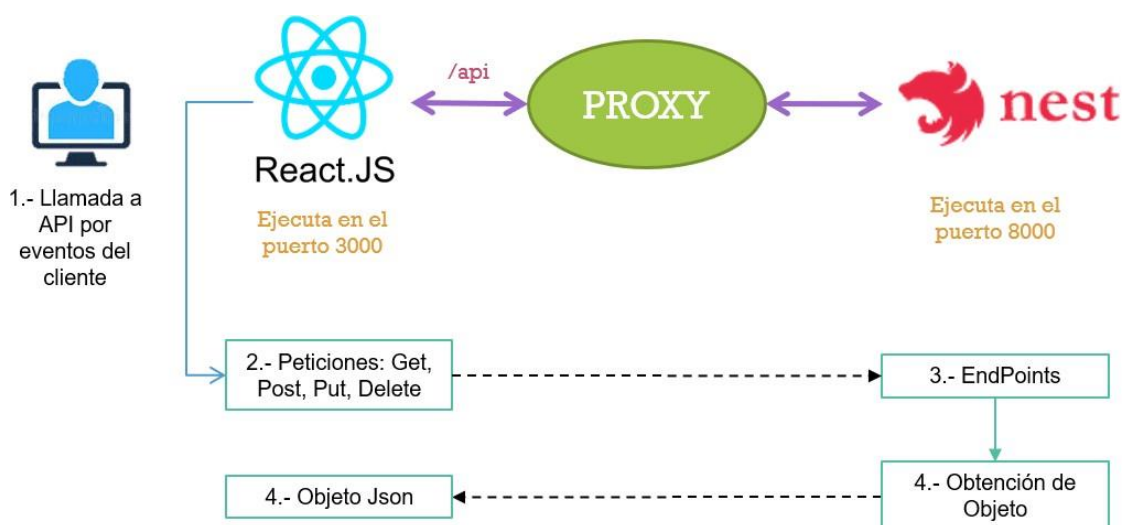
Para el Frontend la arquitectura básica de React permite crear una interfaz de usuario mediante componentes, básicamente son piezas de código que se manipulan para controlar lo que se muestra en pantalla. Cada componente define sus propiedades o características (props), que mediante jerarquía se muestran en pantalla con el método render.

En el archivo "index.html" se renderizan los componentes y se define la estructura mínima del proyecto, como las dimensiones de pantalla y la distribución de elementos. En el archivo del mismo nombre con extensión ".tsx", se insertan las rutas de la aplicación, también se coloca el contenido de la renderización; Finalmente, el archivo "App.tsx" almacena los componentes que se inyectarán por medio del formato "index.tsx" para mostrar en pantalla con "index.html".

Para comunicar el servidor local con la aplicación se usa proxy como puerta de enlace, así la interfaz de usuario se ejecuta en el puerto 3000 y Backend en el puerto 8000, estos valores pueden cambiar, pero nunca deben ser iguales. El intercambio de datos se da mediante peticiones a las rutas de endpoints como GET, POST, entre otros.

Figura 64.

Diagrama de comunicación por peticiones



Nota. Todas las llamadas a “/api” se redirigirán al servidor backend

Las peticiones del lado de Backend funcionan para determinar las rutas de comunicación de los endpoints y establecer canales para administrar el flujo de datos, por otro lado, para Frontend, se requiere de Fetch, siendo una herramienta nativa de React que permite realizar peticiones asíncronas HTTP para la aplicación a partir de un archivo JSON; para esto se utiliza métodos como “GET”, con la finalidad de realizar una solicitud a un URL en el servidor Backend y esperar una respuesta válida o errónea.

Construcción de la estructura Backend / comunicación serial.

La comunicación serial va a ser utilizada para enviar los mensajes de control hacia el maestro desde el puerto ttyUSB0 y a su vez va a recibir un comando de

respuesta con una estructura conformada con el ID de la torre, el número de locker y la acción.

Estas variables se detallan en la siguiente tabla.

Tabla 26.

Estados de las variables el sistema

Variable	Estados	Respuesta
Cerradura	<ul style="list-style-type: none"> • Lock • Unlock 	State
Sensor Hall	<ul style="list-style-type: none"> • Open • Close 	State
Sistema de pesaje	<ul style="list-style-type: none"> • Null • Data 	Data

Se usa dos librerías, “SerialPort” para acceder a los puertos de la Raspberry y “ReadlineParser” para dividir el mensaje con delimitadores o separadores.

En este apartado, la aplicación del cliente tiene un cuadro de texto y un botón que están programados por eventos, los datos de la trama del cuadro se guardan en una variable cada vez que detectan un cambio y pasa a Backend al hacer click en el botón.

Figura 65.

HMI para la comunicación por tramas Backend-maestro/esclavo



Nota. En el HMI se ingresa en el cuadro de texto el comando que inicia el proceso para recibir un paquete en el locker.

La trama se envía con una petición haciendo uso del método GET. Una vez que el servidor local tiene este dato, los servicios de NestJS ejecutan el código de escritura en el puerto serial y simultáneamente manda el mismo mensaje como respuesta a la petición para que se muestra en pantalla.

Figura 66.

Respuesta de la petición por el puerto serial.

```

pi@raspberrypi: ~/lockers-raspberry-platform
Archivo Editar Pestañas Ayuda
Compiled successfully!

You can now view raspberry-frontend in the browser.

  Local:            http://localhost:8000
  On Your Network:  http://192.168.100.25:8000

Note that the development build is not optimized.
To create a production build, use npm run build.

Parser got word from serial: #
parser got word from serial: ID:1212
parser got word from serial: Locker y accion:
parser got word from serial:
parser got word from serial: Usuario incorrecto, ingrese nuevamente
parser got word from serial:
parser got word from serial: ID:alvaro
parser got word from serial: Locker y accion:10_11

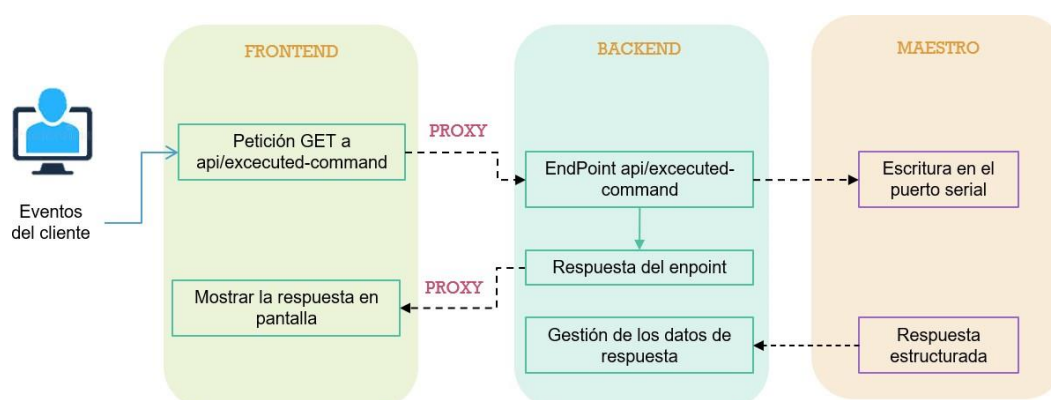
```

Nota. El código envía mensajes al puerto cada vez que acaba con una acción del proceso de recibir un paquete

Una vez se envía el comando al puerto serial, se espera una respuesta del maestro y gracias a la librería “ReadlineParser”, se puede separar los datos y tratarlos con estructuras propias de NestJS. Su función es reenviar esta información al servidor remoto y a su vez controlar el uso de la aplicación.

Figura 67.

Diagrama de comunicación por peticiones para el puerto serial.



Nota. Esta estructura es la base de la comunicación del sistema, todas las funciones posteriores cumplen el mismo principio.

Administración de los periféricos

Los periféricos del sistema son un teclado matricial, un escáner QR y una pantalla. El escáner QR es de carácter HID (Human Interface Device), es decir que el sistema lo reconoce automáticamente y facilita su implementación al funcionar como un teclado externo. Por otro lado, el teclado matricial no cuenta con un sistema Plug & Play, por lo que debe ser conectado a los pines GPIO del sistema como se muestra la tabla 27.

Tabla 27.

GPIO utilizados del teclado matricial

Pin físico Raspberry	Descripción
GPIO [37, 35, 33, 11]	Señales del teclado para definir a las filas
GPIO [40, 38, 36,32]	Señales del teclado para definir las columnas

Pin físico Raspberry	Descripción
5v	Fuente de alimentación para el teclado
GND	Pin de tierra para el teclado

El sistema reconoce el teclado mediante una librería dedicada que trabaja por eventos llamada “rpi-keypad”, permite ejecutar el mapeo de los pines GPIO de la Raspberry como entradas de un teclado matricial. Además, se necesita de una librería llamada “robotjs” que ejecuta a la programación de manera nativa al forzar el sistema a reconocerlo como HID.

Para usar el monitor, la Raspberry requiere de un cable de conexión micro HDMI (High Definition Multimedia Interface) a VGA (Video Graphics Array), además de varios cambios en su carpeta de configuración de inicio o “root” que se especifican en la siguiente tabla:

Tabla 28.

Comandos para configurar la pantalla

Comandos	Descripción
hdmi_force_hotplug=1	Fuerza a usar el puerto HDMI e intentará detectar la pantalla
hdmi_group=2	Grupo DTM (Display Monitor Timing) destinados a monitores de computadora
hdmi_mode=16	Representa las dimensiones y hercios en el grupo DTM
hdmi_drive=2	Alterna el voltaje de salida con el puerto HDMI y configura a la salida DVI (Digital Visual Interface)

Diseño de la estructura Backend-servidor remoto

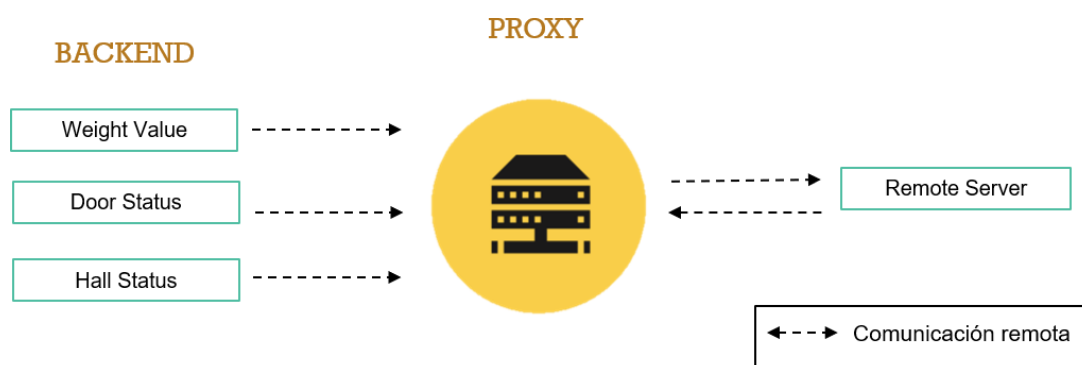
Esta comunicación tiene como principal recurso el uso de MQTT para en envió de datos al servidor. Se usa la estructura básica de microservicios importados de

NestJS, estos tienen una arquitectura que divide las funcionalidades del sistema en varios servicios independientes.

Estos microservicios se conectan al broker que interactúa directamente con el servidor, pero no con el usuario, como se muestra en la figura 68.

Figura 68.

Arquitectura de comunicación MQTT



Conexión del cliente MQTT.

NestJS permite administrar los servicios y controladores por medio de módulos que funcionan mediante microservicios. Mediante el método "ClientsModule", se transporta microservicios al crear una instancia cliente (objeto MQTT) con la finalidad de usar el transportador ("Transport"). Se registra un cliente con la función "register" cada vez que se inicializa el servidor local (Backend) en la Raspberry.

Para inyectar los servicios de "client-MQTT" se define el nombre del dominio como "TEST_CLIENT", posteriormente se usa el microservicio "Transport" de MQTT como canal de comunicación y finalmente se usan las opciones de conexión: usuario, contraseña y URL al servidor remoto. Esta última necesita la IP del dominio y el puerto de comunicación en estado "abierto", entonces su estructura es:

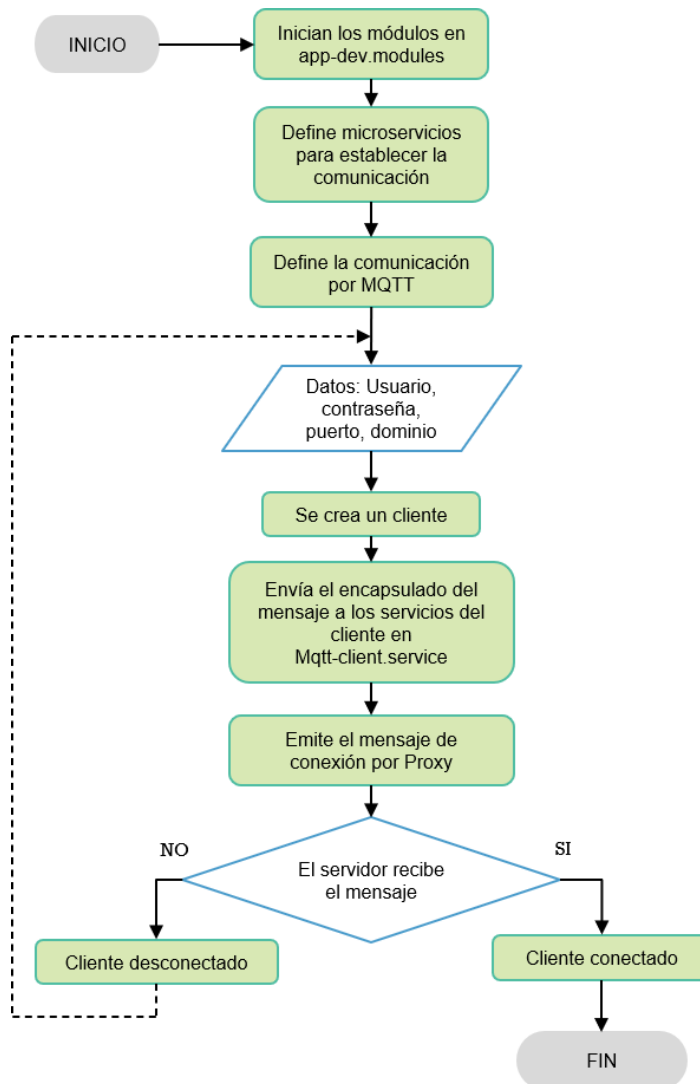
"mqtt://MQTT_DOMAIN:MQTT_PORT".

El servicio del cliente-MQTT recibe el encapsulado del mensaje con la arquitectura de "Transport".

Para enviar el mensaje usa el microservicio “ClientProxy”, se conecta mediante la función “connect” al servidor remoto y se genera un mensaje de respuesta en el servidor como se muestra en la siguiente figura.

Figura 69.

Diagrama de flujo de conexión del cliente MQTT



Nota. La línea entre cortada del cliente desconectado, representa a las reconexiones automáticas, una función propia del sistema MQTT en NestJS que no está programado manualmente.

Envío de datos por MQTT

El envío de datos por MQTT es en tiempo real y parte desde la obtención de datos en el puerto serial con los mensajes de estado de la cerradura, sensor Hall y peso. Se ejecuta en el apartado de servicios de Backend usando un constructor y se activa cada vez que tenga una trama entrante en el puerto serial. Según la estructura del mensaje de respuesta, se separa y se clasifica la información como:

Tabla 29.

Control de comando de respuesta

Comandos	Descripción	Canal MQTT
Lock Status	Se usa cuando el seguro está cerrado (“l”) o abierto (“u”)	sendLockerBoxLockStatus
Door Status	Responde al estado del sensor con los comandos “o” para abierta y “c” cuando está cerrada	sendLockerBoxDoorStatus
Box Weight	Se aparta el valor del peso al final de la trama	sendLockerBoxWeight

Nota. Para los comandos de la tabla se crean canales de comunicación MQTT como publicadores.

La información recibida pasa por MQTT-client que administra y ordena la arquitectura del mensaje a publicar en el servidor como: torre, casillero, acción; toda esta trama se envía al canal con el comando “.emit”, que es una propiedad del cliente.

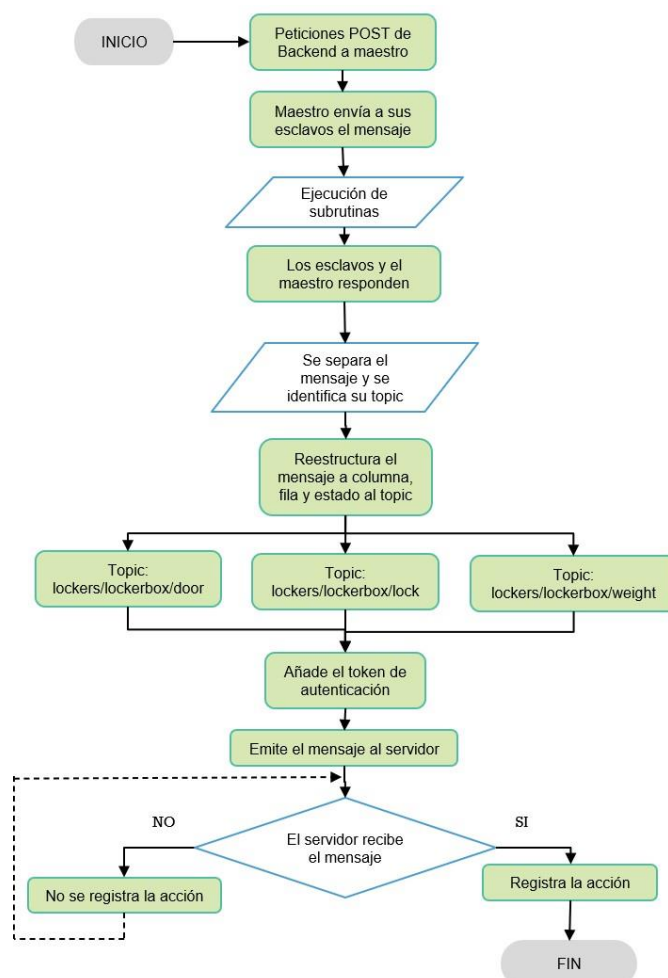
Para que el envío del mensaje sea seguro se utiliza un sistema de autenticación basada en token de acceso con un límite de 65535 bytes. Este se genera una vez y tiene características únicas que lo diferencia de los demás, puede ser una cadena especial donde se encuentra información del puerto a conectarse, permiso para el cliente, contraseña y usuario.

El proceso inicia con una petición al maestro, este envía a sus esclavos un mensaje y administra las respuestas en la Raspberry. La petición de tipo POST se ejecuta a la rama api/execute-command y espera que se impriman los mensajes de tipo string ya clasificados en el puerto serial de Backend.

Finalmente, estos mensajes una vez ya clasificados, llegan al servidor gracias a que el token se envía junto al mensaje para que se autorice la publicación de información y se pueda visualizar la estructura del mensaje que define la posición del locker y sus variables.

Figura 70.

Diagrama de secuencia para el envío de datos por MQTT



Nota. Cuando el servidor no registra una conexión, el último mensaje que el servidor local en Backend a enviado por MQTT, se mantiene en el topic para ser leído por el servidor una vez se reestablezca la conexión.

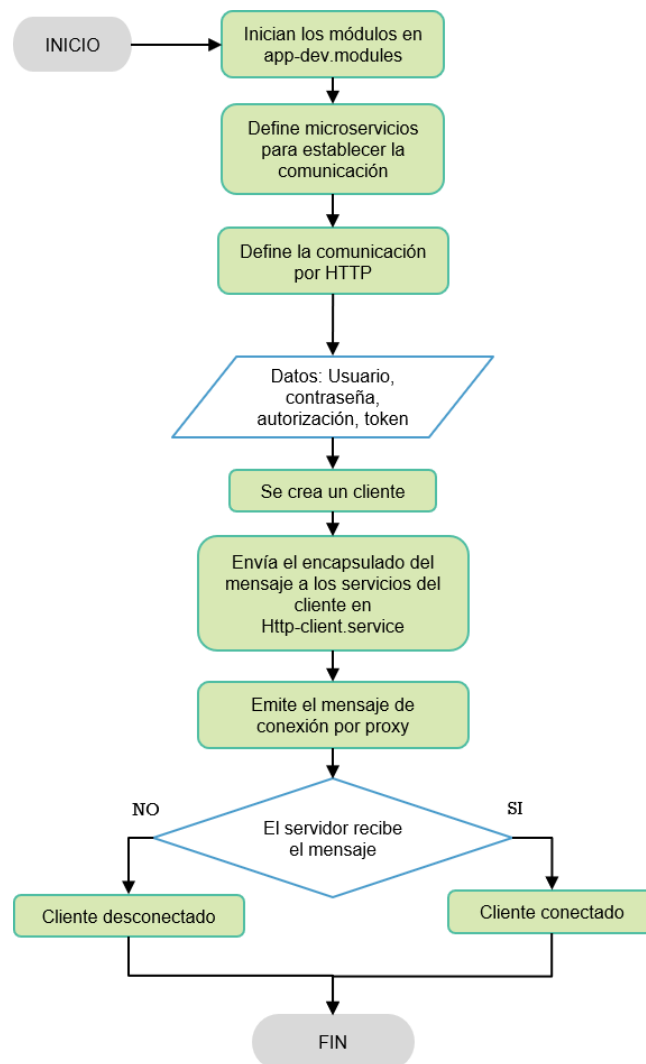
Modo administrador

Para el modo administrador se prueba los estados de conexión por MQTT y HTTP. La conectividad se ejecuta junto a los servicios de arranque del programa en Backend, es decir que al iniciarse la aplicación de Nest también se ejecutan los módulos HTTP y las dependencias del cliente en MQTT.

Conexión del servidor por HTTP

Figura 71.

Diagrama de flujo de conexión del cliente HTTP



Nota. La conexión por HTTP necesita la trama de autorización y el token en el cuadro de proceso.

Figura 72.

Servicios inicializados en el servidor Backend

```
[Nest] 1899 - 20/06/2022, 16:01:55 LOG [NestFactory] Starting Nest application...
[Function (anonymous)]
[Nest] 1899 - 20/06/2022, 16:01:55 LOG [InstanceLoader] ClientsModule dependencies
initialized +280ms
[Nest] 1899 - 20/06/2022, 16:01:55 LOG [InstanceLoader] HttpModule dependencies ini
ialized +1ms
[Nest] 1899 - 20/06/2022, 16:01:55 LOG [InstanceLoader] ConfigHostModule dependenci
es initialized +1ms
[Nest] 1899 - 20/06/2022, 16:01:55 LOG [InstanceLoader] ConfigModule dependencies i
nitialized +2ms
[Nest] 1899 - 20/06/2022, 16:01:55 LOG [InstanceLoader] AppDevModule dependencies i
nitialized +3ms
serial port open
cb constructor
```

En el lado del servidor se tienen los mensajes de respuesta a la conexión de HTTP y MQTT como se muestran a continuación:

Figura 73.

Conexión verificada en el servidor por MQTT y HTTP

```
MQTT
1653750681: Client mqttjs_babb1908 closed its connection.
1653750683: New connection from 172.19.0.1:50182 on port 1883.
1653750683: New client connected from 172.19.0.1:50182 as mqttjs_b6a3c14e (p2, c1, k60, u'wolfsa').
1653750710: Client mqttjs_ec822747 closed its connection.
1653753104: New connection from 157.100.173.34:35212 on port 1883.
1653753104: New client connected from 157.100.173.34:35212 as mqttjs_3b4f1902 (p2, c1, k60, u'rasplocker').

HTTP
serial port open
cb constructor
(node:5285) ExperimentalWarning: The Fetch API is an experimental feature. This feature could change at any time
(Use `node --trace-warnings ...` to show where the warning was created)
{ statusCode: 200, data: true, message: 'OK' }
```

El estado de reconexión en MQTT permite visualizar en el servidor remoto cuando el sistema de desconecte y se vuelva a conectar con datos de hora, fecha y nombre del servicio conectado.

Figura 74.

Conexión verificada en el broker por HTTP.

```

=====
PM2 | 2022-05-25T01:42:36: PM2 log: App [dev
-wolf-server:0] starting in -fork mode-
PM2 | 2022-05-25T01:42:36: PM2 log: App [dev
-wolf-server:0] online

0|dev-wolf-server | lockerbox/locker-06RI01/admin
0|dev-wolf-server | [Nest] 437645 - 06/17/2022,
9:29:59 PM [HTTP] POST /mqtt/lockerbox/9e4246b6-0
03c-4d1f-bf87-8ea323ab9108/execute-command 201 Crea
ted
0|dev-wolf-server | Locker Box LOCK status updated
0|dev-wolf-server | lockerbox/locker-06RI01/admin
0|dev-wolf-server | [Nest] 437645 - 06/17/2022,
9:30:23 PM [HTTP] POST /mqtt/lockerbox/9e4246b6-0
03c-4d1f-bf87-8ea323ab9108/execute-command 201 Crea
ted
0|dev-wolf-server | Locker Box WEIGHT value status
updated
0|dev-wolf-server | lockerbox/locker-06RI01/admin
0|dev-wolf-server | [Nest] 437645 - 06/17/2022,
9:32:16 PM [HTTP] POST /mqtt/lockerbox/9e4246b6-0
03c-4d1f-bf87-8ea323ab9108/execute-command 201 Crea
ted
0|dev-wolf-server | Locker Box LOCK status updated

```

Para administrar el sistema de pesaje desde el servidor, se necesita usar los canales de comunicación Backend – maestro/esclavo. Las peticiones de control se hacen con HTTP desde el servidor remoto al computador central y se usa el canal de comunicación con el puerto serial para ejecutar las acciones.

Como HTTP espera una respuesta, es óptimo utilizar MQTT para la información que se envían desde los casilleros al servidor remoto para actualizar la base de datos y monitorear los casilleros en tiempo real.

Figura 75.

Comandos de respuesta en el puerto serial en el Backend NestJS

```

[Nest] 2822 - 25/06/2022, 09:47:09
[LOGIN] {
  user: {
    email: 'rasplocker@gmail.com',
    firstName: 'Locker',
    firstLastName: 'A',
    avatar: 'https://www.pngitem.com/pi
  }
}
user-0000_10
Chapa: [ '0000', '1', 'u' ]
Puerta: [ '0000', '1', 'o' ]
Puerta: [ '0000', '1', 'c' ]
Chapa: [ '0000', '1', 'l' ]
Peso: [ '0000', '1', '2999' ]

```

Figura 76.*Registro de los cambios de estado del locker en el servidor remoto*

▶	2022-06-25T10:51:23.566-05:00	Jun 25 15:51:23	ip-172-31-93-7	web: lockerbox/locker-06AMBT01/admin
▼	2022-06-25T10:51:25.067-05:00	Jun 25 15:51:23	ip-172-31-93-7	web: #033[32m[Nest] 16147 - #033[39m06/25/2022, 3:51:23 PM #033[32m LOG#033[39m #033[39m#033[32mPOST /mqtt/lockerbox/ /execute-command 201 Created#033[39m
▶	2022-06-25T10:51:26.067-05:00	Jun 25 15:51:24	ip-172-31-93-7	web: Locker Box LOCK STATUS status updated
▶	2022-06-25T10:51:30.516-05:00	Jun 25 15:51:25	ip-172-31-93-7	web: Locker Box DOOR status updated
▶	2022-06-25T10:51:36.322-05:00	Jun 25 15:51:35	ip-172-31-93-7	web: Locker Box DOOR status updated
▶	2022-06-25T10:51:40.516-05:00	Jun 25 15:51:36	ip-172-31-93-7	web: Locker Box LOCK STATUS status updated
▶	2022-06-25T10:51:45.516-05:00	Jun 25 15:51:40	ip-172-31-93-7	web: Locker Box WEIGHT value status updated

Nota. En la línea uno se muestra el nombre del casillero, en la línea dos y tres se verifica la conexión y se registra el comando de inicio de secuencia, las siguientes líneas establecen el estado del locker.

Una vez que los comandos de respuesta se envían al servidor remoto, este registra los cambios de estado del locker y permite llevar un registro de acciones para cada casillero, además de reconocer que torre y que casillero se está monitoreando, también es capaz de identificar desde que punto de recepción de paquetes se están enviando datos, pues la Raspberry tiene registrado el nombre del locker; esto permite definir varios puntos de entrega de paquetes y potencia el monitoreo geográfico de los casilleros.

El modo administrador se limita a conectar el sistema con el servidor remoto y administrar al locker por comandos de forma remota.

Modo de usuario

El modo usuario controla las acciones del locker con una secuencia de proceso, el HMI muestra las etapas al usuario en tiempo real, esta secuencia es:

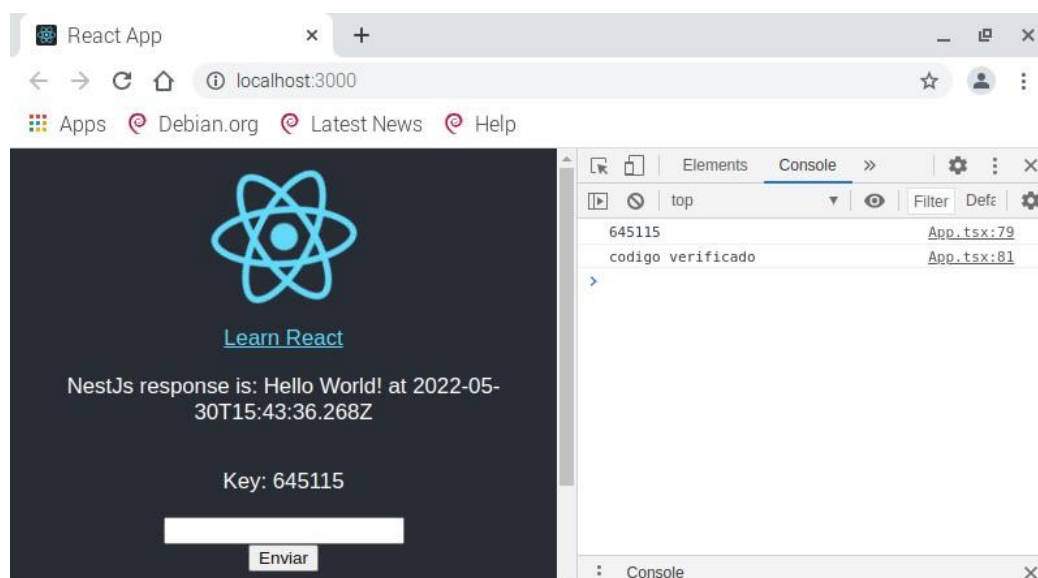
- Con HTTP al servidor remoto se valida el usuario, locker y datos de servicio de paquetería.
- Se define la navegación en función del diagrama de secuencia de proceso

- Se ejecutan los comandos de control desde el servidor local (Backend) al maestro y esclavo, se emplea MQTT para la actualización de estados de locker.
- Con HTTP se verifica el cobro de los servicios con la pasarela de pagos.

La secuencia de usuario del HMI depende del sistema de respuestas, es decir que el cliente necesariamente debe pasar por el proceso de validación, ya sea de datos, comandos o navegación, para que pueda usar el servicio como se muestra en la figura.

Figura 77.

Autenticación de código de verificación



Nota. La verificación usa el protocolo HTTP y lo hace ingresando el código por teclado y por el escáner QR.

La navegación, incluye la respuesta del hardware (teclado, escáner QR), que se traducen en acciones de usuario a comandos, también al sistema de comunicación del puerto serial con RS 485, con MQTT y HTTP. Ante desconexiones, el sistema MQTT permite que el último mensaje enviado al broker se mantenga, hasta que el sistema se reconecte y se verifique que el mensaje llegó. En HTTP no se tiene un método de seguridad ante desconexiones, debido a esto, el sistema de respuesta de mensajes es la única forma de autenticar la comunicación.

Figura 78.

Comando de respuesta en el servidor remoto por MQTT con token

```

0|dev-wolf-server | [Nest] 296179 - 06/08/2022, 3:03:27 PM [Ord
ersLoaded in memory] 0
0|dev-wolf-server | Lock: {
0|dev-wolf-server |   column: '0000',
0|dev-wolf-server |   row: '1',
0|dev-wolf-server |   status: 'unlocked',
0|dev-wolf-server |   token: 'eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.

0|dev-wolf-server | }
0|dev-wolf-server | Weight: {
0|dev-wolf-server |   column: '0000',
0|dev-wolf-server |   row: '1',
0|dev-wolf-server |   weight: '2580',
0|dev-wolf-server |   token: 'evJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.

0|dev-wolf-server | }
0|dev-wolf-server | Door: {
0|dev-wolf-server |   column: '0000',
0|dev-wolf-server |   row: '1',
0|dev-wolf-server |   status: 'open',
0|dev-wolf-server |   token: 'eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.

```

Nota. El token de seguridad permite que el mensaje sea cifrado y llegue de forma segura a su servidor de destino

Las rutas del HMI pueden verse comprometidas si se tienen acceso a la IP de la Raspberry y el puerto de Frontend, sin embargo, se pueden proteger con niveles de seguridad avanzadas de la librería “React Router Dom”. En este apartado se usa validación básica de usuario y contraseña para acceder a la ruta, muy parecido al método de conexión en MQTT.

Construcción del HMI para el usuario

En el capítulo anterior se definió una secuencia para el proceso de envío de paquetes usando HMI. El proceso del sistema de paquetería se divide en dos funciones, uno como punto de envío de un paquete y otra como punto de retiro. En este sentido se establece un orden de operación que el usuario debe seguir para tener éxito y manejar de forma correcta el sistema.

Se planteó como mínimo cuatro páginas para un correcto funcionamiento del HMI, la primera: es la página principal que tendrá información de la empresa y será

visible en todo momento por el usuario mientras no se ejecute ninguna acción por teclado.

La funcionalidad de los periféricos del HMI requieren de configuraciones propias, para el teclado matricial es necesario ajustarlo de tal forma que las teclas no se muevan y den falsos negativos al pulsar un botón, por otro lado, el escáner QR necesita establecer una configuración de tiempo de espera tras leer un código y definir una sensibilidad que se ajuste a los parámetros del sistema.

Con estas configuraciones la funcionalidad del HMI permite al cliente usar el locker sin contar posibles errores propios de los elementos periféricos como serían fallo de la placa de control, deterioro de los cables, etc.

Figura 79.

Página principal de HMI



En apartados anteriores se especificó que el teclado matricial puede detectarse de forma nativa en el sistema operativo de la Raspberry, por lo tanto, React puede limitarse a leer eventos de teclado con la función “useKeyPress”, que es un microservicio que detecta si una tecla es pulsada para luego ejecutar una acción.

Este microservicio se usa en “UseEffect” para que cada vez que el usuario pulse la tecla enter, la página lo detecte como un evento y pase a la fase de navegación sin cambiar de página. Para este proceso las pantallas emergentes sirven para dar información sobre las opciones que tiene el usuario de enviar o retirar paquetes empleando el teclado como se muestra a continuación:

Figura 80.

Selección entre recoger y enviar paquetes



Al pulsar las teclas 4 o 6 se selecciona el servicio y se pasa a una siguiente etapa, sí en un tiempo de 10 segundos no se registra actividad alguna, la fase de navegación se suspende y regresa a la página principal.

Para que la aplicación pueda navegar entre páginas necesita de la librería “react-router-dom”, su instalación es muy sencilla y admite varios tipos de Backends. Su funcionamiento se basa en emplear rutas o links en el puerto donde se encuentra corriendo la aplicación de React.

Se importan los microservicios de “BrowserRouter”, con esto podemos crear una sección de direcciones que define las páginas que se van a crear para la navegación en App.tsx, como son:

Figura 81.

Direcciones de las páginas del HMI

```
import Home from './pages/Home';
import ReactHome from './pages/ReactHome';
import SendPackages from './pages/SendPackages';
import PickupPackages from './pages/PickupPackages';
```

Y a su vez se definió su ruta en el servidor local con backend los componentes “Routes” y “Route”, donde la primera agrupa todas las rutas y en la segunda define la URL y el elemento a renderizar.

Figura 82.*Asignación de las URL*

```
function App() {
  return (
    <Router>
      <Routes>
        <Route path="/" element={<ReactHome />} />
        <Route path="/home" element={<Home />} />
        <Route path="/pickup-packages" element={<PickupPackages />} />
        <Route path="/send-packages" element={<SendPackages />} />
      </Routes>
    </Router>
  );
}
```

Para hacer uso de la navegación se necesita el microservicio “UseNavigate” que redirecciona la página actual a la ruta especificada, esto se ejecuta en la programación del teclado matricial y la secuencia para el cliente.

La siguiente página que se muestra es la elección de ingreso por código o el escáner QR.

Figura 83.*Página de selección entre código y escáner QR*

El dato de autenticación va a ser único por cada cliente y su representación puede ser en un código numérico o QR. La verificación con el servidor se va a realizar con HTTP, es decir que la aplicación va a enviar el código que ingrese el usuario hacia backend para que este confirme en la ruta de comunicación con el servidor y espera su respuesta de confirmación.

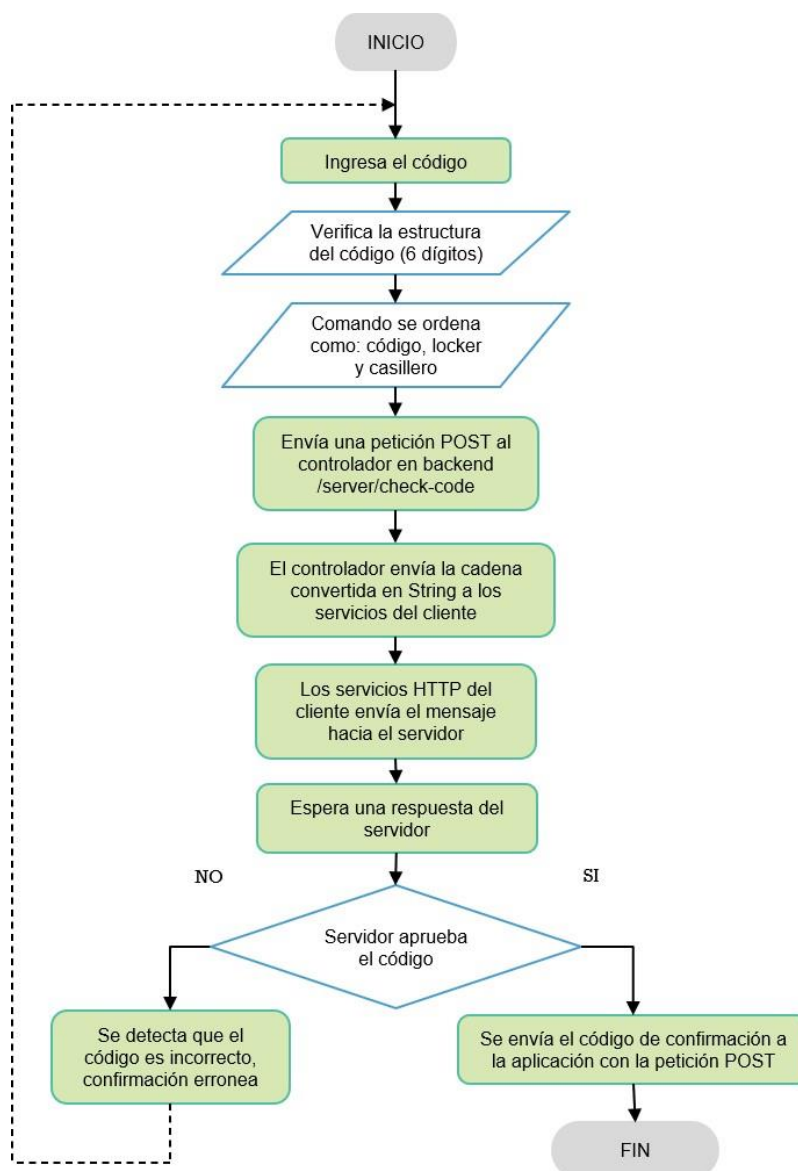
La respuesta del servidor es de carácter booleano es decir que permite o no permite el acceso al usuario, para ello el sistema muestra en pantalla alertas que

informan al cliente el estado de la autorización, si es válida sigue el proceso y en caso contrario permite el reingreso de la clave de acceso.

Una vez que se verifica el envío del paquete, se ejecuta el comando de inicio del proceso de recepción para el casillero en el puerto serial de la raspberry y este mismo puerto espera la respuesta del maestro como se especificó anteriormente.

Figura 84.

Diagrama de secuencia para el ingreso de código de verificación



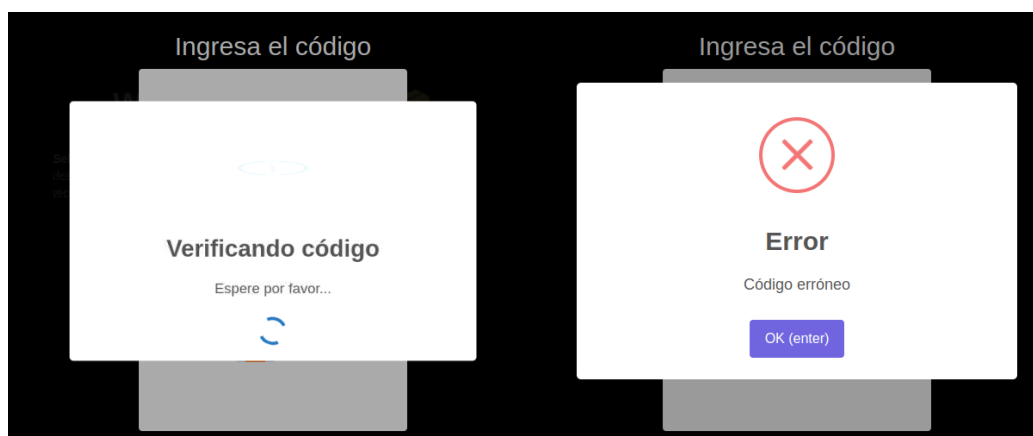
Nota. El ingreso de código puede ser por teclado o por el escáner QR.

Para las alertas de verificación, se utiliza “SweetAlert”, que es una biblioteca de Java compatible con React. Esta puede crear modales (ventanas emergentes) que están en el interior de una función y pueden ser llamados por los eventos.

La siguiente imagen muestran cómo funciona la verificación del código ingresado por teclado usando modales.

Figura 85.

Pantallas de alerta para la verificación del código



Nota. La figura muestra las pantallas emergentes de alerta cuando se autoriza el acceso al usuario.

Una vez que se finaliza el proceso de recepción de paquetes en el locker, pasa a una nueva etapa de verificación de datos y cobro del servicio, en esta etapa usa los sistemas de verificación de código y envío de comando al maestro-esclavo para iniciar la pasarela de pagos y mostrar en pantalla los nuevos valores a cobrar como se muestra en la siguiente figura.

Figura 86.

Pantalla de información para el nuevo tarifado.

Tarifa

Usuario: Vicente Ochoa **Correo electrónico:** jose.ochoa.dev@gmail.com
Ruta: Av. Daniel León Borja, Ambato - Av. los chásquis 1745-1651, Riobamba

Paquetes:	Precio	Peso	Contenido
Caja CP-1	\$4.25	1.17 Kg	Carton con comida

Costo de servicio: \$1.50
Descuento (cupón): \$0.00
Total: \$5.75

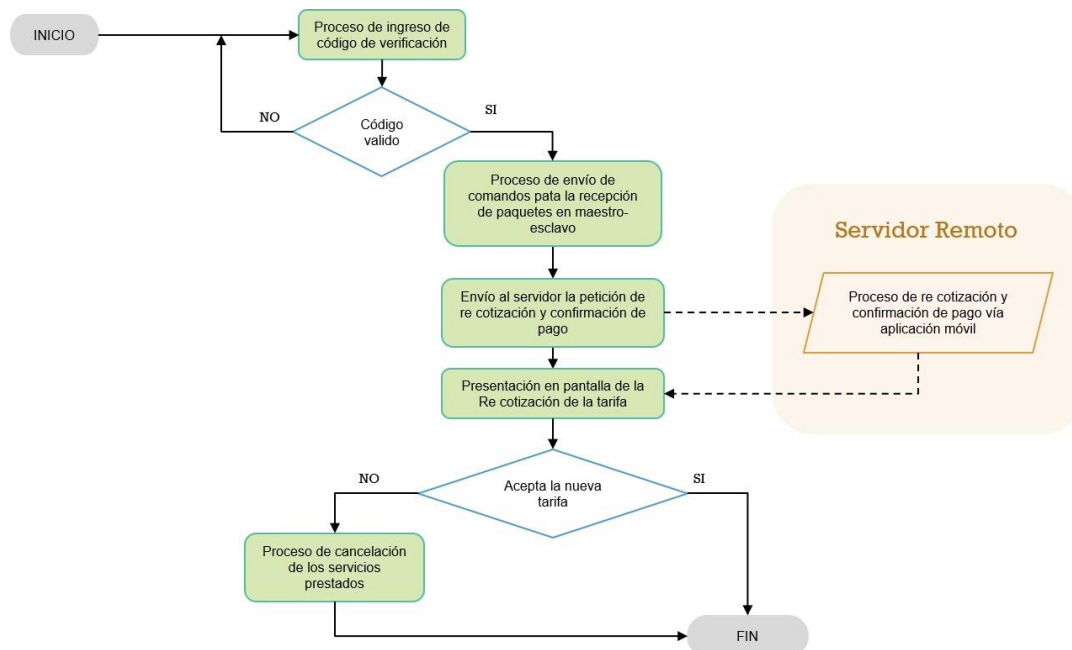
Nota: recuerda que el casillero vuelve a pesar el paquete, si crees que esta averiada la balanza reporta el fallo a soporte técnico.

Aceptar tarifa (enter)
Rechazar (esc)

Nota. esta ventana de mostrará en pantalla hasta que el usuario acepte o no el servicio.

Figura 87.

Diagrama de secuencia para la verificación del pago



Nota. La petición al servidor de pago se hace desde backend al servidor remoto y la cancelación de los servicios prestados hacen referencia al cobro por reserva de casillero.

La pasarela de pagos inicia cuando los paquetes se encuentren asegurados en el casillero y el servidor remoto registre estas acciones en su base de datos. Al finalizar este proceso se envía una petición de re cotización y confirmación de pago por HTTP, se espera una respuesta de la nueva tarifa para mostrar en pantalla, con la confirmación o cancelación del servicio para finalizar el proceso.

El proceso de re tarifado del servidor remoto forma parte de una secuencia de nivel administrativo que no está contemplado en los alcances de esta tesis, sin embargo, es oportuno mencionar que se requiere usar valores reales de peso del paquete brindados por el sistema de galgas del casillero, para recalcular el valor de la tarifa desde un peso base, con esto el usuario puede ver en la pantalla del locker principal los nuevos valores de cobro.

Capítulo V

Pruebas y análisis de resultados

Para poder evaluar la seguridad del locker es necesario subdividir en dos pruebas diferentes, una mecánica y otra electrónica.

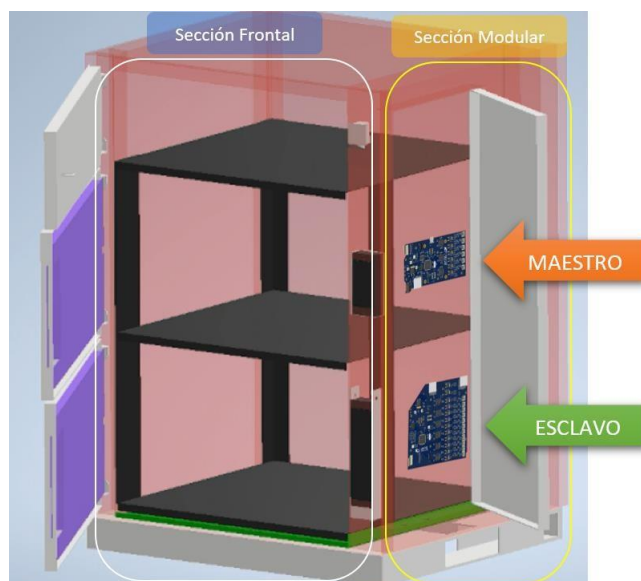
Para probar la seguridad mecánica se construyó un prototipo de armazón a escala 1:3 con la finalidad de evaluar el cierre en cada una de las puertas; de la misma manera verificar las medidas y evitar el ingreso de artefactos para forzar las cerraduras.

La seguridad electrónica va a evaluar el sistema de comunicación maestro-esclavo mediante el envío de tramas en la secuencia de proceso de envío de paquetes y tareas de administrador que validen el modularidad del sistema

La tabulación de los datos recopilados se lo hace mediante pruebas de hipótesis usando el método de Chi-Cuadrado por asociación, para ello se empleó el programa de análisis de datos Minitab que genera un reporte con la evaluación de resultados.

Figura 88.

Prototipo a escala de prueba



Nota. Se implementó las cerraduras pequeña, mediana y grande con la finalidad de probar cual ofrece mejor seguridad.

Pruebas de seguridad mecánica

Para poder verificar la seguridad se calificó la dificultad con la que se puede abrir la puerta, las pruebas se van a realizar en la sección frontal del armazón con el ingreso de destornilladores tipo Phillips con las medidas:

Tabla 30.

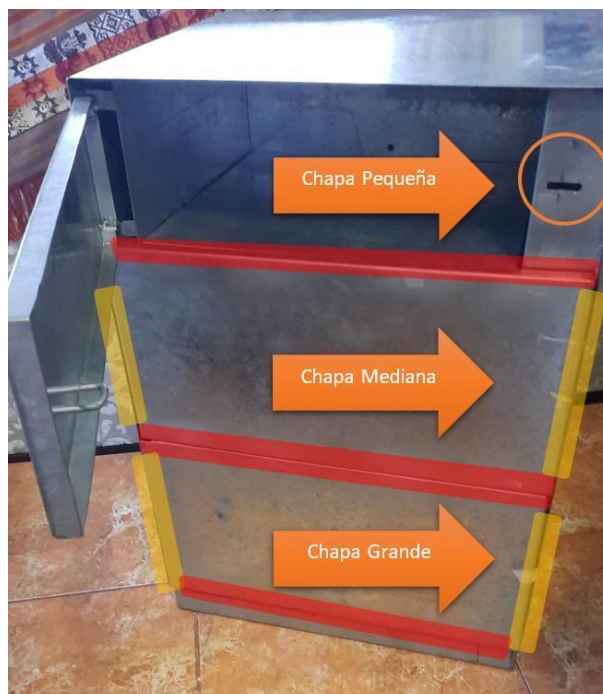
Datos de los destornilladores para las pruebas de seguridad

Destornillador	Medida de punta	Longitud de punta	Longitud total
Pequeño	0,5 x 3,0 mm	75 mm	160 mm
Grande	1,0 x 6,0 mm	150 mm	265 mm

Se probó el ingreso de los destornilladores alrededor de los casilleros en las áreas superior, inferior y laterales como se muestra en la figura 89:

Figura 89.

Áreas de estudio para la seguridad mecánica del locker



Nota. Las secciones del casillero están representadas por colores, de esta forma el color rojo representa las uniones inferiores/superiores y las franjas amarillas representan las uniones laterales.

Las pruebas se realizaron con 5 personas, cada uno hizo dos intentos para aperturar la puerta del casillero haciendo palanca con los destornilladores mientras las cerraduras están energizadas y posteriormente sin energía, tomando en cuenta que la fuerza promedio de halar de una persona de 20-29 años es de 46 kg F o 392.4 N, con estas pruebas se obtuvo la tabla siguiente:

Tabla 31.

Promedio de prueba de cerraduras

Pruebas	Cerraduras					
	Energizada			Sin energizar		
	Pequeña	Mediana	Grande	Pequeña	Mediana	Grande
	NC	NA	NA	NC	NA	NA
1	+	+	+	-	+	+
2	+	+	+	-	+	+
3	+	+	+	-	+	+
4	+	+	+	-	+	+
5	+	+	+	-	+	+
T. Positivos	5	5	5	0	5	5
T. Negativos	0	0	0	5	0	0
Porcentaje	0%	0%	0%	100%	0%	0%

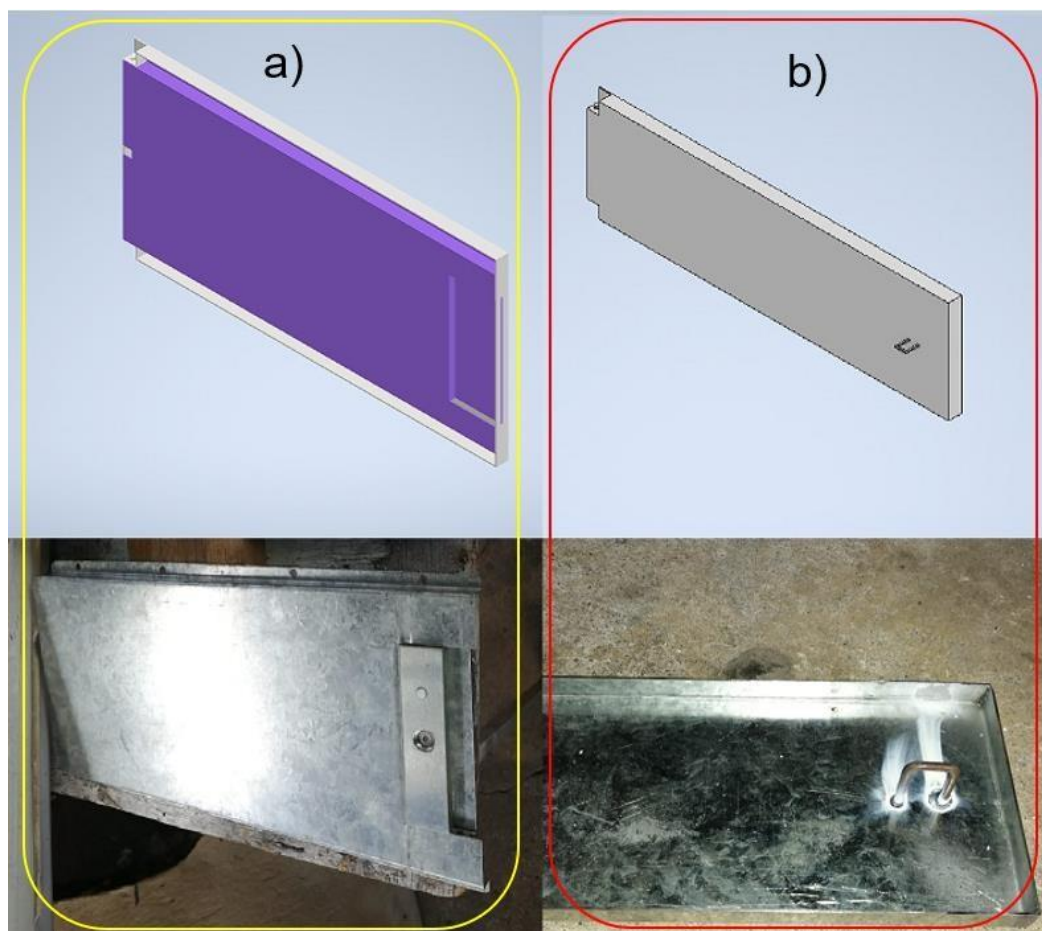
Nota. Las pruebas se evalúan con la marca + si el usuario puede abrir la puerta, en caso contrario se aplica una marca.

La cerradura pequeña por el hecho de que su principio es NC y dispone de un seguro a manera de émbolo, funciona en un 100% ofreciendo mayor seguridad debido a que solo necesita el pulso eléctrico para la liberación del seguro.

El funcionamiento de la cerradura mediana y grande no es aceptado por el hecho de que su principio es NA, ya que al momento de realizar las pruebas de evidenció que:

- Necesariamente debe estar alineada la ferrita con la cerradura, sin ningún desfase: pese a que en las pruebas se engancha la ferrita a la cerradura, debido al error humano en la construcción existe un desfase de 2 mm que al momento de cerrar la puerta se vuelve fácil de liberar.
- La ferrita debe ocupar toda el área magnetizada de la cerradura: si la ferrita abarca mayor área que de la cerradura, presenta un desfase menor a un milímetro del sobrante del armazón, por lo cual no se engancha correctamente y es fácil liberarla.
- El consumo de energía para este tipo de cerraduras es elevado, debido a que necesitan estar todo el tiempo energizadas para poder cerrar la puerta. Siendo este un problema, ya que al momento de existir un posible corte de luz automáticamente los casilleros quedan abiertos.
- Para el diseño mecánico emplean mayor material: Debido a que su funcionamiento depende de dos partes, para la construcción de la puerta se empleó un sistema de 2 planchas a manera de bandeja con la finalidad de poder asegurar la ferrita

Según las pruebas no destructivas de abrir la puerta haciendo uso de objetos externos al casillero dan como resultado que la estructura es bastante robusta desde la parte frontal, por lo cual el diseño propuesto reacciona de manera favorable ante herramientas mecánicas de uso manual, si la herramienta es pequeña tiene mayor acceso a la puerta sin embargo la seguridad no se ve comprometida ya que no puede ejercerse mucha fuerza. Del mismo modo si la herramienta es mediana o de mayor tamaño, el acceso a la puerta no se ve comprometida en ningún sentido.

Figura 90.*Diseño mecánico para puerta*

Nota. a) La figura de la izquierda es el diseño implementado para la prueba con cerraduras NA, b) es el diseño implementado para cerraduras NC.

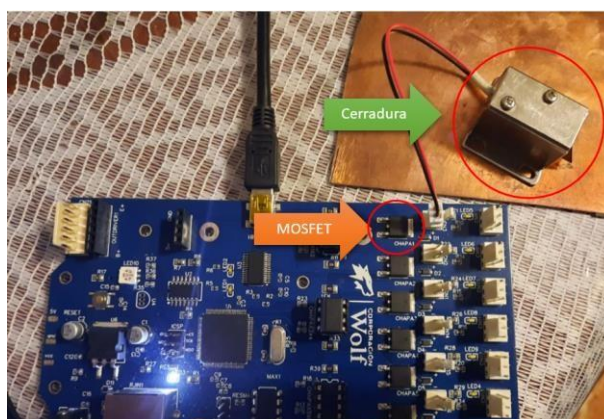
Pese a que la prioridad de implementación a las cerraduras NC fue inferior al de las NA; se evidencia que las cerraduras NC son más fácil de implementar y solucionan el problema de seguridad en el supuesto caso que exista un corte de luz, además requiere menor material para su implementación y ofrecen la seguridad suficiente a los paquetes en caso de que se vulnere el sistema. Se optó por cambiar el diseño a cerraduras pequeñas en todos los cajones (pequeños, medianos y grandes).

Para aumentar la seguridad se dispuso un dobléz en la parte frontal de la percha de 2cm con la finalidad de evitar el ingreso de documentos que no están registrados y autorizados por el sistema.

Figura 91.*Doblez de la percha en la zona frontal****Evaluación de la cerradura pequeña***

Debido a que en el apartado anterior se reestructuró el sistema de seguridad de las puertas; se sometió a pruebas las cerraduras pequeñas.

Se estableció, mediante tren de pulsos, la activación y desactivación de las cerraduras pequeñas por intervalos de 40 segundos a lo largo de 4 horas para poder evidenciar el comportamiento a largo plazo. Por lo tanto, la cerradura tuvo 45 conmutaciones por cada intervalo de 30 min.

Figura 92.*Prueba de cerradura pequeña*

Nota. Se puso a prueba el circuito de potencia con la finalidad de obtener la temperatura a lo largo del tiempo, tanto del mosfet como de la cerradura.

Tabla 32.*Respuesta a lo largo del tiempo de seguridad electrónica*

Tiempo de funcionamiento	Funcionalidad del sistema		Temperatura de cerradura °C			Temperatura de mosfet °C			Efectividad
	Si	No	Alta	Medio	Bajo	Alta	Medio	Bajo	
30 min	45	0	0	24	21	0	0	45	100%
60 min	45	0	40	5	0	0	0	45	100%
90 min	45	0	42	3	0	0	0	45	100%
120 min	45	0	41	4	0	0	0	45	100%
150 min	45	0	39	6	0	0	0	45	100%
180 min	45	0	41	4	0	0	0	45	100%
210 min	45	0	41	4	0	0	0	45	100%
240 min	45	0	40	5	0	0	0	45	100%
Total	360	0	284	55	21	0	0	360	100%
P. Individual (%)	100	0	79	15,28	5,83	0	0	100	100%

Nota. Para evaluar el funcionamiento de la cerradura se define el rango de temperatura según (Plus Electronics, 2022) como bajo ($T \leq 25^{\circ}\text{C}$), medio ($25 < T \leq 40^{\circ}\text{C}$) y alto ($T > 40^{\circ}\text{C}$), el mismo modo se considera que el trabajo normal para el mosfet IRFZ44N entra en el rango de temperatura entre 25°C a 70°C según (VBsemi, 2022), de esta forma se emplea los rangos como bajo ($T \leq 25^{\circ}\text{C}$), medio ($25 < T \leq 50^{\circ}\text{C}$) y alto ($T > 50^{\circ}\text{C}$).

En base a los resultados obtenidos por la activación de la cerradura en periodos prolongados de tiempo, se pudo determinar que, pese a que la cerradura tiene un incremento de temperatura superior al del mosfet, esta sigue trabajando normalmente. La implementación de la cerradura se acepta debido a que las condiciones de trabajo no serán exigentes, donde la mayor parte del tiempo permanecerá desactivada y su activación será como máximo 10 segundos.

Comprobación de hipótesis de seguridad

- Hipótesis nula

Ho = El rango de temperatura que alcanza la cerradura y mosfet no garantiza la seguridad al sistema

- Hipótesis válida

Hi = El rango de temperatura que alcanza la cerradura y mosfet garantiza la seguridad al sistema

Margen de error considerada= 5%

Margen de error tabla (α) = 0.05

Grados de libertad = 2

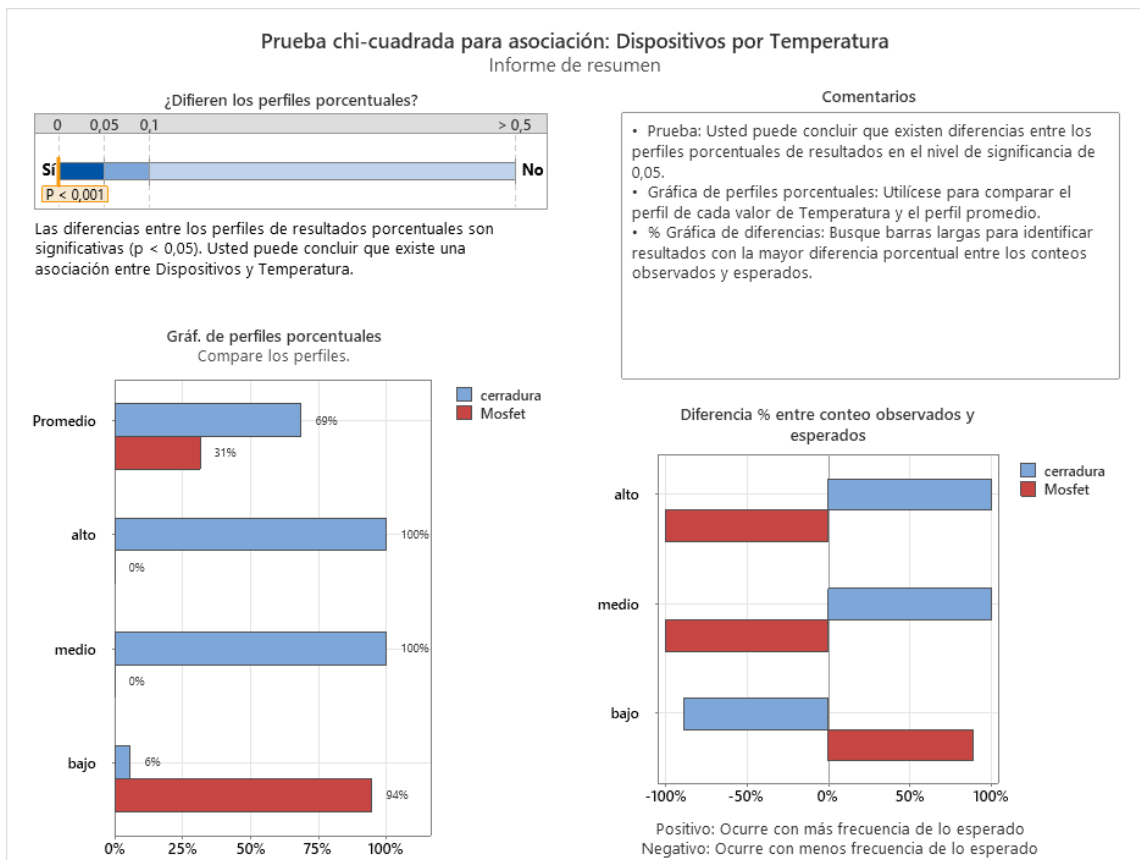
Tabla 33.

Tabla de datos para el chi-cuadrado

	Alto	Medio	Bajo	Total
Cerradura	284	55	21	360
Mosfet	0	0	360	360
Total	284	55	360	720

Figura 93.

Prueba chi-cuadrada para asociación: Dispositivos por temperatura



Según el análisis de Minitab, los valores obtenidos definen que los valores esperados de temperatura media y alta hagan referencia en su totalidad a la cerradura, por otro lado, se espera en un 80% que la temperatura baja haga referencia al mosfet.

Tabla 34.


Tabla de conteos esperados y observados para dispositivos

Conteos observados y esperados

	alto		medio		bajo	
	Obs	Exp	Obs	Exp	Obs	Exp
cerradura	284	142	55	28	21	191
Mosfet	0	142	0	28	360	191
Total	284		55		381	

Los conteos esperados deben ser por lo menos 2 para asegurar la validez del valor p de la prueba.

Figura 94.*Tarjeta de informe para Dispositivos por temperatura*

Tarjeta de informe		
Verificar	Estado	Descripción
Validez de la prueba		Todas las muestras son lo suficientemente grandes para obtener suficientes conteos esperados. El valor p de la prueba debería ser exacto.

$$p < 0.001$$

$$p > \alpha = H_0 \text{ es válida}$$

$$p \leq \alpha = H_1 \text{ es válida}$$

$$0.001 < 0.05 = H_1 \text{ es válida}$$

La hipótesis fue verificada por el método de distribución chi-cuadrado, con un margen de error de 5% considerando el ruido del sistema y la pérdida de calor por el efecto Joule en la placa y en la cerradura. Teniendo como resultado que la vida útil de la placa es mayor que elementos de entrada por lo tanto se comprueba la necesidad de un diseño “Plug and Play” tanto en el apartado electrónico como en el mecánico.

Pruebas de comunicación maestro-esclavo

Para las pruebas de las placas se planteó el envío de tramas para poder interactuar entre maestro y esclavo: siendo las pruebas

- Prueba 1: A través del puerto serial del maestro se envía una letra (1 carácter) y el esclavo imprime la letra recibida (1 carácter).
- Prueba 2: Se emplea un DIP switch para establecer un ID al esclavo (4 caracteres), visualizando este ID a través del puerto serial del maestro (4 caracteres).
- Prueba 3: El maestro a través del puerto serial ingresa el ID (4 caracteres) de uno de los dos esclavos, el esclavo solicitado envía al maestro la confirmación (5 caracteres).
- Prueba 4: Se establece una rutina de usuario para el sensado del paquete donde el maestro solicitará la apertura del casillero indicado (7 caracteres), se

depositará el peso, se cerrará el casillero y retornará el promedio del peso (46 caracteres total).

- Prueba 5: se establecen comandos de administrador (7 caracteres) para obtener valores puntuales de peso en cada locker y realizar acciones para cada esclavo (abrir puertas de casilleros, abrir puerta de mantenimiento, etc.). Dónde la secuencia para retornar todos los pesos imprime en promedio 140 caracteres.
- Prueba 6: se establecen comandos de administrador (7 caracteres) donde el maestro escaneará cuantos esclavos se encuentran conectados. Preguntará a cada esclavo para que le retorne su id, tomando en cuenta que responderá 4 caracteres cada uno.

Tabla 35.

Pruebas de comunicación

N° Prueba	Caracteres (maestro)		Caracteres (esclavo)		Tiempo de secuencia
	Envía	Recibe	Envía	Recibe	
1	1	0	0	1	<1 seg
2	0	4	4	0	<1 seg
3	4	5	5	4	1 seg
4	7	46	46	7	3 min
5	7	140	140	7	1 seg
6	7	4	4	7	5 seg por esclavo

En base a los resultados obtenidos se puede constatar que el tiempo que emplean maestro-esclavo para la comunicación depende de la tarea asignada, de igual manera se pudo verificar la fiabilidad para el retorno de caracteres mediante la implementación de software serial.

Debido a que la función de administrador requiere de mayor retorno de caracteres por parte del esclavo, se realizaron múltiples ensayos para la prueba 5 con

la finalidad de corroborar la fiabilidad de la trama enviada por el esclavo hasta el servidor remoto.

Tabla 36.

Prueba 5 de retorno de pesos de todos los casilleros

Ensayos	Maestro		Esclavo		Error %
	Envía	Recibe	Envía	Recibe	
1	7	111	111	7	0.0
2	7	105	111	7	2.7
3	7	111	111	7	0.0
4	7	111	111	7	0.0
5	7	111	111	7	0.0
6	7	111	111	7	0.0
7	7	111	111	7	0.0
8	7	111	111	7	0.0
9	7	111	111	7	0.0
10	7	111	111	7	0.0
C. Totales	70	1105	1110	70	0.4

El proceso de la tarea 5 almacena toda la información tanto del locker como de la variable peso en una sola trama e imprime los datos en el puerto serial. Se debe tomar en cuenta que se evaluó las tramas en función del número de caracteres que retornan al maestro, se identificó que existe un error del sistema de comunicación del 0.4% debido a que la librería software serial presenta limitaciones en la transferencia de datos.

Figura 95.

Pruebas de comunicación por tramas en maestro-esclavo

```

Maestro
1 #include <SoftwareSerial.h>
2 #include <Separador.h>
3 #include <HX711_ADC.h>
4 //HX711 constructor (CS, CLK)
5 HX711_ADC LoadCell(A3, A4);
6
7 float sf = 19.54;
8 int p = 0;
9
10 Separador s;
11
12 const int entxpin = 50;
13
14 //CHAPAS
15
16 //const int chapa = 6;
17
18 const int chapa_1 = 2;
19 const int chapa_2 = 3;
20 const int chapa_3 = 4;
21 const int chapa_4 = 5;
22
23 const int chapa_mant = 0.058;
24 const int chapa_mod = 0.001;
25
26 //sensore de efecto Hall

```

```

COM3
alvaro0000_21
trama enviada
0000_21
0000_1:0.00
0000_2:0.00
0000_3:0.00
0000_4:0.00
0000_5:0.00
0000_6:0.00
0000_7:0.00
0000_8:0.00
0000_9:0.00
0000_10:0.00
alvaro0000_21
trama enviada
0000_21
0000_1:0.00
0000_2:0.00
0000_3:0.00

```

Nota. Se evidencia que existió un error de trama en la recepción de datos por parte del maestro.

Este error no se podría visualizar en el servidor remoto debido a que no pasa la etapa del filtrado que separa la trama por secciones y compara con el formato de almacenamiento de datos, es decir cuando no se cumple la arquitectura del mensaje, el sistema no envía la información al servidor remoto.

Comprobación de hipótesis de la comunicación serial

- Hipótesis nula

Ho= La comunicación serial entre maestro y esclavo no es deficiente

- Hipótesis válida

Hi= La comunicación serial entre maestro y esclavo es deficiente

Margen de error considerada= 5%

Margen de error tabla= 0.05

Grados de libertad = 2

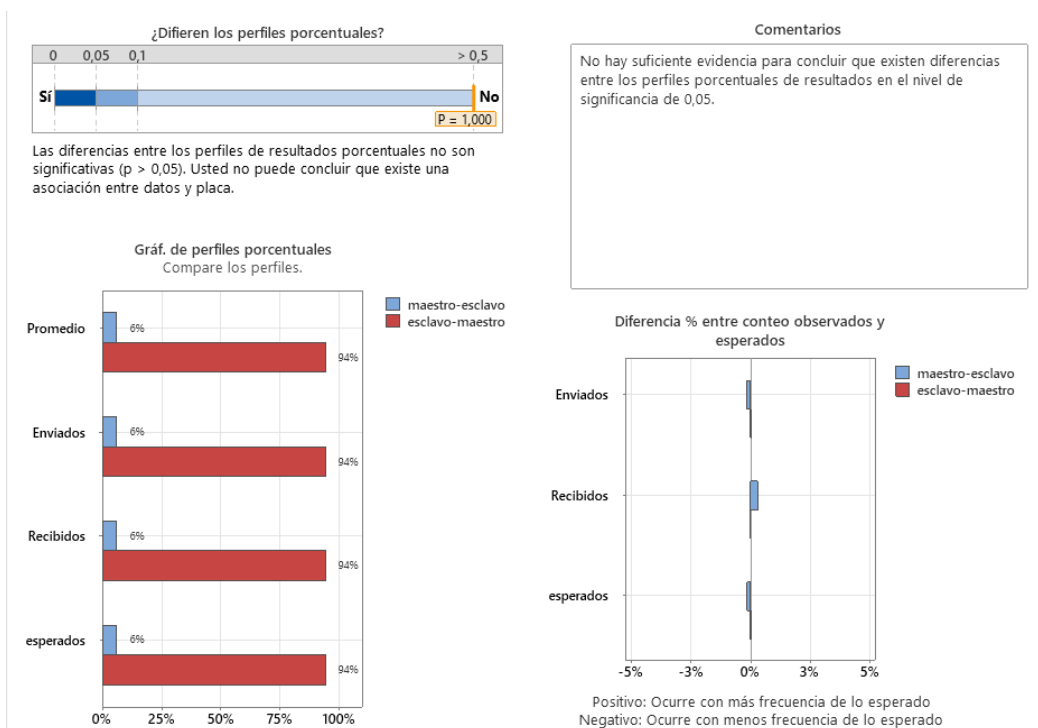
Tabla 37.

Tabla de datos para el chi-cuadrado

	Enviados	Recibidos	Esperados	Total
Maestro a esclavo	70	70	70	210
Esclavo a maestro	1115	1110	1115	3340
Total	1185	1180	1185	1185

Figura 96.

Prueba chi cuadrada para asociación: caracteres



Se espera que el maestro reciba más datos que el esclavo, por ende, la probabilidad de pérdida de información es alta.

Tabla 38.


Tabla de conteos esperados y observados para tramas

	Enviados		Recibidos		esperados	
	Obs	Exp	Obs	Exp	Obs	Exp
maestro-escl	70	70	70	70	70	70
esclavo-maes	1115	1115	1110	1110	1115	1115
Total	1185		1180		1185	

Los conteos esperados deben ser por lo menos 2 para asegurar la validez del valor p de la prueba.

Figura 97.

Tarjeta de informe para caracteres enviados y recibidos

Prueba chi-cuadrada para asociación: datos por placa Tarjeta de informe		
Verificar	Estado	Descripción
Validez de la prueba		Todas las muestras son lo suficientemente grandes para obtener suficientes conteos esperados. El valor p de la prueba debería ser exacto.

$$p = 1$$

$$p > \alpha = H_0 \text{ es válida}$$

$$p \leq \alpha = H_1 \text{ es válida}$$

$$1 > 0.05 = H_0 \text{ es válida}$$

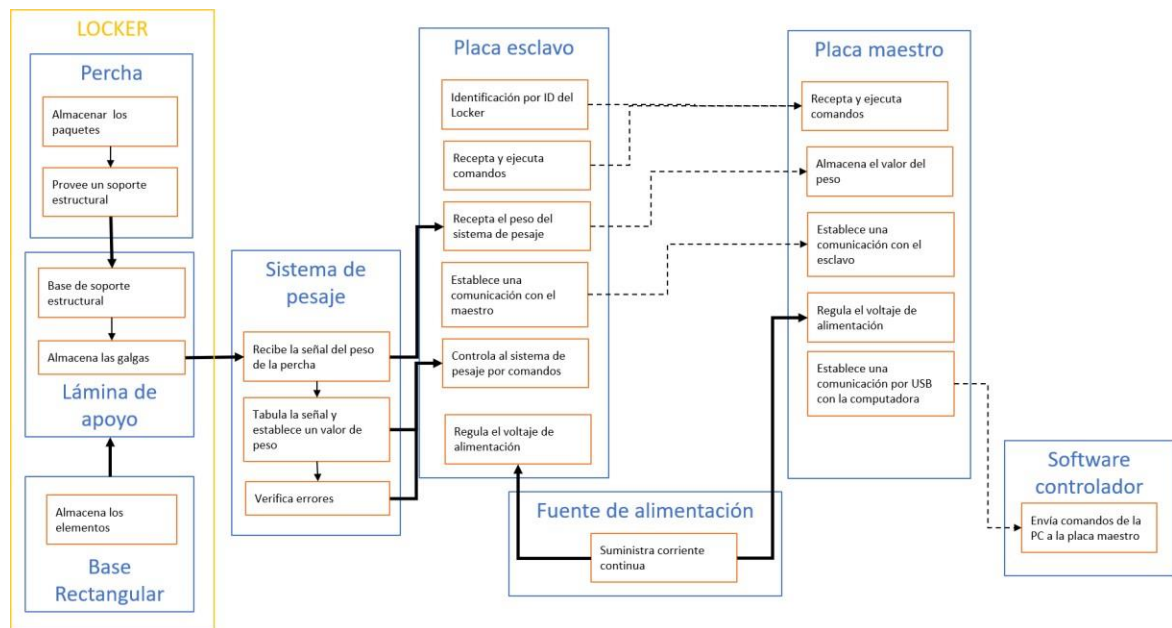
La hipótesis fue verificada por el método de distribución chi-cuadrado, con un margen de error de 5% considerando el ruido de los cables de RJ-45, la distancia de la comunicación y por la implementación de un puerto serial por software.

Pruebas del sistema de pesaje

Posterior a la prueba de conexión entre esclavo maestro, la adquisición y tabulación de datos obtenidos por las galgas es fundamental, dónde el subsistema de sensado es una función conjunta entre la parte eléctrica y mecánica que contribuye a su rendimiento general.

Figura 98.

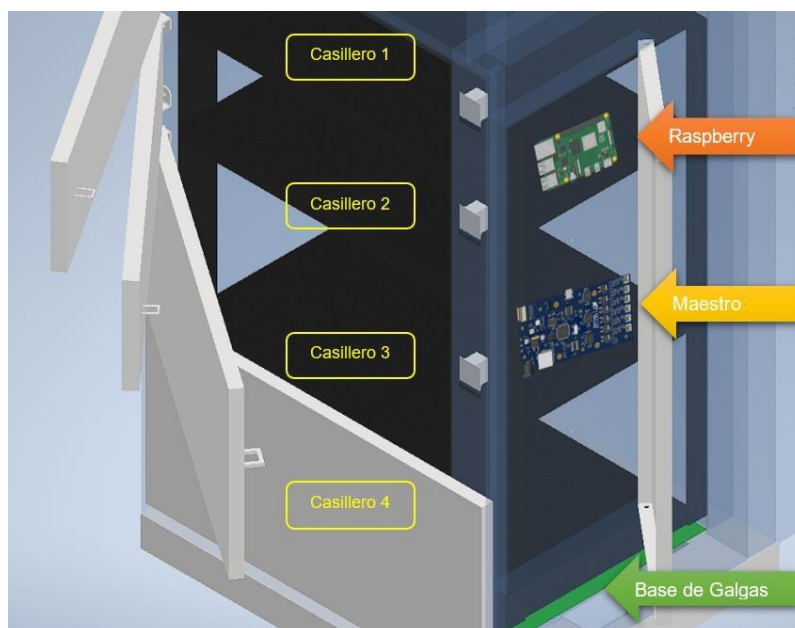
Esquema del producto y su agrupación por elementos



Es fundamental que el sistema esté sujeto a su estructura fija que soporte a toda la percha, además el sistema de pesaje se encuentra en la parte inferior de la misma y la comunicación se realiza utilizando protocolo RS485 que permite enviar los datos hacia el ordenador.

Figura 99.

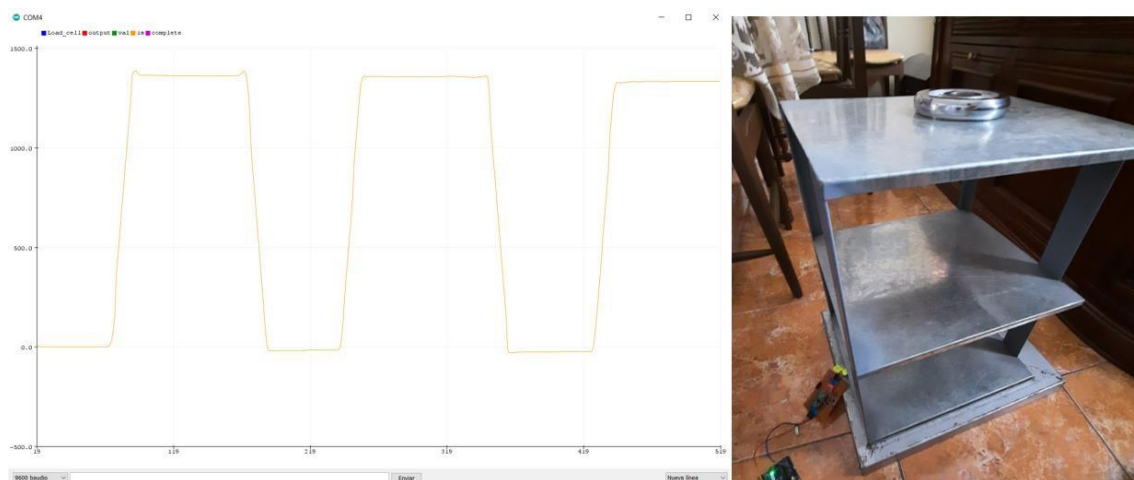
Disposición geométrica



Las pruebas inician con la obtención del factor de calibración para el driver, esto establece una relación entre un peso conocido (disco) y el peso desconocido (percha); se coloca en cero el sistema bajo normativa, después se añade un peso en cada nivel de la percha, se registra el peso cuando se vuelva estable y se lo retira para volver a medir como se muestra en la figura 100.

Figura 100.

Pruebas de funcionamiento de sensado.



Nota. Para realizar las pruebas de obtuvo un factor de calibración de 20 unidades empleando un peso conocido de 1250 g

Se tomó el quinto valor promedio de 100 datos obtenidos al sensar para cada uno de los pesos en la misma posición con la finalidad de poder saber el error porcentual de las mediciones en estado estable.

En el punto 4.6.3, “La exactitud de las mediciones para un dispositivo electrónico de tara automático debe ser de ± 0.25 ”, de la normativa NTE INEN-OIML R 76-1 (2013) Fuente, OIML R 76-1:2006. Esto es el valor absoluto de la diferencia entre los valores medidos por el sistema con el valor nominal de la pesa, sobre el valor real como:

$$e = \frac{|V_{real} - V_{aprox}|}{V_{real}}$$

Tabla 39.*Pruebas de pesaje el locker*

		CASILLERO					
Prueba		Casillero 3 (Abajo)		Casillero 2 (Medio)		Casillero 1 (Arriba)	
		Peso (g)	Error %	Peso (g)	Error %	Peso (g)	Error %
500 g	1	494	1,20%	468	6,40%	426	14,80%
	2	501	0,20%	467	6,60%	436	12,80%
	3	518	3,60%	461	7,80%	432	13,60%
	4	512	2,40%	463	7,40%	431	13,80%
	5	506	1,20%	475	5,00%	421	15,80%
	6	501	0,20%	474	5,20%	449	10,20%
	7	507	1,40%	462	7,60%	423	15,40%
	8	518	3,60%	463	7,40%	422	15,60%
	9	512	2,40%	471	5,80%	426	14,80%
	10	490	2,00%	460	8,00%	428	14,40%
1250 g	1	1220	2,40%	1054	15,7%	1061	15,12%
	2	1226	1,92%	1117	10,6%	1072	14,24%
	3	1213	2,96%	1119	10,5%	1086	13,12%
	4	1220	2,40%	1119	10,5%	1063	14,96%
	5	1212	3,04%	1151	7,9%	1051	15,92%
	6	1238	0,96%	1112	11,0%	1055	15,60%
	7	1229	1,68%	1111	11,1%	1078	13,76%
	8	1222	2,24%	1112	11,0%	1063	14,96%
	9	1219	2,48%	1136	9,1%	1045	16,40%
	10	1216	2,72%	1127	9,8%	1054	15,68%
2500 g	1	2670	6,80%	2280	8,80%	2190	12,40%
	2	2659	6,36%	2284	8,64%	2168	13,28%

CASILLERO						
Prueba	Casillero 3 (Abajo)		Casillero 2 (Medio)		Casillero 1 (Arriba)	
	Peso (g)	Error %	Peso (g)	Error %	Peso (g)	Error %
3	2656	6,24%	2277	8,92%	2174	13,04%
4	2616	4,64%	2243	10,28%	2137	14,52%
5	2700	8,00%	2260	9,60%	2106	15,76%
6	2696	7,84%	2250	10,00%	2072	17,12%
7	2666	6,64%	2310	7,60%	2129	14,84%
8	2685	7,40%	2218	11,28%	2167	13,32%
9	2613	4,52%	2243	10,28%	2106	15,76%
10	2687	7,48%	2276	8,96%	2157	13,72%
Promedio de error	3,56%		8,96%		14,49%	

Se puede observar que el casillero con menor margen de error es el de abajo debido a que en este se estableció el factor de calibración con un peso conocido, además se identifica que mientras el valor de la carga se aleje del peso con el que se estableció la tara, el error incrementa, dando un error promedio del sistema de pesaje de 9%.

Bajo las normas de diseño para el sistema de pesaje, se define el rango de los errores máximos permitidos para un instrumento de clase III, sin embargo, hay que considerar que el sistema necesita estar en estado estable con una exactitud de 0.25% entre mediciones.

Tabla 40.*Errores máximos permitidos*

Errores máximos permitidos en la verificación inicial	Para cargas m expresadas en divisiones de verificación e			
	Clase I	Clase II	Clase III	Clase IIII
$\pm 0,5 e$	$0 \leq m \leq 50\ 000$	$0 < m \leq 5000$	$0 < m \leq 500$	$0 < m \leq 50$
$\pm 1 e$	$50\ 000 < m \leq 200\ 000$	$5000 < m \leq 20000$	$500 < m \leq 2000$	$50 < m \leq 200$
$\pm 1,5 e$	$200\ 000 < m$	$20000 < m \leq 100000$	$2000 < m \leq 10000$	$200 < m \leq 1000$

Según la norma NTE INEN-OIML R 76-1 (2013) Fuente, OIML R 76-1:2006, en el punto 3.5.2 se define que “Los errores máximos permitidos en servicio deben ser iguales al doble de los errores máximos permitidos en la verificación inicial”. Por lo tanto, el sistema de forma general puede establecer un error máximo permitido en servicio entre 1.5%, 2.5% y 3.5%.

Para poder establecer el error máximo permitido se debe tomar en cuenta las siguientes consideraciones:

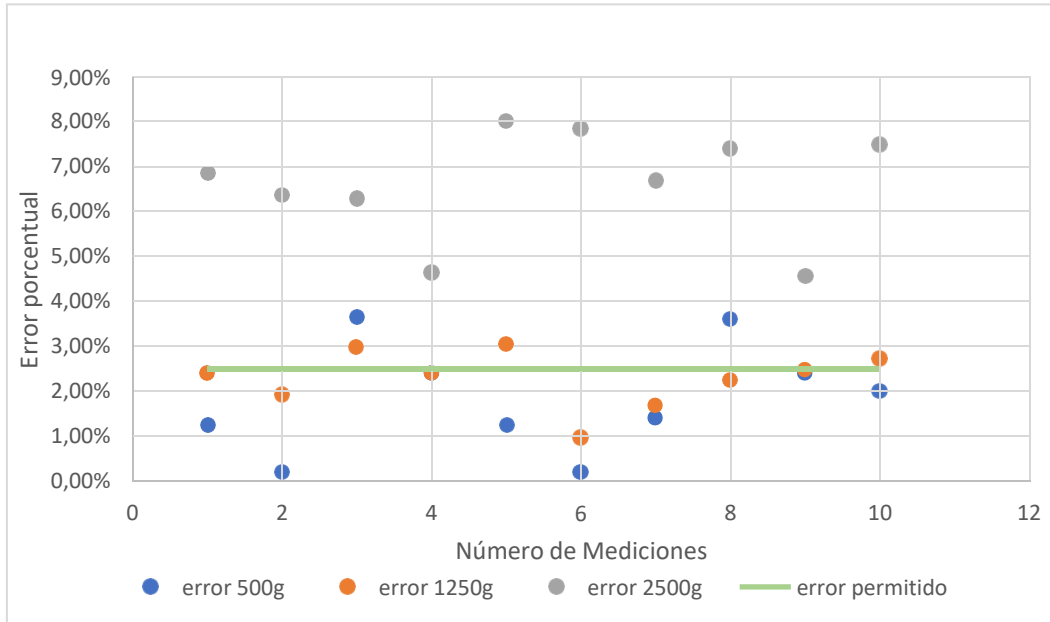
- El proceso de tarificado se establece en rangos por kilogramos, se define el rango en 1 Kg a 4 Kg.
- Para el diseño estructural de la percha se establece que la carga por piso es de 2.5 Kg para el análisis, sin embargo, la estructura podría soportar hasta 4Kg.

En función de las condiciones planteadas se establece que el valor de error máximo aceptado es de 2.5% para cada casillero.

Para poder visualizar de mejor manera los errores porcentuales generados de acuerdo a cada casillero, se emplea un gráfico de dispersión con el rango de error máximo aceptado.

Figura 101.

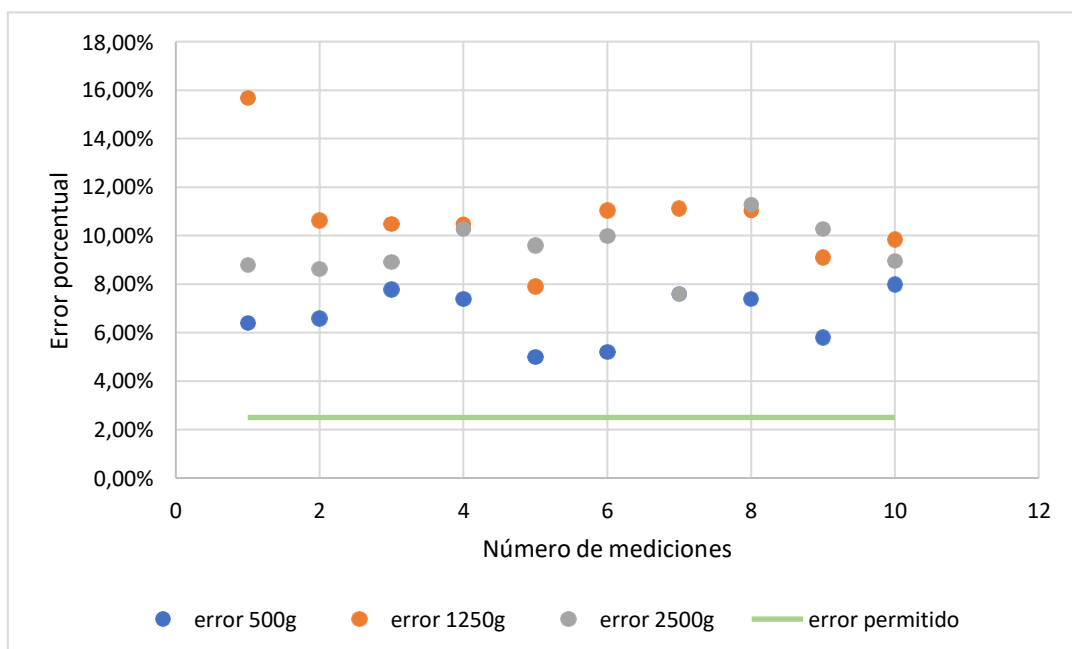
Error locker 3 (Abajo)



Nota. De acuerdo al gráfico se puede visualizar que el error obtenido para 500g y 1250g en gran parte se encuentran dentro del límite establecido, mientras que para el de 2500g ningún valor es aceptado.

Figura 102.

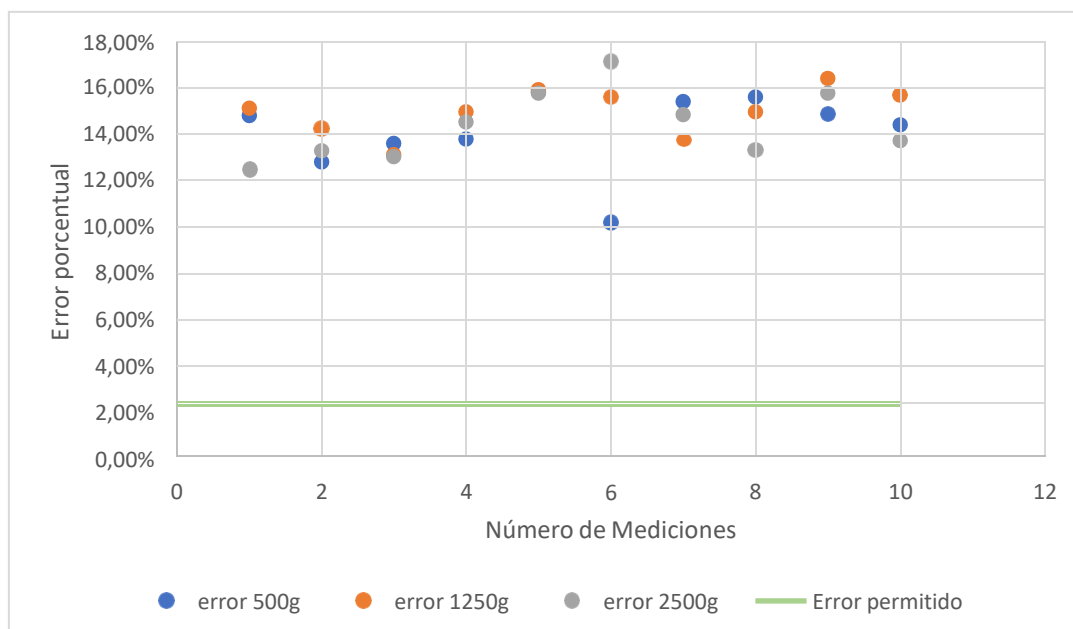
Error locker 2 (Medio)



Nota. De acuerdo al gráfico se puede visualizar que el error obtenido para 500g, 1250g y 2500g se encuentran fuera del límite establecido, por lo tanto, ninguno se acepta.

Figura 103.

Error locker 1 (Arriba)



Nota. De acuerdo al gráfico se puede visualizar que el error obtenido para 500g, 1250g y 2500g se encuentran fuera del límite establecido, por lo tanto, ninguno se acepta.

Se observa que el error generado para los datos en cada uno de los casilleros toma valores similares entre sí, sin tomar en cuenta el peso medido; por lo que se puede aplicar una corrección de valores en función del piso que se ubique con la finalidad de que los pesos sean aceptados.

Tabla 41.

Pruebas de pesaje el locker con el factor de corrección

Prueba	Casillero 3 (Abajo)				Casillero 2 (Medio)				Casillero 1 (Arriba)			
	Peso (g)	Error %	<2,5%		Peso (g)	Error %	<2,5%		Peso (g)	Error %	<2,5%	
500	1	494	1,20%	+	511	2,26%	+		498	0,32%	+	
g	2	501	0,20%	+	510	2,04%	+		510	2,02%	+	

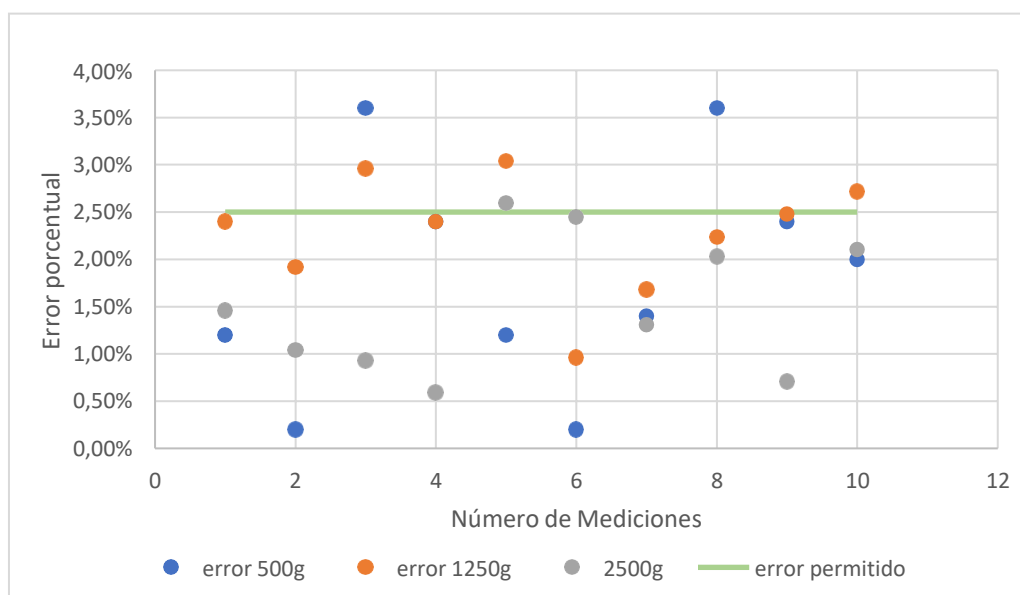
Prueba	Casillero 3 (Abajo)			Casillero 2 (Medio)			Casillero 1 (Arriba)			
	Peso	Error	<2,5%	Peso	Error	<2,5%	Peso	Error	<2,5%	
	(g)	%		(g)	%		(g)	%		
	3	518	3,60%	-	503	0,73%	+	505	1,09%	+
	4	512	2,40%	+	505	1,17%	+	504	0,85%	+
	5	506	1,20%	+	518	3,79%	-	492	1,49%	+
	6	501	0,20%	+	518	3,57%	-	525	5,07%	-
	7	507	1,40%	+	505	0,95%	+	495	1,02%	+
	8	518	3,60%	-	506	1,17%	+	494	1,25%	+
	9	512	2,40%	+	515	2,91%	-	498	0,32%	+
	10	490	2,00%	+	503	8,00%	-	501	0,15%	+
	1	1220	2,40%	+	1152	7,88%	-	1241	0,69%	+
	2	1226	1,92%	+	1220	2,37%	+	1254	0,34%	+
	3	1213	2,96%	-	1223	2,20%	+	1271	1,65%	+
	4	1220	2,40%	+	1223	2,20%	+	1244	0,50%	+
1250	5	1212	3,04%	-	1257	0,60%	+	1230	1,63%	+
g	6	1238	0,96%	+	1215	2,81%	-	1234	1,25%	+
	7	1229	1,68%	+	1214	2,90%	-	1261	0,90%	+
	8	1222	2,24%	+	1215	2,81%	-	1244	0,50%	+
	9	1219	2,48%	+	1241	0,71%	+	1223	2,19%	+
	10	1216	2,72%	-	1231	1,50%	+	1233	1,35%	+
	1	2536	1,46%	+	2491	0,36%	+	2562	2,49%	+
	2	2526	1,04%	+	2495	0,19%	+	2537	1,46%	+
2500	3	2523	0,93%	+	2488	0,50%	+	2544	1,74%	+
g	4	2485	0,59%	+	2450	1,98%	+	2500	0,01%	+
	5	2565	2,60%	-	2469	1,24%	+	2464	1,44%	+
	6	2561	2,45%	+	2458	1,68%	+	2424	3,03%	-

Prueba	Casillero 3 (Abajo)			Casillero 2 (Medio)			Casillero 1 (Arriba)		
	Peso (g)	Error %	<2,5%	Peso (g)	Error %	<2,5%	Peso (g)	Error %	<2,5%
7	2532	1,31%	+	2524	0,95%	+	2491	0,36%	+
8	2550	2,03%	+	2423	3,07%	-	2535	1,42%	+
9	2485	0,71%	+	2450	1,98%	+	2464	1,44%	+
10	2552	2,11%	+	2487	0,54%	+	2524	0,95%	+
P. Positivos		24			21			28	
P. Negativos		6			9			2	
Porcentaje		80%			70%			93,2%	
Promedio de error		1,87%			2,17%			1,30%	

En el locker 3 (Abajo) debido a que el error se genera solo para el de 2500g y no para el resto de valores, no se puede aplicar el factor de corrección de todos los datos, sin embargo, si se aplica un factor de corrección de -5% del valor obtenido, se puede conseguir desplazar los valores a un valor aceptable.

Figura 104.

Error corregido locker 3 (Abajo)

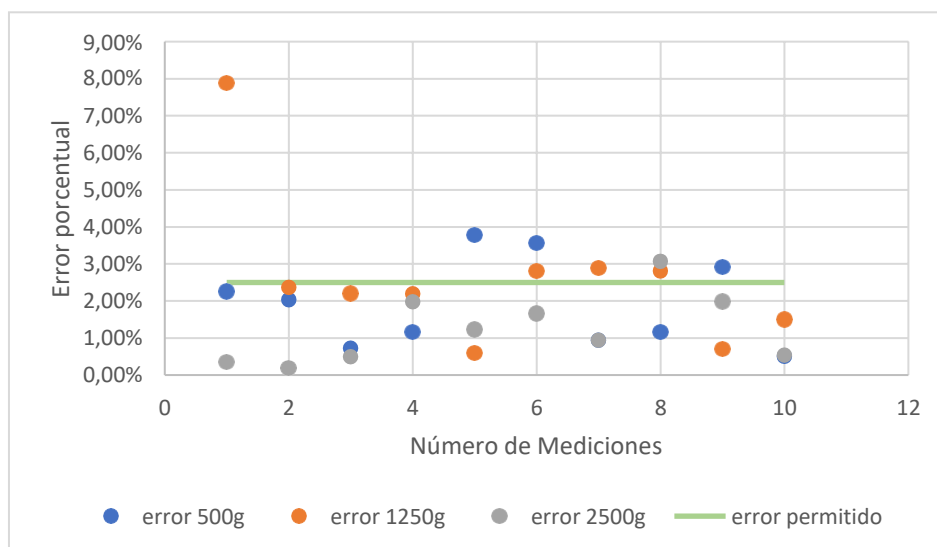


Nota. De acuerdo al gráfico se puede visualizar que el error obtenido para 500g, 1250g y 2500g.

En el locker 2 (Medio) debido a que el error se genera para todos los valores, se puede aplicar un factor de corrección al valor obtenido de 9.25% para conseguir que los valores se desplacen e ingresen en el rango permitido

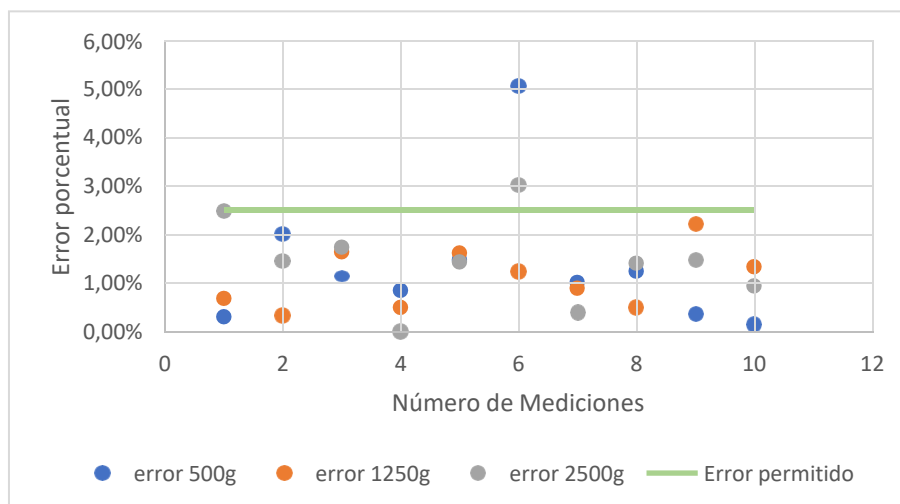
Figura 105.

Error corregido locker 2 (Medio)



Nota. De acuerdo al gráfico se puede visualizar que el error obtenido para 500g, 1250g y 2500g se encuentran dentro del límite establecido, por lo tanto, se acepta.

En el locker 1 (arriba) debido a que el error se genera para todos los valores, se puede aplicar un factor de corrección al valor obtenido de 17% para conseguir que los datos se desplacen e ingresen en el rango permitido.

Figura 106.*Error corregido locker 1 (Arriba)*

Nota. De acuerdo al gráfico se puede visualizar que el error obtenido para 500g, 1250g y 2500g se encuentran dentro del límite establecido, por lo tanto, se acepta.

Debido a que la obtención de tarificado se va a trabajar por rangos, no se necesita que el valor pesado sea exacto, sin embargo, internamente en la programación de la placa se establece el factor de corrección dependiendo el casillero que se va a emplear.

Comprobación de hipótesis de la medición del sistema de pesaje

- Hipótesis nula

H_0 = La exactitud del sistema de pesaje no depende del casillero

- Hipótesis valida

H_i =La exactitud del sistema de pesaje depende del casillero

Margen de error considerada= 5%

Margen de error tabla= 0.05

Grados de libertad = 2

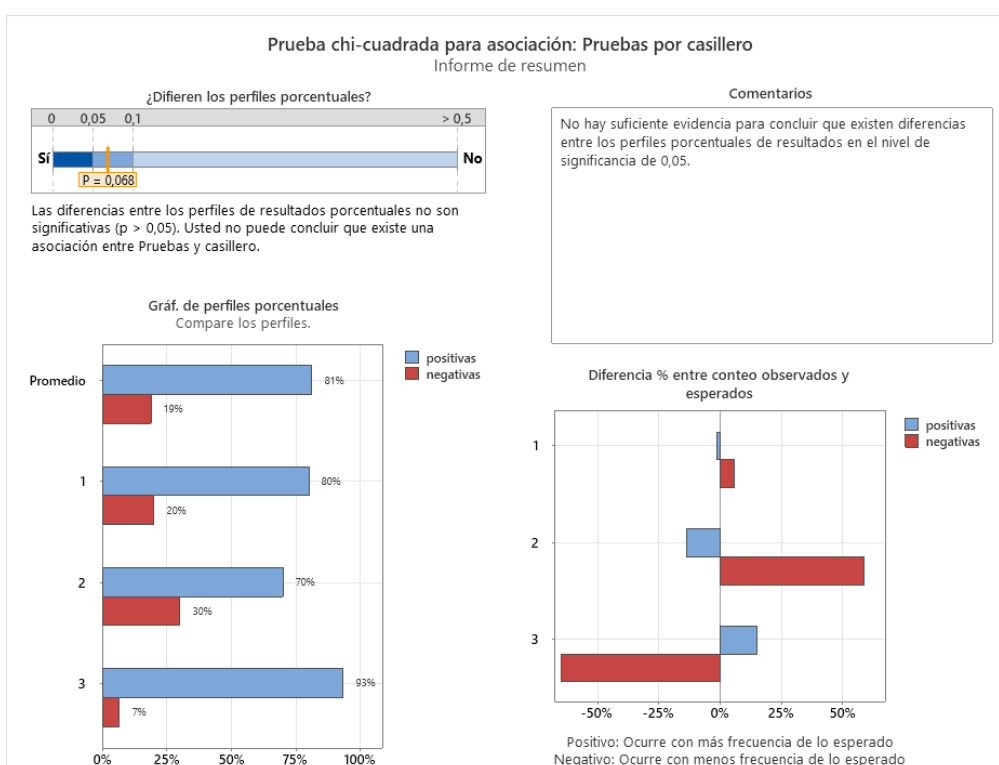
Tabla 42.

Tabla de datos para el chi-cuadrado

Locker	Pruebas positivas	Pruebas negativas	Total
Casillero 1	24	6	30
Casillero 2	21	9	30
Casillero 3	28	2	30
Total	73	17	90

Figura 107.

Prueba chi-cuadrada por asociación: Pruebas por casillero



De acuerdo a la gráfica obtenida entre el conteo esperado y observado, se espera que en el casillero intermedio tienda a experimentar con mayor frecuencia pruebas negativas, por el contrario, en el casillero 3, es muy poco probable que el sistema falle.

Tabla 43.


Tabla de conteos esperados y observados para pruebas por casillero

Conteos observados y esperados						
	1		2		3	
	Obs	Exp	Obs	Exp	Obs	Exp
positivas	24	24	21	24	28	24
negativas	6	5,7	9	5,7	2	5,7
Total	30		30		30	

Los conteos esperados deben ser por lo menos 2 para asegurar la validez del valor p de la prueba.

Figura 108.

Tarjeta de informe para pruebas de los casilleros

Prueba chi-cuadrada para asociación: Pruebas por casillero		
Tarjeta de informe		
Verificar	Estado	Descripción
Validez de la prueba		Todas las muestras son lo suficientemente grandes para obtener suficientes conteos esperados. El valor p de la prueba debería ser exacto.

$$p = 0.068$$

$$p > \alpha = H_0 \text{ es válida}$$

$$p \leq \alpha = H_1 \text{ es válida}$$

$$0.068 > 0.05 = H_0 \text{ es válida}$$

La hipótesis fue verificada por el método de distribución chi-cuadrado, con un margen de error de 5% considerando el ruido que se produce en el cableado de las galgas y considerando la variación del peso según la posición en la que se coloque el paquete sobre la percha.

Implementación de RPA en el proceso de envío de encomiendas

Figura 109.

Proceso de envío de paquetes con la adición de RPA.



Nota. El color verde representa las interacciones del cliente, el color azul corresponde al personal dedicado a comprobar el estado del paquete, el color morado representa las actividades del servidor remoto y el color naranja muestra las funciones del robot CSC.

Las pruebas se realizaron usando el sistema integrado de la parte mecánica, eléctrica y software. El sistema secuencial necesita que el usuario elija si va a enviar o a retirar el paquete y posteriormente decidir si va a usar el código de validación por escáner QR o por ingreso de teclado. Una vez se autoriza y válida el envío o recepción de paquete, se inicia la secuencia en las placas maestro-esclavo con los datos de cliente y del casillero, se actualiza la base de datos en el servidor remoto para cambiar los estados del casillero y generar el cobro con el peso sensado.

Figura 110.

Pruebas en el ensamblaje total del locker principal



El proceso puede dividirse en tres fases, la primera que se refiere a la manipulación del HMI y selección del tipo de servicio, la segunda donde el cliente deposita el paquete y finalmente la tercera etapa donde se envían los datos al servidor remoto para realizar el cobro usando la pasarela de pagos.

Tabla 44.

Datos del tiempo de uso de sistema de pesaje

Prueba	Sujeto	Edad	Fases			Total
			Fase 1	Fase 2	Fase 3	
1	1	21	15	17	27	59
	2	23	14	15	25	54
	3	24	16	25	31	72
	4	29	15	19	31	65
	5	32	14	20	30	64
	6	42	17	15	35	67
	7	46	18	22	32	72

Prueba	Sujeto	Edad	Fases			Total
			Fase 1	Fase 2	Fase 3	
2	1	21	13	17	23	53
	2	23	10	14	26	50
	3	24	12	21	27	60
	4	29	8	19	28	55
	5	32	11	15	25	51
	6	42	10	15	27	52
	7	46	16	20	29	65
3	1	21	13	16	20	49
	2	23	10	15	25	50
	3	24	11	22	23	56
	4	29	12	19	24	55
	5	32	15	18	26	59
	6	42	12	16	24	52
	7	46	18	19	21	58
C. Totales	14	31,	13,3	18	26,6	58

Nota. Las personas realizaron dos pruebas para el envío de paquetes y a los tiempos menores a 60 segundos se los cataloga como positivos.

Las pruebas realizadas se dieron en tres partes, en la primera es notorio que el usuario necesita bastante tiempo para explorar el HMI por lo cual solo uno pasa la prueba, sin embargo, en el segundo y tercer intento las pruebas resultaron favorecedoras pues se tiene 13 pruebas positivas. Con estos valores se puede proyectar una disminución del tiempo de uso del servicio.

Se obtuvo la siguiente tabla de funcionamiento del sistema para las dos etapas de estudio en donde se considera la influencia del tiempo de uso del servicio, la seguridad del paquete y la exactitud del sensado del peso.

Tabla 45.*Datos recolectados del sistema en cada intento de envío del paquete*

Prueba	Sujeto	Tiempos	Seguridad	Sensado
1	1	+	+	-
	2	+	+	+
	3	-	+	+
	4	-	+	-
	5	-	+	+
	6	-	+	+
	7	-	+	+
2	1	+	+	+
	2	+	+	+
	3	+	+	+
	4	+	+	-
	5	+	+	+
	6	+	+	+
	7	-	+	+
3	1	+	+	+
	2	+	+	+
	3	+	+	+
	4	+	+	+
	5	+	+	+
	6	+	+	+
	7	+	+	+
Pruebas positivas		15	21	18
Pruebas negativas		6	0	3
Porcentaje (%)		71,42	100	85,71

Nota. La seguridad y el Sensado se evalúa bajo los mismos criterios de evaluación de pruebas anteriores.

Validación de la hipótesis del sistema

Mediante las pruebas realizadas, correspondiente a los valores medidos con el sistema de pesaje con diferentes muestras, se procede a calcular el chi-cuadrado.

- Hipótesis nula

Ho= El diseño y construcción de lockers automáticos para la recepción de paquetes utilizando RPA (robotic process automation) no permitirá agilizar el proceso de envío de encomiendas a WOLF

- Hipótesis válida

Hi= El diseño y construcción de lockers automáticos para la recepción de paquetes utilizando RPA (robotic process automation) permitirá agilizar el proceso de envío de encomiendas a WOLF.

Margen de error considerada= 5%

Margen de error tabla= 0.05

Grados de libertad = 2

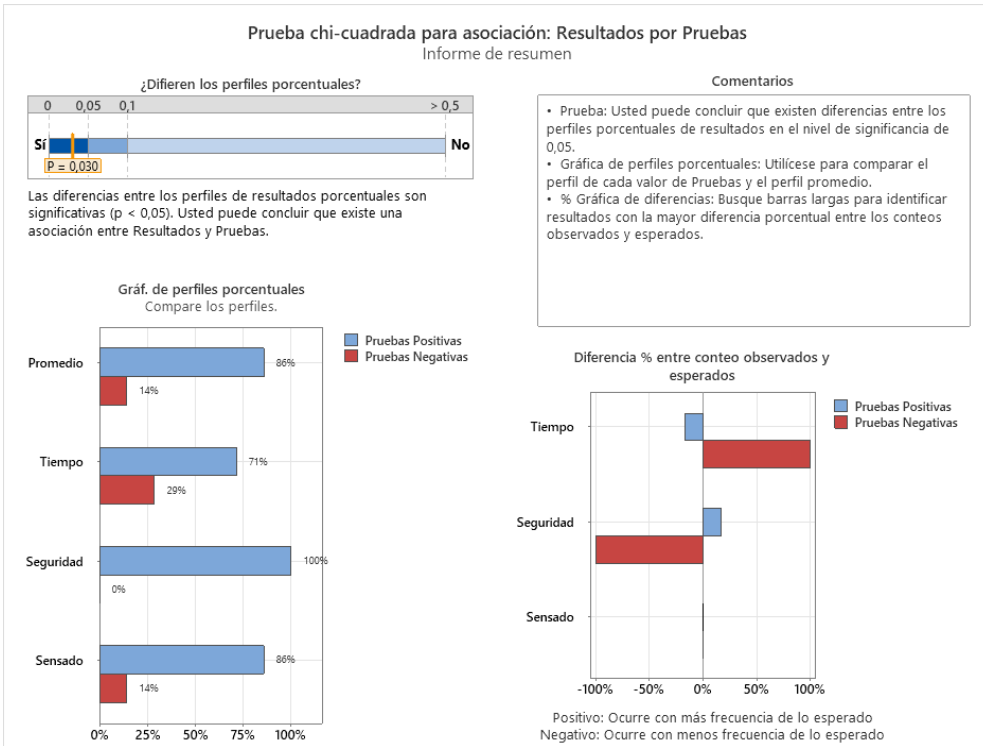
Tabla 46.

Tabla de datos para el chi-cuadrado

Locker	Tiempo	Seguridad	Sensado	Total
Pruebas positivas	15	21	18	54
Pruebas negativas	6	0	3	9
Total	21	21	21	63

Figura 111.

Prueba de chi-cuadrada para asociación: Funcionalidad



El sistema en promedio tiene mayores pruebas positivas que negativas, en gran parte se debe a la fiabilidad de la seguridad mecánica y la precisión del sensado, sin embargo, es preciso recalcar que el tiempo que el usuario se demora en manipular el HMI tiene una tendencia a reducirse mientras más interacciones tenga con del sistema.

Tabla 47.

Informa de diagnóstico para asociación: tiempo, seguridad y sensado


Conteos observados y esperados

	Tiempo		Seguridad		Sensado	
	Obs	Exp	Obs	Exp	Obs	Exp
Pruebas Posi	15	18	21	18	18	18
Pruebas Nega	6	3	0	3	3	3
Total	21		21		21	

Los conteos esperados deben ser por lo menos 2 para asegurar la validez del valor p de la prueba.

Figura 112.

Tarjeta de informe para asociación: Funcionalidad

Prueba chi-cuadrada para asociación: Resultados por Pruebas		
Tarjeta de informe		
Verificar	Estado	Descripción
Validez de la prueba		Todas las muestras son lo suficientemente grandes para obtener suficientes conteos esperados. El valor p de la prueba debería ser exacto.

$$p = 0.03$$

$$p > \alpha = H_0 \text{ es válida}$$

$$p \leq \alpha = H_1 \text{ es válida}$$

$$0.03 < 0.05 = H_1 \text{ es válida}$$

Posterior al análisis de los datos de relación entre el tiempo, seguridad y sensado, se acepta la hipótesis válida (H_1), por lo tanto, este proyecto agiliza el proceso de envío de encomiendas ofreciendo fiabilidad en el sensado con variaciones mínimas de error dando como resultado un sistema de pesaje confiable, además proporciona seguridad a los paquetes pues tiene bajas probabilidades de fallo a niveles electrónico y mecánico.

Capítulo VI

Conclusiones y recomendaciones

Conclusiones

La aplicación de un robot CSC en el sistema de paquetería con RPA, es capaz de reducir tiempo de servicio y asegurar su correcto funcionamiento, en este sentido se puede identificar los puntos clave que permiten optimizar al sistema, siendo la toma de datos del paquete y el cobro por el servicio. La automatización de estos puntos significa la reducción del coste para el cliente en comparación al sistema tradicional, además que brinda seguridad ante errores y tiene una escalabilidad gracias a herramientas externas, especialmente en la pasarela de pagos.

En el prototipo final se emplearon cerraduras pequeñas NC ya que, en caso de existir un corte de luz, los casilleros quedarían bloqueados y de esta manera se ofrecería seguridad a los paquetes depositados; de igual manera, al emplear este tipo de cerraduras se reduce costos de producción porque el diseño de las puertas emplea menos material y el sistema ya no requeriría el uso de una celda de energía de respaldo (UPS).

La torre se considera robusta a nivel mecánico bajo condiciones exigentes, debido a que las cerraduras, pese a que lleguen a un punto de temperatura alta, siguen operando normalmente de acuerdo a la prueba realizada a lo largo de 3 horas. De igual manera, al momento de cerrarse las puertas, el espacio entre ellas no permite el ingreso de objetos con la finalidad de forzar la apertura de los casilleros y vulnerar la seguridad del paquete depositado.

En base a la investigación bibliográfica realizada, se desarrolló el sistema mecánico y electrónico para poder sensor los paquetes depositados, resultando óptimo siempre y cuando se tomen en cuenta las siguientes consideraciones: La percha interna debe en lo posible evitar el roce con las paredes para disminuir el error al momento de tomar el peso; para poder reducir el error se aplicó un promedio de 100 valores y un factor de corrección dependiendo del casillero en el que se va a depositar

el paquete; el factor de calibración es fijo y el sistema encera la medición antes de depositar el paquete dentro de un casillero.

Se diseñaron dos placas PCB (esclavo-maestro) para controlar los lockers implementando un protocolo de comunicación RS 485 con la finalidad de poder conectar de manera fácil y rápida los esclavos con ID individual secuencialmente al sistema en función de la capacidad requerida por la empresa. Las placas finales se construyeron empleando el microcontrolador Atmega 2560 debido a que posee más pines analógicos y digitales.

Se aplicó la normativa EMC para distribuir por bloques los elementos electrónicos al interior de la placa y unirlos de manera correcta, creando pistas a doble capa mediante vías para la parte de control y para la parte de potencia, basado en cálculos para el ancho de las mismas. Se emplearon componentes SMD para el prototipo final debido a que emplean menor espacio y cumplen la misma función; las conexiones de cerraduras y de sensores mediante Molex permiten que en caso de avería se puedan sustituir de manera fácil y rápida sin la necesidad de soldar volviendo modular a la placa por su arquitectura interna.

Mediante el componente MAX 485 y la librería de software serial se desarrolló la conexión tipo bus entre las placas maestro/esclavo. Se establecieron comandos para administradores y usuarios que la placa maestra gestionará y enviará al esclavo para que responda las peticiones; con la implementación de una rutina para el maestro se comprobó la modularidad del sistema, ya que detecta cuantas placas esclavas se encuentran conectadas a él e incluso imprime una trama con el ID de todas las que reconozca.

La comunicación del puerto serial entre Raspberry y maestro es síncrona, es decir que no permite una comunicación simultánea, pues comparten el mismo buffer, estas características se aprovechan para establecer el sistema secuencial de comando-respuesta. Esta comunicación se ejecuta a cortas distancias, por lo que es necesario colocar en una misma torre a la placa maestra con la Raspberry.

Cada placa dispondrá de un regulador de 5v para que pueda operar el microcontrolador, pero en caso de este fallar, la placa con el regulador averiado obtendrá el voltaje a través del cable de par trenzado conectado mediante los puertos RJ-45 permitiendo que la unión de más esclavos sea fácil y rápida por su modelo “Plug and Play”.

El sistema de sensado de peso, es independiente para cada torre y se encuentra susceptible a dar información errónea por efecto de vibraciones de la percha al momento de receptar el paquete, de igual manera mientras más alejado del centro de la placa se deposite el paquete, aumenta la probabilidad de errores en el sensado debido a la deformación no homogénea de la estructura.

Se mejora significativamente la programación del Atmega 2560 al momento de implementar RTOS (Real Time Operating System), porque pese a que se implementó la programación por bucle infinito y funcionó correctamente, se separó por diferentes tareas la programación, mejorando el rendimiento de las placas y ofreciendo un sistema embebido.

La Raspberry Pi 4 se empleó como computador central para la administración de periféricos, uso de servicios de Backend-Frontend, conexión al servidor remoto y comunicación maestro/esclavo gracias a la potencia de memoria RAM, sin embargo, este presenta limitaciones para una placa que integre reguladores de voltaje, controladores y elementos de potencia por lo que se decidió tener una placa dedicada (maestro) que sirva como administrador de la trama de datos de sus esclavos, tenga el control del hardware y sea el único enlace de comunicación con el computador central o Raspberry.

NestJS como servidor Backend permitió construir una arquitectura flexible, ya que admite el uso de varias librerías externas como “Rpi-Keypad” para el teclado, proporciona la capacidad de hacer pruebas y hacer cambios en caliente con el modo de desarrollador. En este servidor local se definió la funcionalidad del robot CSC, es

decir, que el servidor remoto a manera de controlador asigna y supervisa las actividades del robot.

Se utilizó MQTT para monitorear al sistema de lockers en tiempo real, pues se aprovecha que la velocidad de envío de datos es rápida y como su comunicación es unidireccional, solo se dedica a actualizar los estados del casillero, lo que es idóneo para el modo usuario.

El sistema se protege ante desconexiones gracias a la arquitectura de MQTT, pues el nivel de seguridad permite enviar el último mensaje al servidor remoto y guardarlo en el puerto del broker hasta que se vuelva a conectar el sistema, además MQTT puede identificar el estado de conexión de un locker y en caso de una desconexión no programada, el servidor remoto puede identificar un fallo en el sistema eléctrico y alertar.

Se empleó la comunicación HTTP para métodos de verificación de usuarios, permisos de autorización de uso del servicio, aceptar el tarifado y registros de estados, siendo estas características fundamentales para el modo administrador por la necesidad de confirmaciones de eventos del servidor remoto y del locker.

Se estableció que el nivel de seguridad es confiable, pues la Raspberry se inicializa con el servidor remoto con peticiones HTTP, permitiendo conectar al locker mediante un usuario y contraseña que hace referencia al nombre del locker, su ubicación, los casilleros que tiene disponibles. Además, los datos del cliente necesariamente deben pasar por el proceso de verificación de código de autorización para hacer uso del servicio, además los protocolos de comunicación usan un token de validación para enviar el mensaje y garantizar que este llegue al servidor remoto. Sin embargo, si se tiene acceso al software de la Raspberry también se tiene acceso al token y se vuelve vulnerable a ataques.

Se protegió la conectividad del servidor local (Localhost) mediante la implementación de una clave de seguridad usando la librería "React Router Dom" por

lo tanto, solo la Raspberry tiene acceso a los URL y con ello se protege los canales de datos entre Frontend-Backend y Backend-Frontend.

Al implementar el código QR se agiliza el proceso de validación de códigos y por ende el proceso de recepción de paquetes, sin embargo, se puede emplear el teclado para ingresar el código y habilitar la navegación en las páginas del HMI de React.

La implementación de React para una interfaz de usuario resulto satisfactorio a nivel de desarrollo, pues tiene muchas herramientas disponibles que permiten el control de datos, visualizaciones de eventos y logra generar una navegación intuitiva entre ventanas con CSC para mejorar la organización y disposición de elementos en las páginas que se presentan en pantalla.

El tiempo de uso del servicio por parte del cliente se ve reducido significativamente mientras se va familiarizando con el HMI y con el proceso, esto se obtiene al interactuar repetidamente con el sistema.

Recomendaciones

Es recomendable familiarizarse con el framework de TypeScript para aprovechar las ventajas que brinda NestJS, pues no hacerlo puede llevar a una mal interpretación de la lógica de funcionamiento y manejar de forma errónea los parámetros de las variables, pues debe responder a estructuras concretas por lo que es susceptible a añadir o quitar propiedades a las funciones.

A nivel de desarrollo se recomienda emplear GitHub para realizar respaldos continuos de los archivos y generar un control de versiones, tiene la potencia de trabajar en equipo y llevar un registro de cambios en el proyecto.

Es necesario evaluar el sistema antes de aplicar MQTT, pues este tiene diferentes herramientas para configurar su funcionamiento, desde niveles de seguridad hasta la arquitectura de ancho de banda del mensaje. Es primordial tener un conocimiento básico sobre el protocolo MQTT porque puede ser un poco confuso entender conceptos y funcionamientos de broker, Topic, suscriptor y publicador.

Para la verificación de códigos de autorización con HTTP se recomienda almacenar la contraseña de forma cifrada en una base de datos, con ello al realizar consultas o peticiones al servidor remoto, se espera una respuesta sin el uso de un comodín.

Al momento de implementar la comunicación serial se recomienda utilizar una arquitectura jerárquica en el envío de la trama para que no exista una colisión de datos, respetando el orden de pregunta (maestro) – respuesta (esclavo) y mejorando el flujo de datos.

Al momento de diseñar las placas, se recomienda implementar pads para la medición de voltajes en diferentes puntos en la capa superior con la finalidad de facilitar la verificación de conexiones, puesto que, las placas tienen un recubrimiento en toda la superficie de la capa que aísla las pistas del exterior.

Si la producción de las placas va a ser masiva, se podría elaborar un solo modelo que funcione como maestro o esclavo; de esta manera se abarataría costos de

producción para PCB y simplemente se soldarían componentes de acuerdo a qué función va a desempeñar (maestro o esclavo).

Al implementar un microcontrolador de Atmega, pese a que en el datasheet permite un rango de 18 pF a 30 pF para los capacitores en el cristal, se recomienda emplear los de 22 pF para un correcto funcionamiento al diseñar una PCB; de igual manera al momento de mallar la placa, es recomendable dejar una amplia área de cobre o colocar pequeños orificios debajo del mismo, tanto para el integrado, así como para el regulador, con la finalidad de disipar el calor generado en su operación.

En diseños posteriores se puede implementar una rutina para el led RGB que muestre estados de la placa mediante indicadores lumínicos, con la finalidad de que las personas que den mantenimiento a los lockers sepan puntualmente que fallas están ocurriendo en la torre al solo ver el color del led.

Resulta muy útil cargar programas en la placa por medio de ICSP; sin embargo, para una producción masiva, la actualización del programa en cada placa se volvería demasiado complicada. Por lo que se propondría buscar un medio de actualización remota en la cual la Raspberry (servidor local) descargue las actualizaciones y ordene al maestro quemar el programa por el puerto serial a todas las placas esclavas.

Cuando se carga el archivo “.hex” por ICSP usando el programa “Progisp.exe”, se debe tener cuidado de no mover los cables de datos durante el proceso, debido a que la desconexión puede bloquear el microcontrolador y dejarlo inservible para la lectura y carga del programa. Si se da el caso de bloqueo es preferible cambiar de microcontrolador.

Al momento de generar las placas en SMD, la herramienta de Easyeda permite elegir, cambiar o reemplazar elementos eléctricos de diferentes marcas, por lo cual se recomienda revisar el datasheet de los componentes que se encuentran en la placa, tomando en cuenta que el encapsulado y las características sean las mismas para las que se realizó el diseño.

Para mejorar el subsistema de pesaje, en posteriores diseños se podría emplear un material más liviano para la percha, con la finalidad de generar menor error de sensado para los casilleros.

Para la puerta de mantenimiento se recomienda utilizar 2 seguros; una cerradura pilotada por comandos desde la placa NA y una cerradura con llave a la cual solo tendría acceso personal de mantenimiento, con la finalidad de que puedan acceder sin comando incluso en un corte de luz.

Los periféricos conectados a la Raspberry pueden ser solo de tipo HID para que el sistema los reconozca sin la necesidad de instalar drivers o librerías externas.

El programa final necesita un espacio de memoria RAM superior a los 3GB por lo que se recomienda usar una placa de control que soporte esta magnitud de carga y procesamiento de datos. Además, se recomienda utilizar disipadores de calor o un ventilador para proteger al procesador y el controlador USB de la Raspberry.

Se recomienda mejorar el enlace entre la secuencia de ingreso de paquete y el HMI, esto se puede lograr con la utilización de la librería Socket.io para obtener la lectura de los valores de estado del locker y sincronizarlos con la navegación del HMI.

Bibliografía

- Agencia de Regulación y control Postal. (2018). *Informe estadístico Anual del sector postal Ecuatoriano. Estudios de Ecuador*. <https://regulacion.mintel.gob.ec/wp-content/uploads/2019/07/BOLETIN-ESTADISTICO-ANUAL-2018-v3.pdf>
- Aguirre, Santiago & Rodriguez, Alejandro. (2017). *Automation of a Business Process Using Robotic Process Automation (RPA): A Case Study*. 65-71. 10.1007/978-3-319-66963-2_7.
- Aldana, S. (3 de febrero de 2017). *Automatización de los servicios: el futuro de los negocios*. LR la República. Recuperado el 16 de 04 de 2021, de <https://www.larepublica.co/internet-economy/automatizacion-de-los-servicios-el-futuro-de-los-negocios-2467486>
- ATEC. (2022). *Pirámide CIM de Automatización Industrial*. Recuperado el 06 de Junio de 2022, de [atec-energy.com](https://atec-energy.com/Blog/piramide-de-automatizacion-cim): <https://atec-energy.com/Blog/piramide-de-automatizacion-cim>
- Beer, F. & Johnston, E. (2007). *Mecánica vectorial para Ingenieros*. McGraw-Hill, Octava Edición.
- CCI-es (2022). *CIBERSEGURIDAD EN LA PIRÁMIDE DE AUTOMATIZACIÓN INDUSTRIAL* [Archivo PDF]. <https://www.cci-es.org/wp-content/uploads/Guia-Bolsillo-Piramide-de-Automatizacion-Industrial-1.pdf>
- Correos del Ecuador. (2016). *Correos del ecuador a través del tiempo*. <https://www.correosdelecuador.gob.ec/correos-del-ecuador-a-traves-del-tiempo/>.
- De la Hoz, A. (8 de septiembre de 2016). *Los centros de Servicios Compartidos (CSC) y la automatización de procesos a través de Robots (RPA)*. linkedin. Recuperado el 16 de 04 de 2021, de <https://www.linkedin.com/pulse/los-centros-de-servicios-compartidos-csc-y-la-trav%C3%A9s-robots/?originalSubdomain=es>

- Deloitte (2016). *La era de la Automatización Implementación de Robotics en los Centros de Servicios Compartidos* [Archivo PDF].
https://www2.deloitte.com/content/dam/Deloitte/mx/Documents/strategy/pov_robotics.pdf
- Deloitte (2017). *Automatización Robótica de Procesos (RPA)* [Archivo PDF].
https://www2.deloitte.com/content/dam/Deloitte/mx/Documents/strategy/Automatizacion_Rob%C3%B3tica_Procesos.pdf
- Digikey Electronics (2022). *HX711 Datasheet by SparkFun Electronics* [Archivo PDF].
<https://www.digikey.com/htmldatasheets/production/1836471/0/0/1/hx711.html>
- Dow Jones & Company Inc. (2018). *Big Companies Fine-Tune The Bot Revolution -- WSJ*. Dow Jones Institutional
- Fixon, S. (2001). *Three Perspectives on Modularity – A Literature Review of a Product Concept for Assembled Hardware Products*. Massachusetts Institute of Technology, ESD-WP-2001-06.
- Fonseca, D., Pérez, W. & Faurés, M. (2013). *PASARELA DE PAGOS PARA LA SEGURIDAD DE TRANSACCIONES BANCARIAS EN LINEA*. Revista de investigación; Editada por el Área de Innovación y desarrollo.
<https://www.3ciencias.com/wp-content/uploads/2013/08/pasarela-de-pagos.pdf>
- Frigieri, E. P., Mazzer, D., & Parreira, L. F. C. G. (2015). *M2M Protocols for Constrained Environments in the Context of IoT: A Comparison of Approaches*. XXXIII Brazilian Telecommunications Symposium (SBrt), (September), 1–4.
<http://doi.org/10.14209/sbrt.2015.2>
- Hernández, S., Ramirez, J., Rodrigues, C. & Cabrera, N. (2008). *Diseño y Construcción de un Sistema Automático de Pesaje y Control Estadístico Mediante Básculas Electrónicas Inteligentes* [Tesis para obtención de título de ingeniero Electromecánico, Universidad Nacional de Loja]
<https://dspace.unl.edu.ec/jspui/handle/123456789/16758>

- Honeywell (2022). *Hall effect sensing and application* [Archivo PDF].
<http://denethor.wlu.ca/pc300/projects/sensors/hallbook.pdf>
- HVRMA (19 de Marzo de 2021). *What Is Electromagnetic Lock and How It Works?*.
Recuperado de <https://www.hvrmagnet.com/blog/what-is-electromagnetic-lock-and-how-it-works/>
- Kyowa (2022). *Technical Strain gages* [Archivo PDF]. https://www.kyowa-ei.com/eng/file/download/technical/strain_gages/pdf_index_001_eng.pdf
- Marín, L. (2016). *Centro de servicios compartidos "Recurso humano y productividad"* [Tesis para obtención de título de Magíster en Administración con Énfasis en Gerencia de Desarrollo Humano, Universidad EAFIT]
<https://core.ac.uk/download/pdf/81651447.pdf>
- Marone, J. (2012). *Microcontroladores de Atmel*. [Archivo PDF] Obtenido de
https://users.exa.unicen.edu.ar/catedras/tmicrocon/Material/3_Overview_Microcontroladores_ATMEL.pdf
- Nilson, J., & Riedel, S. (2005). *Circuitos Eléctricos*. Pearson Education Inc, Séptima Edición.
- Norma Ecuatoriana de la Construcción (2013). *Capítulo 15 Instalaciones Electromecánicas. Código NEC-SB-IE*. <https://www.habitatyvivienda.gob.ec/wp-content/uploads/downloads/2019/03/NEC-SB-IE-Final.pdf>
- Norma Ecuatoriana de la Construcción (2018). *Instalaciones Eléctricas. Código NEC-SB-IE*. <https://www.habitatyvivienda.gob.ec/wp-content/uploads/downloads/2019/03/NEC-SB-IE-Final.pdf>
- Norma Ecuatoriana de la Construcción (2018). *Instalaciones Eléctricas. Código NEC-SB-IE*. <https://www.habitatyvivienda.gob.ec/wp-content/uploads/downloads/2019/03/NEC-SB-IE-Final.pdf>
- NOVUS Automation (2022). *Conceptos Fundamentales De RS485 Y RS422* [Archivo PDF].

<https://www.novusautomation.com/downloads/Arquivos/conceptos%20fundamentales%20de%20rs485%20y%20rs422%20-%20espa%C3%B1ol.pdf>

Pérez, R., Ahuett, H, & Molina, A., Rodríguez, G. (10 de febrero de 2005). *El Diseño Modular en el contexto del desarrollo de Máquinas Herramienta Reconfigurables*. Ingeniería Mecánica, vol. 8, núm. 2, mayo-agosto, 2005, pp. 51-58. Recuperado el 06 de 07 de 2021, de <https://www.redalyc.org/pdf/2251/225118187007.pdf>

Ricoh. (2019). *Casilleros Inteligentes* [Archivo PDF]. https://www.ricoh.es/media/Ricoh-Smart-Lockers-All-ES-online_tcm77-36004.pdf

Ricoh. (2019). *Casilleros Inteligentes* [Archivo PDF]. https://www.ricoh.es/media/Ricoh-Smart-Lockers-All-ES-online_tcm77-36004.pdf

Rodríguez, P. (2013). *Diseño de Interfaces Hombre - Máquina (HMI)*. [Archivo PDF]. Obtenido de sistemamid.com: <http://sistemamid.com/panel/uploads/biblioteca/1/619/620/621/3620.pdf>

Salichs, E. (2012). *Desarrollo de un sistema HMI para un almacén automatizado [Tesis para obtención de título de Ingeniero Industrial, Universidad Carlos III de Madrid]* https://e-archivo.uc3m.es/bitstream/handle/10016/16073/Proyecto_ESTHER_SALICHS.pdf

Semle, A. (2016). *Protocolos IoT para considerar*. [Archivo PDF]. Obtenido de AADECA: https://editores-srl.com.ar/sites/default/files/aa2_semle_protocolos_ilot.pdf

Stone, R. (1997) *Towards a Theory of Modular Design*. [Tesis Doctoral, University of Texas at Austin], United States of America.

Tello, V. (2015). *Modelo de gestión de comercialización para incrementar el nivel de ventas en la empresa autosustentable comercializadora de servicios de envío –*

recepción de correspondencia y paquetería “Correos Del Ecuador Cde Ep” sucursal Riobamba [Tesis para obtención de título de Ingeniería comercial, Escuela Superior Politécnica De Chimborazo]

<http://dspace.esPOCH.edu.ec/bitstream/123456789/10035/3/102T0102.pdf>

Tseng, Mitchell M., Yue Wang, Roger J. Jiao. (2018) Modular Design. In: Chatti S., Laperrière L., Reinhart G., Tolio T., *The International Academy for Production (eds) CIRP Encyclopedia of Production Engineering*. Springer, Berlin, Heidelberg

Ulrich K. & Eppinger S. (2013). *Diseño y desarrollo de productos*. McGraw-Hill, Quinta Edición.

Weis, O. (20 de Octubre de 2021). *Guía de la comunicación RS485*. Recuperado de <https://www.eltima.com/es/article/rs485-communication-guide/>

Worton. (15 de Julio de 2021). *T568A y T568B: dos estándares de cable de red RJ45*. Recuperado de <https://community.fs.com/es/blog/t568a-vs-t568b-difference-between-straight-through-and-crossover-cable.html>

XATAKA. (25 Julio 2019). *Raspberry Pi 4 es oficial una completa actualización con procesador Cortex-A72, hasta 4 GB de RAM y desde 35 dólares*. Obtenido de AADECA: <https://www.xataka.com/ordenadores/raspberry-pi-4-caracteristicas-precio-ficha-tecnica>

ANEXOS