



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE MECÁNICA
CARRERA INGENIERÍA INDUSTRIAL

**DESARROLLO DE UN SISTEMA INTELIGENTE PARA LA
GESTIÓN DE LA RECOLECCIÓN DE DESECHOS SÓLIDOS
URBANOS BASADO EN COMUNICACIÓN INALÁMBRICA**

Trabajo de Integración Curricular

Tipo: Propuesta Tecnológica

Presentado para optar al grado académico de:

INGENIERA/O INDUSTRIAL

AUTORES:

LESSLY JOSÉ GUAMÁN PACHECO

KEVIN ALEX LÓPEZ LÓPEZ

Riobamba - Ecuador

2023



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE MECÁNICA
CARRERA INGENIERÍA INDUSTRIAL

**DESARROLLO DE UN SISTEMA INTELIGENTE PARA LA
GESTIÓN DE LA RECOLECCIÓN DE DESECHOS SÓLIDOS
URBANOS BASADO EN COMUNICACIÓN INALÁMBRICA**

Trabajo de Integración Curricular

Tipo: Propuesta Tecnológica

Presentado para optar al grado académico de:

INGENIERA/O INDUSTRIAL

AUTORES: LESSLY JOSÉ GUAMÁN PACHECO

KEVIN ALEX LÓPEZ LÓPEZ

DIRECTOR: Ing. EDUARDO FRANCISCO GARCÍA CABEZAS Mgs.

Riobamba - Ecuador

2023

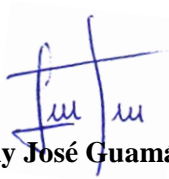
©2023, Lessly José Guamán Pacheco; & Kevin Alex López López

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

Nosotros, Lessly José Guamán Pacheco y Kevin Alex López López declaramos que el presente Trabajo de Integración Curricular es de nuestra autoría y los resultados de este son auténticos. Los textos en el documento que provienen de otras fuentes están debidamente citados y referenciados.

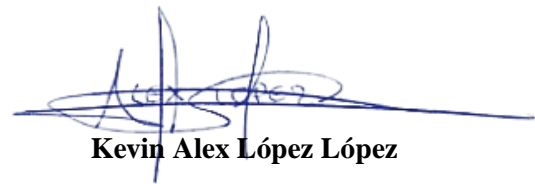
Como autores asumimos la responsabilidad legal y académica de los contenidos de este Trabajo de Integración Curricular; el patrimonio intelectual pertenece a la Escuela Superior Politécnica de Chimborazo.

Riobamba, 02 de agosto del 2023



Lessly José Guamán Pacheco

210098534-6



Kevin Alex López López

060436508-0

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE MECÁNICA
CARRERA INGENIERÍA INDUSTRIAL

El Tribunal del Trabajo de Integración Curricular certifica que: El Trabajo de Integración Curricular; Tipo: Propuesta Tecnológica, **DESARROLLO DE UN SISTEMA INTELIGENTE PARA LA GESTIÓN DE LA RECOLECCIÓN DE DESECHOS SÓLIDOS URBANOS BASADO EN COMUNICACIÓN INALÁMBRICA**, realizado por los señores: **LESSLY JOSÉ GUAMÁN PACHECO Y KEVIN ALEX LÓPEZ LÓPEZ**, ha sido minuciosamente revisado por los Miembros del Tribunal del Trabajo de Integración Curricular, el mismo que cumple con los requisitos científicos, técnicos, legales, en tal virtud que el Tribunal autoriza su presentación.

NOMBRE	FIRMA	FECHA
Ing. José Francisco Pérez Fiallos PRESIDENTE DEL TRIBUNAL		2023-08-02
Ing. Eduardo Francisco García Cabezas DIRECTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR		2023-08-02
Ing. Julio César Moyano Alulema ASESOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR		2023-08-02

DEDICATORIA

El presente trabajo de titulación está dedicado a mis padres, Alonso y Rebeca, por su amor, paciencia y apoyo incondicional durante toda mi vida, sin ustedes, no habría llegado a conseguir este logro importante.

A mis hermanos Edwin, Lenin, Marilyn y Anthony, por ser mi guía y ejemplo de superación, por permitirme ser partícipe de los momentos buenos y malos, por hacerme sentir orgulloso de las personas que se están convirtiendo.

A mis abuelos, tíos y primos quienes siempre han estado ahí para mí, por su sabiduría, por sus consejos y por su comprensión.

A la familia López Cordovez quienes me han acogido en su hogar como un hijo más, por su amor y apoyo incondicional, que me ha motivado a seguir adelante y a alcanzar este logro.

A mis amigos a los cuales considero como mis hermanos, por corregirme cuando era necesario, ayudarme cuando lo necesitaba, consolarme en los momentos tristes, por brindarme su amistad en los momentos buenos y malos.

A mi compañera de tesis, por su dedicación y compromiso, por siempre motivarme y sacar lo mejor de mí.

Kevin López

AGRADECIMIENTO

A mi familia, en especial a mis padres, quienes me apoyaron incondicionalmente durante toda mi formación académica. Su infinito apoyo, comprensión y paciencia han sido fundamentales para que yo haya podido llegar hasta aquí y culminar mi carrera.

También agradezco a mis hermanos y a mis amigos por el apoyo emocional que me han ofrecido durante estos largos años. Sus ánimos y palabras de aliento me han ayudado a seguir adelante.

A la Escuela Superior Politécnica de Chimborazo, a la Facultad de Mecánica y a la Carrera de Ingeniería Industrial por formarme a lo largo de todo este tiempo.

Agradezco enormemente a los docentes y al personal por ofrecerme una gran educación y por guiarme a lo largo de mi formación académica preparándome para el futuro y a desarrollar mis habilidades.

Finalmente, el sincero agradecimiento a los ingenieros Eduardo García y Julio Moyano, por brindarme la oportunidad para desarrollar este trabajo de titulación, con su incondicional soporte y dirección.

Kevin López

ÍNDICE DE CONTENIDO

ÍNDICE DE TABLAS.....	x
ÍNDICE DE ILUSTRACIONES.....	xi
ÍNDICE DE ANEXOS.....	xiii
RESUMEN.....	xiv
SUMMARY.....	xv
INTRODUCCIÓN.....	1

CAPÍTULO I

1	DIAGNÓSTICO DEL PROBLEMA.....	2
1.1	Antecedentes.....	2
1.2	Planteamiento del problema.....	3
1.3	Justificación.....	4
1.4	Objetivos.....	5
1.4.1	<i>Objetivo general</i>	5
1.4.2	<i>Objetivos específicos</i>	5

CAPÍTULO II

2	MARCO TEÓRICO.....	6
2.1	Antecedentes de investigación.....	6
2.2	Referencias teóricas.....	7
2.2.1	<i>Gestión de desechos sólidos urbanos</i>	7
2.2.1.1	<i>Desecho</i>	8
2.2.1.2	<i>Gestión integral de desechos sólidos en el Ecuador</i>	8
2.2.1.3	<i>Normativa vigente en el Ecuador</i>	8
2.2.2	<i>Sistema inteligente para la gestión de recolección de residuos sólidos urbanos</i>	10
2.2.2.1	<i>Internet de las cosas</i>	10
2.2.2.2	<i>Visión artificial</i>	11
2.2.2.3	<i>Sistema interfaz hombre máquina</i>	12
2.2.3	<i>Componentes electrónicos</i>	14
2.2.3.1	<i>Raspberry Pi</i>	15
2.2.3.2	<i>Arduino</i>	15
2.2.3.3	<i>GPS</i>	15

2.2.4	<i>Actuadores eléctricos</i>	15
2.2.4.1	<i>Servomotor</i>	16
2.2.5	<i>Comunicación inalámbrica</i>	16
2.2.5.1	<i>Red inalámbrica Wifi</i>	16
2.2.5.2	<i>Web server</i>	17
2.2.6	<i>Software</i>	17
2.2.6.1	<i>SolidWorks</i>	17
2.2.6.2	<i>Proteus</i>	18
2.2.7	<i>Lenguajes de programación</i>	18

CAPÍTULO III

3	MARCO METODOLÓGICO	19
3.1	Consolidación general del prototipo	19
3.2	Definición de requerimientos	20
3.3	Arquitectura general del prototipo	21
3.4	Diseño CAD	22
3.4.1	<i>Diseño de la estructura</i>	22
3.4.2	<i>Diseño de la cubierta</i>	23
3.4.3	<i>Diseño de puerta trasera</i>	24
3.4.4	<i>Diseño del contenedor de basura</i>	24
3.5	Hardware	26
3.5.1	<i>Hardware para la visión artificial</i>	26
3.5.2	<i>Hardware para la comunicación inalámbrica</i>	28
3.5.3	<i>Hardware para el bloqueo de la puerta</i>	29
3.5.4	<i>Hardware para el sistema de alerta</i>	30
3.5.5	<i>Hardware para alerta de audio</i>	32
3.5.6	<i>Hardware de procesamiento</i>	33
3.5.7	<i>Fuente de alimentación</i>	35
3.5.8	<i>Diagramas de conexión del hardware</i>	36
3.6	Software	39
3.6.1	<i>Lenguajes de programación</i>	39
3.6.2	<i>Plataforma IoT</i>	40
3.7	Desarrollo de la programación	40
3.7.1	<i>Visión artificial</i>	41
3.7.2	<i>Codificación para la ubicación de los contenedores de basura</i>	43
3.7.3	<i>Codificación para la activación del servo motor</i>	44

3.7.4	<i>Codificación para la interfaz gráfica remota</i>	45
3.7.5	<i>Codificación general del sistema</i>	46

CAPÍTULO IV

4	MARCO DE ANÁLISIS E INTERPRETACIÓN DE RESULTADOS	48
4.1	Prototipo	48
4.2	Visión artificial	50
4.2.1	<i>Eficiencia de la visión artificial</i>	50
4.2.1.1	<i>Prueba 1</i>	52
4.2.1.2	<i>Prueba 2</i>	53
4.2.1.3	<i>Prueba 3</i>	54
4.3	Validación de la flexibilidad del sistema	56
4.4	Interfaz gráfica	57
4.4.1	<i>Interfaz gráfica web</i>	57
4.4.2	<i>Interfaz gráfica desde la aplicación móvil</i>	58
4.5	Estabilidad de los datos de ubicación del prototipo	59
4.5.1	<i>Ubicación 1</i>	60
4.5.2	<i>Ubicación 2</i>	61
4.6	Validación punto más cercano	63
4.6.1	<i>Prueba 1</i>	63
4.6.2	<i>Prueba 2</i>	65
4.6.3	<i>Prueba 3</i>	67
4.7	Validación del sistema de bloqueo de la puerta	69
4.8	Validación del sistema de alarma	70
4.9	Costos	71

	CONCLUSIONES	73
--	---------------------------	----

	RECOMENDACIONES	74
--	------------------------------	----

BIBLIOGRAFÍA

ANEXOS

ÍNDICE DE TABLAS

Tabla 3-1: Elementos utilizados en el contenedor de basura	26
Tabla 3-2: Comparación de cámaras	27
Tabla 3-3: Comparación de módulos GPS.	28
Tabla 3-4: Comparación de actuadores eléctricos	30
Tabla 3-5: Comparación de sensores.....	31
Tabla 3-6: Características del parlante	32
Tabla 3-7: Características amplificador de audio mono TDA2030A.....	33
Tabla 3-8: Comparación de ordenadores.....	34
Tabla 3-9: Comparación fuentes de alimentación	35
Tabla 3-10: Consumo eléctrico del prototipo	36
Tabla 3-11: Detalles de conexión del hardware.....	38
Tabla 3-12: Comparación de Lenguajes de Programación.	40
Tabla 4-1: Matriz de confusión	51
Tabla 4-2: Datos de la prueba 1.....	52
Tabla 4-3: Matriz de confusión de la prueba 1	53
Tabla 4-4: Datos de la prueba 2.....	53
Tabla 4-5: Matriz de confusión de la prueba 2	54
Tabla 4-6: Datos de la prueba 3.....	55
Tabla 4-7: Matriz de confusión de la prueba 3	55
Tabla 4-8: Datos emitidos por el prototipo en la ubicación 1	60
Tabla 4-9: Datos emitidos por el prototipo en la ubicación 2	61
Tabla 4-10: Datos de la prueba 1 en la interfaz gráfica	63
Tabla 4-11: Datos de la ruta más corta en la prueba 1	65
Tabla 4-12: Datos de la prueba 2 en la interfaz gráfica	65
Tabla 4-13: Datos de la ruta más corta en la prueba 2.....	67
Tabla 4-14: Datos de la prueba 3 en la interfaz gráfica	67
Tabla 4-15: Datos de la ruta más corta en la prueba 3.....	69
Tabla 4-16: Tiempos de respuesta del sistema de bloqueo.....	69
Tabla 4-17: Tiempos de respuesta del sistema de alarma.....	70
Tabla 4-18: Costos directos del prototipo.....	72
Tabla 4-19: Costos indirectos del prototipo	72
Tabla 4-20: Costos totales del prototipo.....	72

ÍNDICE DE ILUSTRACIONES

Ilustración 2-1:	Arquitectura del SGen con dispositivos IoT	11
Ilustración 2-2:	Logo de Adafruit Industries.....	12
Ilustración 2-3:	Logo ThingSpeak.....	13
Ilustración 2-4:	Logo Blink IoT.....	13
Ilustración 2-5:	Logo de Ubidots.....	14
Ilustración 2-6:	Servomotor.....	16
Ilustración 2-7:	Esquema de una WLAN.....	17
Ilustración 2-8:	Logotipo SolidWorks	18
Ilustración 3-1:	Consolidación general del prototipo	20
Ilustración 3-2:	Arquitectura general del prototipo.....	21
Ilustración 3-3:	Dimensiones de la estructura.....	23
Ilustración 3-4:	Diseño CAD de la cubierta.....	23
Ilustración 3-5:	Dimensiones de la cubierta.....	24
Ilustración 3-6:	Diseño CAD y dimensiones de la puerta trasera.....	24
Ilustración 3-7:	Diseño CAD del contenedor de basura.....	25
Ilustración 3-8:	Vista frontal y posterior del contenedor de basura.....	25
Ilustración 3-9:	Parlante	32
Ilustración 3-10:	Módulo amplificador de audio mono TDA2030A.....	33
Ilustración 3-11:	Esquema de conexión completo del hardware	37
Ilustración 3-12:	Recolección de datos.....	41
Ilustración 3-13:	Análisis y almacenamiento de datos.....	42
Ilustración 3-14:	Detección del nivel de llenado del basurero cuando está al 50%	43
Ilustración 3-15:	Obtención de la ubicación de los basureros	43
Ilustración 3-16:	Procedimiento para el bloqueo del contenedor de basura	44
Ilustración 3-17:	Diagrama de flujo del funcionamiento del sistema.....	46
Ilustración 3-18:	Diagrama de flujo del funcionamiento del sistema.....	47
Ilustración 4-1:	Prototipo finalizado	48
Ilustración 4-2:	Parte interna del prototipo	49
Ilustración 4-3:	Funcionamiento del algoritmo de visión artificial.	50
Ilustración 4-4:	VP, FP y FN en las imágenes capturadas.	51
Ilustración 4-5:	VN y FN en las imágenes capturadas	52
Ilustración 4-6:	Prueba 1 del algoritmo de visión artificial	53
Ilustración 4-7:	Prueba 2 del algoritmo de visión artificial	54
Ilustración 4-8:	Prueba 3 del algoritmo de visión artificial	55

Ilustración 4-9:	Comunicación del sistema.	56
Ilustración 4-10:	Desarrollo inicial de la interfaz gráfica remota.....	56
Ilustración 4-11:	Interfaz gráfica web cuando los basureros están llenos.	57
Ilustración 4-12:	Interfaz gráfica desde la app móvil.....	58
Ilustración 4-13:	Desbordamiento de contenedor de basura	59
Ilustración 4-14:	Prueba 1 del punto más cercano.	64
Ilustración 4-15:	Ruta más corta al basurero 1 en la prueba 1.	64
Ilustración 4-16:	Ruta más corta al basurero 2 en la prueba 1.	64
Ilustración 4-17:	Prueba 2 del punto más cercano.	66
Ilustración 4-18:	Ruta más corta al basurero 1 en la prueba 2.	66
Ilustración 4-19:	Ruta más corta al basurero 2 en la prueba 2.	67
Ilustración 4-20:	Prueba 3 del punto más cercano.	68
Ilustración 4-21:	Ruta más corta al basurero 1 en la prueba 3.	68
Ilustración 4-22:	Ruta más corta al basurero 2 en la prueba 3.	69

ÍNDICE DE ANEXOS

ANEXO A: PRUEBAS DE CONEXIÓN DE LA RASPBERRY PI CON EL MÓDULO GPS

ANEXO B: DISEÑO DE LA PLACA PCB EN EL SOFTWARE PROTEUS

ANEXO C: VISTA 3D DE LA PLACA PCB

ANEXO D: ENSAMBLE DE COMPONENTES ELECTRÓNICOS EN LA PLACA PCB

ANEXO E: SISTEMA DE BLOQUEO DE ACCESO A LA PUERTA DEL BASURERO

ANEXO F: CÁMARA EMPLEADA PARA LA VISIÓN ARTIFICIAL

ANEXO G: PROGRAMACIÓN DE VISIÓN ARTIFICIAL

ANEXO H: CONSTRUCCIÓN DEL PROTOTIPO

ANEXO I: VIDEO DEL FUNCIONAMIENTO DEL PROTOTIPO

RESUMEN

El presente trabajo de titulación tuvo como objetivo desarrollar un sistema inteligente para la gestión de la recolección de desechos sólidos urbanos basado en comunicación inalámbrica, para lograrlo se definieron como requerimientos la generación de un modelo de red inalámbrica que considere a los basureros como nodos que se enlazan a un interfaz gráfica remota, la codificación de un algoritmo de visión artificial que identifique el nivel de llenado de los contenedores de basura y la obtención de su ubicación, como resultado se obtuvo un sistema inteligente, codificado en Python y procesado en una Raspberry Pi 4 Model B que mediante un modelo de visión artificial basado en el reconocimiento óptico de caracteres (OCR) detecta el nivel de llenado de los contenedores de basura y envía esta información a una interfaz gráfica remota creada en la plataforma Blynk IoT que monitorea el estado de los contenedores de basura, muestra su ubicación, el contenedor lleno más cercano y le permite al usuario controlar el acceso a los mismos, los prototipos tienen acoplado un servomotor MG996R que bloquea la puerta cuando están llenos y un sensor infrarrojo FC-51 que activa una alerta de voz cuando se intenta acceder a estos. Con las pruebas de funcionalidad se determinó que el algoritmo de visión artificial tiene una eficiencia del 86,36% cuando es usado con desperdicios de colores oscuros, la función del basurero más cercano es útil únicamente si se emplea en zonas residenciales, el tiempo de respuesta del sistema de bloqueo de la puerta es de 28,19 segundos, mientras que el del sistema de alarma es de 20,11 segundos. Para trabajos futuros se recomienda mantener una iluminación constante y probar el prototipo con una red de internet estable.

Palabras clave: <VISIÓN ARTIFICIAL> <COMUNICACIÓN INALÁMBRICA>
<INTERFAZ GRÁFICA> <DESECHOS SÓLIDOS> <BASURERO INTELIGENTE>.

1656-DBRA-UPT-2023



SUMMARY

The objective of this degree work was to develop an intelligent system for the management of urban solid waste collection based on wireless communication. To achieve this, some requirements have been defined, such as the generation of a wireless network model that considers the dumpsters as nodes connected to a remote graphical interface, the coding of an artificial vision algorithm that identifies the fill level of the dumpsters, and the determining of their location. As a result, an intelligent system coded in Python and processed on a Raspberry Pi 4 Model B was obtained, which by means of an artificial vision model based on optical character recognition (OCR) detects the filling level of the garbage containers and sends this information to a remote graphic interface created on the Blynk IoT platform. The remote graphical interface monitors the status of dumpsters, displays their location and the closest filled bin, as well as allows the user controls access to them. The prototypes are coupled with an MG996R servo motor that locks the door when full and an FC-51 infrared sensor that triggers a voice alert when access is attempted. Functional tests showed that the image processing algorithm has an efficiency of 86.36% when used with dark-colored waste, and the "closest bin" function only proved useful when used in residential areas. Finally, the response time of the door locking system is 28.19 seconds, while that of the alarm system is 20.11 seconds. For future work, it is recommended to maintain constant lighting and test the prototype with a stable internet network.

Keywords: <ARTIFICIAL VISION> <WIRELESS COMMUNICATION> <GRAPHIC INTERFACE> < SOLID WASTE > <INTELLIGENT WASTE BIN>.



Lic. Angela Cecibel Moreno Novillo
0602603938

INTRODUCCIÓN

En el Ecuador la mayor cantidad de habitantes reside en zonas urbanas, lugares que generan grandes cantidades de desechos sólidos, para brindar el servicio de recolección de estos desechos, cada Gobierno Autónomo Descentralizado Municipal (GADM) desplaza camiones recolectores que realizan recorridos por cada contenedor de la ciudad según un horario y ruta planificada, esta gestión provoca que se utilicen recursos públicos sin conocer si se requiere o no del servicio. En ocasiones los contenedores tienen un nivel bajo de llenado, por lo que se producen desperdicios de recursos económicos y humanos, a más de contribuir a la contaminación ambiental por la movilización innecesaria de vehículos pesados. Por otro lado, cuando se ha alcanzado el nivel de llenado máximo y la recolección no es oportuna, se crean microbasurales que obstruyen las alcantarillas y afectan la imagen de la ciudad.

En el presente trabajo de titulación se propone el desarrollo de un sistema inteligente para la gestión de la recolección de desechos sólidos urbanos basado en comunicación inalámbrica que permita monitorear el nivel de llenado de los contenedores de basura a través de una interfaz gráfica web y móvil amigable con el usuario, que facilite la gestión del proceso de recolección y optimice el uso de recursos.

Para alcanzar los objetivos planteados se llevó a cabo la revisión bibliográfica de sistemas inteligentes similares, y con ello se definieron los requerimientos del sistema, se diseñó y desarrolló el prototipo utilizando visión artificial y finalmente se realizaron pruebas de funcionalidad que validaron el sistema.

CAPÍTULO I

1 DIAGNÓSTICO DEL PROBLEMA

1.1 Antecedentes

Debido al acelerado crecimiento demográfico y a la concentración poblacional en las zonas urbanas, la cantidad de basura generada diariamente va en aumento, creándose un escenario difícil para la gestión de recolección de desechos sólidos, que inquieta a los Gobiernos Autónomos Descentralizados (GAD) Municipales del Ecuador, quienes son los responsables de prestar este servicio público.

Según Borozdukhin; Dolinina; Pechenkin (2016) las ciudades manejan la recolección de residuos sólidos realizando recorridos de acuerdo a rutas y horarios preestablecidos, lo que origina un proceso poco eficiente que no favorece al hecho de que el servicio de recolección y transporte de desechos sólidos es uno de los componentes más costosos, que requieren de una minuciosa planificación.

Frente a esta situación los gobiernos municipales se ven obligados a buscar mecanismos apropiados para mantener un servicio de calidad, a costos razonables (EMAC, 2019), como son los avances tecnológicos en el área del Internet de las Cosas (IoT) (Fujdiak et al., 2016, pp.1-6).

En Ecuador, según la ministra de Telecomunicaciones y de la Sociedad de la Información, Vianna Maino, las telecomunicaciones tienen pilares fundamentales en dos ejes dentro del Plan Nacional de Creación de Oportunidades 2021-2025, crear nuevas oportunidades, brindar bienestar en zonas rurales y la simplificación de trámites. Maino menciona que se están comenzando a generar políticas para propiciar las nuevas tecnologías de robótica, analítica, inteligencia artificial, tecnologías cognitivas, nanotecnología e IoT, sin embargo, para que todo ello suceda, primero es necesaria la conectividad. (BNamericas, 2021)

Por otro lado, en países desarrollados como España, ya se han implementado las TICS, como es el caso de la empresa municipal LIPASAM encargada de la recolección de residuos urbanos en Sevilla que, a través del proyecto LIFE EWAS propusieron la utilización del sensor de medición QUAMTRA, dispositivo que dota de inteligencia a los contenedores, lo cual les permite obtener datos de su estado en tiempo real mediante un haz de ultrasonidos que puede ser configurado dependiendo de la distancia a la cual se desea que se notifique la recolección, transmite los datos de manera inalámbrica hacia la plataforma encargada de la gestión y análisis para la recolección municipal de los residuos. De esta manera el gestor encargado de la recolección obtiene información útil sobre las rutas que se deben realizar. (Guerrero, 2019)

De igual manera, Barcelona ha implementado unos contenedores inteligentes que detectan el nivel de residuos que contienen y así los trabajadores pueden planificar una ruta de recogida más eficiente. Algunos contenedores se conectan directamente a repositorios subterráneos y los residuos son aspirados por el vacío a través de tuberías subterráneas. Reduciendo así olores, ruido y la congestión causada por los camiones recolectores. Más tarde estos residuos son quemados y la energía producida se utiliza para el sistema de calefacción de la ciudad. (Triano, 2020, p.26)

En 2011 la compañía surcoreana Ecube Labs creó una solución inteligente de gestión de residuos mediante contenedores que compactan la basura con la utilización de energía solar, además los contenedores cuentan con sensores que permiten monitorear en tiempo real su estado, con lo cual optimizan las rutas de recolección, esta propuesta ha permitido reducir los costos operativos en el país y ha brindado mejoras en el saneamiento público. (Ecube Labs, 2022)

En Latinoamérica, Argentina implementó en la ciudad de Buenos Aires el proyecto startup FLEXBIT nacido en el centro de emprendedores de la UTN en 2015, busca soluciones para la recolección de residuos mediante el IoT utilizando la tecnología cognitiva de IBM WATSON, el cual conecta a internet los contenedores de residuos lo que permite el monitoreo en tiempo real del nivel de llenado de dichos contenedores con el objetivo de optimizar la gestión de recolección. Posee un sensor autoinstalable capaz de conectarse con una laptop o smartphone apareándose a la red wifi de la compañía permitiendo su gestión mediante una aplicación móvil enviando alertas de nivel de llenado y mediante su plataforma web monitorea su ubicación con el api de Google Maps. (Guerrero, 2019)

1.2 Planteamiento del problema

En Ecuador en el año 2020 se generaron en promedio 12 612,5 toneladas diarias de residuos sólidos que corresponden a la producción de 0,83 kg de residuos por cada habitante del sector urbano (INEC et al., 2021, p.5), para recolectar esta excesiva cantidad de desechos sólidos se realizan recorridos en camiones recolectores por cada contenedor de la ciudad de acuerdo a un horario y ruta establecida, este tipo de gestión resulta ineficiente, ya que se realizan recorridos sin ninguna certeza de que los contenedores requieren ser vaciados, y existe una alta probabilidad de emplear tiempo, combustible y mano de obra en vano. Así mismo, se dan los casos contrarios, en donde los contenedores han superado su capacidad máxima y el servicio de recolección se realiza tardíamente, ocasionando malestar en los usuarios.

En países subdesarrollados, como Ecuador se atraviesa un panorama complicado con presencia de problemas de índole económica, tecnológica, educativa, social y cultural, por lo que sus gobernantes priorizan aspectos urgentes que afectan a su población, descuidando el problema de los desechos sólidos, esto sumado al escaso interés de sus habitantes que se muestran indiferentes

y poco participativos en contribuir a la solución de esta problemática. (Boadi y Kuitunen, 2003, pp.211-218)

Tal es el caso de los “minadores” o “recicladores de base”, quienes en su afán de encontrar en los contenedores de basura objetos reciclables como papel, plásticos y metales, crean vertidos en las calles y veredas (Municipio del Distrito Metropolitano de Quito et al., 2016, p.85) que obstruyen el sistema de drenaje provocando inundaciones en época de lluvia y afectando gravemente la higiene de la ciudad. De igual manera, la ciudadanía carece de educación ambiental, pues cuando se encuentra con un contenedor completamente lleno, opta por apilar los residuos alrededor del mismo, creando un foco de atracción para animales que transmiten enfermedades y dispersan la basura por los alrededores perjudicando la imagen de la ciudad.

Entonces, el problema radica en la ineficiencia del sistema de gestión de recolección de desechos sólidos por la falta de atención de los gobernantes, sin embargo, la deficiente cultura ambiental de la sociedad empeora cada vez más la situación.

1.3 Justificación

En los últimos años las zonas urbanas han crecido aceleradamente, por lo que la gestión de recolección de residuos sólidos se ha convertido en un desafío para los GAD Municipales del Ecuador, que se enfrentan a elevados costos y una evidente ineficiencia en el servicio, ya que según datos proporcionados en el 2010 por el Programa Nacional de Gestión Integral de Desechos Sólidos, el MIDUVI y otras instituciones, se determinó que la cobertura del servicio de recolección de residuos sólidos tiene una cobertura nacional promedio en zonas urbanas del 84,2% y del 54,1% en el área rural (Ministerio del Ambiente, Agua y Transición Ecológica, 2020), exhibiendo la necesidad de un cambio en la gestión tradicional de recolección de desechos sólidos, paso que no se ha dado, pues al ser competencia única del sector público existe poco interés hacia la innovación y mejora en la calidad del servicio (CNC, 2019, p.44).

Como solución a este problema se propone el desarrollo de un sistema inteligente que se encargue de llevar a cabo la gestión, el cual permita una interacción entre los contenedores de basura y el personal de recolección, intercambiando información sobre el nivel de llenado y ubicación, datos útiles para trazar rutas adecuadas que prioricen la recolección de los basureros que han llegado a su capacidad máxima, con lo cual se optimizan recursos y tiempo, que se traducen en ahorros en combustible, una mayor disponibilidad del servicio, una disminución en las emisiones de CO2 y contaminación auditiva, ya que los viajes realizados serán los estrictamente necesarios. Así mismo, se reducirá la creación de microbasurales, mejorando el ornato de la ciudad, y contribuyendo a construir una ciudad sana, limpia y sostenible.

1.4 Objetivos

1.4.1 Objetivo general

Desarrollar un sistema inteligente para la gestión de la recolección de desechos sólidos urbanos basado en comunicación inalámbrica.

1.4.2 Objetivos específicos

- Definir los requerimientos del sistema mediante la revisión bibliográfica de investigaciones similares.
- Adecuar los contenedores de basura con la inserción de recursos tecnológicos para la generación de alertas de voz locales y bloqueos de acceso al contenedor cuando se alcance su capacidad máxima de llenado.
- Implementar una red inalámbrica de sensores y actuadores que permita la comunicación bidireccional de una central de monitoreo con los contenedores de basura.
- Monitorear el nivel de llenado de los contenedores de basura por medio de una interfaz gráfica remota.
- Programar un algoritmo en la interfaz gráfica remota que mejore la logística de recolección mediante la determinación de la ruta más corta del camión recolector.
- Verificar la funcionalidad del sistema con la realización de pruebas.

CAPÍTULO II

2 MARCO TEÓRICO

2.1 Antecedentes de investigación

Según Torrente; Chifari; Ripa; Giampietro (2020) los países que se encuentran en vías de desarrollo presentan una mala gestión de residuos sólidos urbanos (RSU), esto debido a que no existe una relación cercana entre la administración pública y la ciudadanía. De la misma forma los sistemas de gestión de RSU no reconocen los problemas organizacionales que existen en el interior tales como, los costos de operación, objetos disfuncionales o las dificultades para monitorear o supervisar el desempeño en ciudades (De Mori et al., 2014, pp.50-56), esto provoca que la gestión se convierta en un tema secundario. Lo que no debería pasar ya que una aplicación adecuada de la gestión de RSU es de mucha relevancia para los ciudadanos la misma que debe ser brindada por las organizaciones gubernamentales (Toledo y Quintero, 2021, pp.2-4).

Por su parte Marotta; Moraes; Carbonelli (2017) aseguran que otro problema organizacional es la insuficiencia de cobertura de la recolección y el tratamiento de los RSU, se entiende por cobertura al porcentaje de usuarios que son beneficiarios del servicio de recolección. Si se llegara a comprender la importancia que tiene una buena gestión de recolección de RSU, las organizaciones gubernamentales no dudarían en invertir en la implementación de sistemas inteligentes que puedan llegar a tener una mayor eficiencia, como muestra de esto se mencionan las siguientes investigaciones:

En el trabajo de investigación titulado “Sistema multimodal que permita monitorear y controlar la recolección de desechos sólidos en tiempo real para Smart Cities” publicado en 2018 se menciona que para el desarrollo del sistema se establece un proceso de bloques en el cual, el primero es un bloque de entradas que se encarga de recoger información como, el nivel de llenado, localización, alerta de incendio y manipulación de un contenedor de desechos por medio de sensores, en el siguiente bloque se procesa la información recolectada previamente mediante una placa de Hardware Libre evaluando los datos y en el bloque final que es de salida se envían los valores obtenidos por medio de un sistema de comunicación móvil hacia un servidor en la nube que contiene una aplicación Web con una interfaz amigable con el usuario. Esta aplicación tiene las herramientas necesarias para que el personal encargado administre y gestione el estado de los contenedores y mediante Google Maps se trazan rutas para la recolección de contenedores que se encuentran próximos a llegar a su capacidad, optimizando el proceso de recolección de desechos y mejorando los parámetros ambientales.

Además, se manifiesta que al momento de realizar las pruebas del sistema el tiempo que tardan en actualizarse los datos en la página Web es de 1 a 3 minutos y tiene una pérdida de señal en espacios cerrados, por lo que se recomienda utilizar una operadora de amplia cobertura en el sistema de comunicación móvil. (Morocho et al., 2018, p.1)

Burgos y Prieto (2014) en su investigación titulada “Aplicación de las TIG para la optimización de los servicios de recogida y gestión de los residuos sólidos urbanos del municipio de Aspe (Alicante)” afirman que en la actualidad los Sistemas de Información Geográfica (SIG) constituyen una herramienta imprescindible para resolver problemas que tienen un claro componente geográfico. Dichas tecnologías deberían ocupar un lugar destacado en el conjunto de la gestión municipal, en cualquier caso, más allá de una absoluta resolución automática de problemas, estas técnicas de trabajo se deben contemplar como un recurso para generar, analizar y representar resoluciones que servirán para la toma de decisiones frente a determinados problemas de tipo geográfico. Los autores aseguran que para una ejecución adecuada y óptima de los SIG son necesarios dos tipos de datos, cuantitativos y cualitativos, los cuantitativos son aquellos que emiten datos tales como la latitud y longitud con los cuales se logra establecer una ubicación precisa, y los datos cualitativos son aquellos que consideran opiniones, preferencias y órdenes del personal encargado del proyecto.

De acuerdo a García (2021) para que el sistema de recolección se pueda realizar de manera correcta es necesario que cada contenedor disponga de un dispositivo instalado en su parte superior, en la tapa, la cual debe permanecer siempre cerrada, este dispositivo está compuesto por un sensor ultrasónico que permite detectar los distintos niveles de basura, un sensor de temperatura para monitorear la misma, un sensor de fuego que detecta la presencia de brasas prendidas y una placa Arduino que almacena información y haciendo uso del Internet de las Cosas (IoT) se comunica con el sistema a través de la plataforma Sigfox, posteriormente la información se guarda en el servidor permaneciendo disponible para ser consultada por una computadora, con ello se evita que el operario tenga que encargarse del llenado de una hoja de registro y únicamente se enfoque en su trabajo de recolección, ya que el monitoreo sobre los distintos niveles de basura en cada contenedor le compete al operador responsable de la central de control.

2.2 Referencias teóricas

2.2.1 *Gestión de desechos sólidos urbanos*

Es el conjunto de actividades aplicadas durante el ciclo de vida de los desechos sólidos, por lo general el ciclo de vida consta de las siguientes etapas: generación, recolección, transporte, tratamiento, reciclaje y disposición final de forma segura (Jiménez, 2015, pp.2-3).

2.2.1.1 Desecho

En el Código Orgánico del Ambiente (COA) se lo define como:

Son las sustancias sólidas, semisólidas, líquidas, gaseosas o materiales compuestos resultantes de un proceso de producción, extracción, transformación, reciclaje, utilización o consumo, a cuya eliminación o disposición final se procede conforme a lo dispuesto en la legislación ambiental nacional e internacional aplicable y no es susceptible de aprovechamiento o valorización. (República del Ecuador, 2017, p.90)

2.2.1.2 Gestión integral de desechos sólidos en el Ecuador

La gestión de desechos sólidos en el Ecuador es competencia de los gobiernos autónomos descentralizados municipales (GADM), en un principio esto fue establecida en la Ley Orgánica de Régimen Municipal de 1971, ley que ya no está en vigencia. Actualmente, en la Constitución de la República del Ecuador y el Código Orgánico de Organización Territorial, Autonomía y Descentralización (COOTAD) se dispone que el manejo de desechos sólidos es jurisdicción exclusiva de este nivel de gobierno.

En este contexto, en el Código Orgánico Ambiental (COA), en el artículo 23 se estipula que la Autoridad Ambiental Nacional (AAN) es el Ministerio del Ambiente, por lo que está en facultad de planificar, regular, controlar y manejar integralmente los residuos sólidos, por su parte los gobiernos autónomos descentralizados municipales (GADM) son la Autoridad Ambiental de Aplicación (AAA) en el suministro de este servicio público dentro de su área de jurisdicción. (Consejo Nacional de Competencias, 2019, p.10)

2.2.1.3 Normativa vigente en el Ecuador

La Constitución del Ecuador en el artículo 14 indica que:

*Se reconoce el derecho de la población a vivir en un ambiente sano y ecológicamente equilibrado, que garantice la sostenibilidad y el buen vivir, *sumak kawsay*.* (Asamblea Nacional de la República del Ecuador, 2008, p.29)

Así mismo, en el artículo 264 señala que:

Los gobiernos municipales tendrán las siguientes competencias exclusivas sin perjuicio de otras que determine la ley: [...] Prestar los servicios públicos de agua potable, alcantarillado, depuración de aguas residuales, manejo de desechos sólidos, actividades de saneamiento ambiental y aquellos que establezca la ley (Asamblea Nacional de la República del Ecuador, 2008, p.130).

En cuanto a normativas que rigen el Ecuador se encuentra el Código Orgánico de Organización Territorial, Autonomía y Descentralización (COOTAD) en el artículo 137 establece que:

[...] Las competencias de prestación de servicios públicos de alcantarillado, depuración de aguas residuales, manejo de desechos sólidos, y actividades de saneamiento ambiental, en todas sus fases, las ejecutarán los gobiernos autónomos descentralizados municipales con sus respectivas normativas. Cuando estos servicios se presten en las parroquias rurales se deberá coordinar con los gobiernos autónomos descentralizados parroquiales rurales. (República del Ecuador, 2019, p.78)

De igual forma en el Código Orgánico Ambiental en el artículo 27 se menciona que:

En el marco de sus competencias ambientales exclusivas y concurrentes corresponde a los Gobiernos Autónomos Descentralizados Metropolitanos y Municipales el ejercicio de las siguientes facultades, en concordancia con las políticas y normas emitidas por los Gobiernos Autónomos Provinciales y la Autoridad Ambiental Nacional:

[...] 6. Elaborar planes, programas y proyectos para los sistemas de recolección, transporte, tratamiento y disposición final de residuos o desechos sólidos;

7. Generar normas y procedimientos para la gestión integral de los residuos y desechos para prevenirlos, aprovecharlos o eliminarlos, según corresponda. (República del Ecuador, 2017, p.19)

En cuanto a la recolección y transporte de desechos sólidos no peligrosos en el parágrafo IV de la Sección I del Texto Unificado de Legislación Secundaria de Medio Ambiente se dispone que:

Art. 66 De la recolección.- *Es responsabilidad de los Gobiernos Autónomos Descentralizados Municipales la recolección de los residuos y/o desechos sólidos no peligrosos tomando en cuenta los siguientes parámetros:*

- a) La recolección de los residuos sólidos y/o desechos no peligrosos, se realizará mediante los siguientes mecanismos: recolección manual, semi mecanizada y mecanizada.*
- b) La recolección de los residuos sólidos no peligrosos se realizará mediante las siguientes metodologías: de esquina, de acera, intra domiciliario, de contenedores, y las que establezca la autoridad ambiental para el efecto.*
- c) Establecer el servicio de recolección de residuos y/o desechos sólidos no peligrosos de tal forma que éstos no alteren o propicien condiciones adversas en la salud de las personas o contaminen el ambiente.*
- d) Durante el proceso de recolección, los operarios del servicio deberán proceder la totalidad de los residuos y/o desechos sólidos no peligrosos, evitando dejar residuos y lixiviados esparcidos en la vía pública.*

- e) *Establecer el servicio de barrido de residuos y/o desechos sólidos no peligrosos de tal forma que éstos no alteren o propicien condiciones adversas en la salud de las personas o contaminen el ambiente.*

Art.67 Del transporte.- *El traslado de los residuos y/o desechos sólidos desde el lugar de su generación hasta un centro de acopio y/o transferencia deberá contemplar procedimientos que cumplan con lo siguiente:*

- a) *Los equipos de transporte y recolección de residuos y/o desechos sólidos no peligrosos deben ser apropiados al medio y para la actividad.*
- b) *Evitar el derrame de los mismos durante el transporte hasta colocarlos en el centro de acopio y/o transferencia.*
- c) *Limpieza, desinfección y mantenimiento de los recipientes, vehículos de recolección y demás implementos utilizados en el transporte.*
- d) *Destinar únicamente residuos no peligrosos asimilables a domésticos al sistema de recolección local.*
- e) *El transporte de desechos peligrosos estará sujeto a lo dispuesto en la normativa correspondiente.* (República del Ecuador, 2015, p.21)

2.2.2 Sistema inteligente para la gestión de recolección de residuos sólidos urbanos

Los sistemas inteligentes han mejorado el proceso de recolección y reciclaje de desechos sólidos urbanos al realizar procesos veloces, de manera autónoma y optimizando recursos. Desde incluir máquinas que aplican Reverse vending que consisten en insertar botellas vacías en una máquina a cambio de una cantidad monetaria, hasta la generación de contenedores Smart como es el caso de la empresa Compology la cual ha creado sensores que detectan los niveles de basura y notifican a los camiones que se debe realizar la recolección. (Compology, 2022)

2.2.2.1 Internet de las cosas

Con el crecimiento agigantado de internet han surgido nuevas tecnologías, servicios y plataformas dando vida a un fenómeno conocido como “Internet de las cosas” o IOT por su nombre en inglés “Internet of Things”, es una red de objetos físicos conectados a internet, que se comunican, intercambian información entre sí y procesan dicha información para actuar autónomamente sin la intervención del humano. (Quiñonez, 2019, p.20)

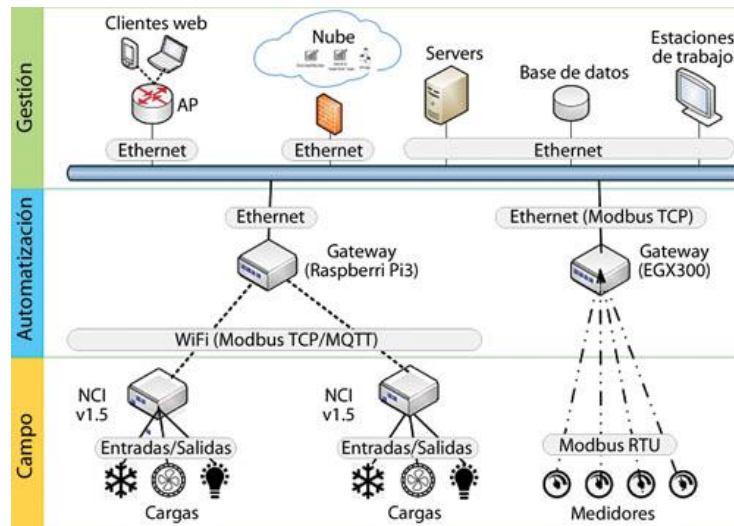


Ilustración 2-1: Arquitectura del SGEN con dispositivos IoT

Fuente: (Nicolau et al., 2019)

Este fenómeno supone la evolución del internet desde una red de ordenadores interconectados a objetos, es por esta razón que muchos consideran que el IOT es un agente importante en la cuarta revolución industrial, la industria 4.0, junto con la inteligencia artificial, la robótica, la impresión 3D y 4D, la nanotecnología, la biotecnología o la ciencia de materiales, nombrando algunos agentes fundamentales.

El internet de las cosas se está utilizando para el desarrollo de accesorios inteligentes tales como relojes y pulseras, así mismo está implementada en electrodomésticos como frigoríficos, aspiradoras, televisores, videoconsolas, automóviles, en elementos de viviendas como cerraduras, termostatos, bombillas, cámaras de seguridad, controles de acceso, incluso se encuentran en grandes infraestructuras como puentes, autopistas o ciudades, abriendo así la puerta a la interacción máquina-máquina (M2M). También se está aplicando en sistemas interconectados con animales, como sucede en algunas industrias ganaderas o en determinados programas protectores de la biodiversidad. (Moisés, 2020, pp.19-20)

2.2.2.2 Visión artificial

La visión artificial es también conocida como visión por computador, algunos autores la definen como la ciencia de programar un computador para interpretar y procesar imágenes o videos (Culjak et al, 2012, pp.1725-1730).

La visión artificial consiste en transformar los datos obtenidos de una cámara fija o de video en una decisión o una representación, todas las transformaciones se realizan para que el microcontrolador pueda tomar decisiones y lograr un objetivo (Bradski y Kaehler, 2009, p.100).

Los requerimientos para implementar visión artificial son:

- Un lenguaje de programación para crear algoritmos.
- Una biblioteca de visión artificial específica para trabajar con datos y algoritmos.
- Un ordenador capaz de procesar los datos.
- Una cámara para procesar los datos de entrada.
- Una conexión a Internet para compartir y descargar datos.
- Un entorno de desarrollo para programar, como un IDE o editor de código.

2.2.2.2.1 *Open CV*

Es una biblioteca de código abierto para procesamiento de imágenes y visión artificial, es una de las bibliotecas más populares y ampliamente utilizadas para aplicaciones de visión por computadora y análisis de imágenes. Proporciona una amplia variedad de funciones y herramientas que pueden ser utilizadas para procesar imágenes y videos en tiempo real, incluyendo detección de objetos, seguimiento de objetos, reconocimiento de patrones, análisis de movimiento, calibración de cámaras, y mucho más.

OpenCV fue inicialmente desarrollado por Intel en 1999, pero desde entonces ha sido mantenido y mejorado por una amplia comunidad de desarrolladores. La biblioteca es compatible con varios lenguajes de programación, incluyendo C++, Python, Java y MATLAB, además se puede utilizar en sistemas operativos como Windows, Linux, Android e iOS. (Igal y Medrano, 2008, p.7)

2.2.2.3 *Sistema interfaz hombre máquina*

También denominada HMI (Human Machine Interface) es un sistema que consiste en presentar la información necesaria a un operador (humano), el cual puede tomar las decisiones pertinentes del proceso desde un ordenador. Son necesarios dos medios para el desarrollo de una interfaz HMI: entrada, permite al usuario controlar el sistema; producto, reproduce las órdenes que el operario haya asignado. (Grueso, 2008, pp.19-20)

2.2.2.3.1 *Adafruit*



Ilustración 2-2: Logo de Adafruit Industries

Fuente: (Adafruit Industries, 2023)

Adafruit es una herramienta de desarrollo de código abierto que permite a los usuarios crear interfaces gráficas para Arduino, Raspberry Pi, ESP y otras plataformas de placas de desarrollo para controlar dispositivos como sensores, actuadores y pantallas táctiles. La plataforma administra los datos a través de la nube, ofrece una amplia variedad de librerías y herramientas de desarrollo que ayudan a los usuarios a crear aplicaciones interactivas con animaciones y contenido multimedia en poco tiempo. (Mubashir y Mahnoor, 2020, pp.335-341)

La velocidad de respuesta de Adafruit para interfaz gráfica depende del dispositivo y la plataforma que se esté utilizando, por lo general se puede tener tiempos de respuesta entre 10 y 30 s para dispositivos como Arduino y Raspberry Pi, además como la plataforma proporciona funciones de depuración, permite identificar rápidamente los errores y mejorar el rendimiento de su aplicación.

2.2.2.3.2 *ThingSpeak*



Ilustración 2-3: Logo ThingSpeak

Fuente: (MathWorks, 2023)

Es una plataforma de IoT para diseñar HMI, permite recopilar, analizar y compartir datos de dispositivos electrónicos conectados a la nube, dispone de una API abierta y modificable, es integrable con Arduino, Raspberry, aplicaciones móviles, redes sociales, entre otros y se caracteriza por ser de simple programación. El tiempo de respuesta del programa varía según la velocidad de conexión a Internet y del dispositivo que se esté utilizando, aunque está diseñada para responder al instante a la recopilación y almacenamiento de datos, en general tiene un tiempo de respuesta entre 1 y 5 segundos. (Gómez et al, 2014, p.3)

2.2.2.3.3 *Blink IoT*

Blynk es una plataforma de IoT para el diseño de aplicaciones de control remoto inalámbrico de dispositivos conectados a la web, tiene una interfaz gráfica intuitiva, fácil de usar y con herramientas para crear interfaces personalizadas. (Tomalá y Santana, 2022, p.32)



Ilustración 2-4: Logo Blink IoT

Fuente: (Blynk International, 2023)

Sus principales características son:

- Una sencilla interfaz de usuario, por lo que se pueden crear proyectos sin la necesidad de un conocimiento profundo en programación.
- Tiempo de respuesta casi inmediata.
- Una API para conectar el servidor de Blynk con dispositivos externos, esto permite un control remoto de los dispositivos conectados.
- Recopila y almacena datos.
- Cuenta con una variedad de widgets prediseñados para controlar los dispositivos, como interruptores, botones, sliders.
- Una comunidad de desarrolladores de Blynk donde los usuarios pueden compartir sus proyectos y recibir ayuda.

2.2.2.3.4 Ubidots



Ilustración 2-5: Logo de Ubidots

Fuente: (Ubidots, 2023)

Ubidots es una plataforma de Internet de las cosas (IoT) que permite supervisar y administrar dispositivos y datos e interactuar con ellos. Con la plataforma se pueden crear aplicaciones para rastrear y administrar dispositivos y datos desde una sola consola, cuenta con herramientas para optimizar la gestión de datos como paneles, alertas, widgets, entre otros, además integra múltiples plataformas de terceros lo que le permite monitorear, analizar, comprender y actuar sobre los datos en un tiempo considerable. Ubidots cuenta con una API que integra sus aplicaciones en la plataforma, por lo que los usuarios pueden acceder a sus datos desde aplicaciones móviles o aplicaciones web. La plataforma tiene una versión gratuita de prueba para aquellos que desean probar la plataforma antes de comprarla. (Revelo y Henao, 2017, p.288)

2.2.3 Componentes electrónicos

Los componentes electrónicos son dispositivos que se utilizan para la construcción de circuitos electrónicos, entre los componentes más comunes se encuentran las resistencias, condensadores, transistores, sensores, relés y muchos otros, empleados en una amplia variedad de aplicaciones, desde la industria aeroespacial hasta los dispositivos domésticos. (Alloza, 2021, p.30)

2.2.3.1 *Raspberry Pi*

Es un ordenador de una sola placa o placa simple (single Board Computer, SBC), de tamaño reducido similar a una tarjeta de crédito, fue desarrollado en Reino Unido en el año 2006 por Raspberry Pi Foundation con la misión de poner a disposición el poder de la informática y la creación digital a los jóvenes para que desarrollen su potencial, la placa es de bajo costo, el acceso a sus librerías es libre y es compatible con múltiples sistemas operativos. (Raspberry Pi Ltd, 2022).

De manera general el microcontrolador cuenta con un procesador, una memoria RAM, una unidad de procesamiento gráfico (GPU), procesador broadcam, puertos USB, HDMI, ethernet y diversos pines de entrada y salida. (Escalante y Vargas, 2019)

2.2.3.2 *Arduino*

Es una plataforma electrónica de código abierto basado en un software y hardware de fácil manejo, los microcontroladores de Arduino son capaces de leer entradas analógicas y digitales, y dependiendo de las instrucciones generar salidas que cumplan con las necesidades. Para programar las instrucciones a la placa de Arduino se utiliza el lenguaje de programación (basado en Wiring), y el Software Arduino (IDE), basado en Processing. Arduino fue creado en 2005 en el Instituto de Diseño de Interacción Ivrea como una herramienta para la creación de prototipos dirigida a estudiantes sin conocimientos en electrónica y programación. A través de los años Arduino ha ido adaptándose a las nuevas necesidades y desafíos que aparecen con los avances tecnológicos por lo que actualmente ofrece productos para aplicaciones como IoT, wearables, impresión 3D y entornos integrados. (Arduino, 2022)

2.2.3.3 *GPS*

GPS (Global Positioning System) es un sistema de navegación por satélite que puede utilizarse para determinar la ubicación exacta de un objeto en la Tierra, este sistema consta de una red de satélites y estaciones colocadas estratégicamente en todo el mundo, los satélites emiten señales que son recibidas por los receptores GPS que procesan dichas señales y calculan la posición en el espacio de determinado punto. Se utilizan para proporcionar direcciones de viaje, ubicaciones de referencia, tiempo real y otros datos de ubicación. (Pozo et al., 2000, pp.2-3)

2.2.4 *Actuadores eléctricos*

Los actuadores eléctricos son dispositivos de control de posición que convierten una señal de control eléctrica en movimiento mecánico, se usan en una variedad de aplicaciones incluyendo

sistemas de control de procesos, robots, maquinarias, y muchos otros. Los actuadores eléctricos se clasifican en dos grandes grupos: lineales y rotativos. Los actuadores lineales convierten la energía eléctrica en movimiento lineal a través de una vía de desplazamiento, mientras que los actuadores rotativos convierten la energía eléctrica en un movimiento rotacional como es el caso de los motores. (Corona et al., 2014, p.280)

2.2.4.1 *Servomotor*

Son motores que trabajan en un sistema cerrado que por lo general está conformado por: un dispositivo de control externo, un controlador electrónico (servo drive), un sistema de realimentación (encoder) y el servomotor. Se caracterizan porque permite el control de la posición del eje en un momento determinado (Fraile, 2021, p.2).



Ilustración 2-6: Servomotor

Fuente: (Ligo, 2020)

2.2.5 *Comunicación inalámbrica*

“En un sentido amplio y general, se entiende por comunicaciones inalámbricas aquellas comunicaciones entre dispositivos (móviles o no) o entre personas que intercambian información utilizando el espectro electromagnético” (Prieto, 2011, p.10).

2.2.5.1 *Red inalámbrica Wifi*

Una red inalámbrica es aquella que conecta dispositivos mediante ondas de radio, sin necesidad de emplear una red cableada. Las redes inalámbricas de área local (Wireless Local Area Networks, WLAN) que son comercializadas con el nombre de Wi-Fi son un tipo de red inalámbrica capaz de proporcionar acceso con un rango de hasta 100 metros, están normadas bajo el estándar 802.11 del IEEE y pertenecen a la categoría de mediano alcance. (Salazar, 2016, p.13)



Ilustración 2-7: Esquema de una WLAN

Fuente: (Salazar, 2016)

2.2.5.2 *Web server*

Un web server es un software y hardware que utiliza un protocolo de transferencia de hipertextos (HTTP) los cuales responden a las necesidades de los usuarios por medio de la World Wide Web. La función principal de un servidor web es aquel que, mediante el almacenamiento, el procesamiento muestra el contenido del sitio web a los usuarios, otros protocolos que admite un servidor web son SMTP (Protocolo simple de transferencia de correo) y FTP (Protocolo de transferencia de archivos), que se utilizan para correo electrónico, transferencia de archivos y almacenamiento. Para el hardware de un servidor web es necesario que tenga conexión a internet para intercambiar información con otros dispositivos conectados, la información adquirida se mostrará y será controlada por el usuario en el software. El servidor web sigue un modelo Cliente/Servidor. (Gillis, 2020)

2.2.6 *Software*

El uso de software es una excelente forma de aumentar la productividad y mejorar la eficiencia, pues permite realizar tareas de forma rápida, organizar proyectos en el menor tiempo posible y proporciona herramientas para una correcta toma de decisiones, existen softwares libres para el diseño CAD y también para el diseño electrónico los cuales facilitan el diseño de cualquier prototipo. (Adell y Bernabé, 2007, p.19)

2.2.6.1 *SolidWorks*

Es un software CAD (diseño asistido por computadora) que permite efectuar el proceso completo de diseño mecánico, desde la croquización de ideas hasta el desarrollo de planos detallados en 2D necesarios para su fabricación (Vidal y Maroño, 2015, p.13).



Ilustración 2-8: Logotipo SolidWorks

Fuente: (Dassault Systemes, 2022)

2.2.6.2 *Proteus*

Proteus es un software de diseño, simulación y verificación de circuitos electrónicos que proporciona un entorno gráfico de uso sencillo y cuenta con herramientas que permiten al usuario trabajar con circuitos electrónicos complejos. Además, el software incluye una amplia gama de librerías de componentes electrónicos que lo vuelven sumamente didáctico. (Barrón, 2000, p.1)

2.2.7 *Lenguajes de programación*

Los lenguajes de programación son un conjunto de reglas, símbolos y lógica usados para emitir ordenes creando programas informáticos, se emplean para dar instrucciones a una computadora para que realice una variedad de tareas. Existen muchos lenguajes de programación cada uno con sus propias características y estructura, algunos son más adecuados para la programación estructurada, mientras que otros son mejores para la programación orientada a objetos, los lenguajes de programación más comunes son C/C++, Java, Python, Visual Basic y JavaScript. (Desongles et al., 2006, p.312)

CAPÍTULO III

3 MARCO METODOLÓGICO

A fin de materializar los objetivos planteados se establece un conjunto de pasos ordenados a seguir que se muestran de manera detallada a continuación.

En primer lugar, se definen los requerimientos del sistema que consiste en especificar las características necesarias para que el prototipo satisfaga los objetivos planteados, los requerimientos del sistema se dividen en estructurales, del hardware y software.

Posteriormente, se diseña el sistema para lo cual se realiza una comparación de los componentes electrónicos a utilizar, de las plataformas IoT más relevantes para desarrollar interfaces gráficas remotas y de los lenguajes de programación más populares para visión artificial; de cada grupo se selecciona la opción que cumpla todos los requerimientos del sistema definidos previamente y que brinde ventajas frente a las otras opciones analizadas. Luego, se diseñan las conexiones del hardware en un software de diseño electrónico y se lleva a cabo la programación en el lenguaje de programación seleccionado.

Después, se desarrolla el prototipo por lo que se ensamblan todos los componentes electrónicos en la estructura y se enlaza la parte estructural del sistema con la interfaz gráfica, es decir este paso comprende todo el proceso de construcción del prototipo.

Finalmente, se somete al prototipo culminado a varias pruebas para obtener la eficiencia del algoritmo de visión artificial, la variación de lectura de las coordenadas de ubicación del módulo GPS, la validación del punto más cercano, el tiempo de latencia del sistema de bloqueo de la puerta y del sistema de alerta.

3.1 Consolidación general del prototipo

En la ilustración 3-1 se describe de forma gráfica el funcionamiento esperado del sistema inteligente para la gestión de recolección de desechos sólidos urbanos basado en comunicación inalámbrica, el sistema posee dos entornos:

- **Local:** está constituido por el contenedor de basura y los componentes eléctricos y electrónicos del prototipo.
- **Remoto:** conformado por una interfaz gráfica a la que se puede acceder desde cualquier lugar al ingresar a una plataforma IoT alojada en la nube. La interfaz gráfica muestra información del prototipo y le permite al usuario bloquear y desbloquear el acceso a este.

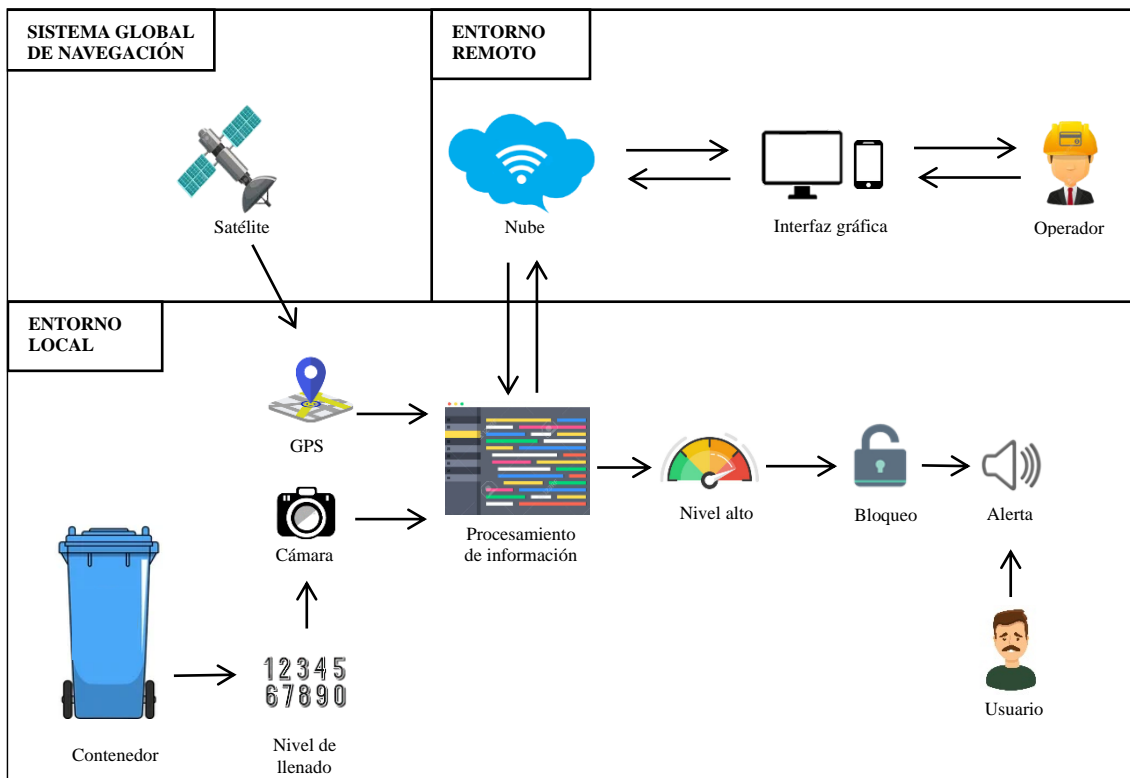


Ilustración 3-1: Consolidación general del prototipo

Realizado por: Guamán Lessly, López Kevin, 2023

El proceso inicia con la recepción de información del GPS sobre la ubicación del prototipo y la adquisición de imágenes del interior del basurero, las cuales son procesadas mediante un modelo de visión artificial basado en el reconocimiento óptico de caracteres (OCR) que extrae el texto de las imágenes capturadas, las compara e identifica el número menor que corresponde al nivel de llenado del contenedor. Esta información es enviada a la nube y es expuesta en una interfaz gráfica remota que le permite al usuario monitorear el estado del prototipo y controlar el acceso al mismo. Cuando se alcance el nivel de llenado máximo, se ejecutan las siguientes tareas:

- Bloqueo automático de la puerta del contenedor.
- Activación de una alerta en caso de intento de acceso.

3.2 Definición de requerimientos

El prototipo del sistema de gestión de recolección de basura debe poseer características indispensables para su funcionamiento denominados requerimientos del sistema, los cuales están constituidos de los requerimientos estructurales, del hardware y software.

- Diseñar en el software de diseño CAD 3D SolidWorks el prototipo de la estructura metálica del contenedor de basura en un tamaño y material óptimo para su manipulación, transporte y resistencia a las condiciones medioambientales a las que estará sometida.
- Construir la estructura metálica basado en el diseño realizado en el software CAD.
- Adoptar una escala de medición para identificar el nivel de llenado de los contenedores de basura y acondicionar el ambiente para la obtención de imágenes para visión artificial.
- Obtener la ubicación en tiempo real de los basureros.
- Codificar un algoritmo de visión artificial en un lenguaje de programación compacto y de código abierto.
- Crear una interfaz gráfica remota que monitoree el estado de los contenedores de basura e integre recursos gráficos que favorezcan la interpretación de la información expuesta.
- Generar un modelo de red inalámbrica que considere a los basureros como un nodo que se conectan a la interfaz gráfica remota mediante WiFi.
- Seleccionar hardware que permita la comunicación inalámbricamente con el sistema, que gestione señales de control, sea capaz de soportar un algoritmo de procesamiento de imágenes y se conecte a la nube.
- Implementar un sistema que bloquee la puerta del contenedor de basura cuando se alcance el nivel de llenado máximo y genere una alerta en caso de intento de acceso.

3.3 Arquitectura general del prototipo

Este apartado involucra un conjunto de bloques que relacionan el accionamiento local y remoto de los componentes del hardware y software para obtener el funcionamiento planteado en la consolidación general del prototipo. En la ilustración 3-2 se muestra la arquitectura general del prototipo representado en un diagrama de bloques.

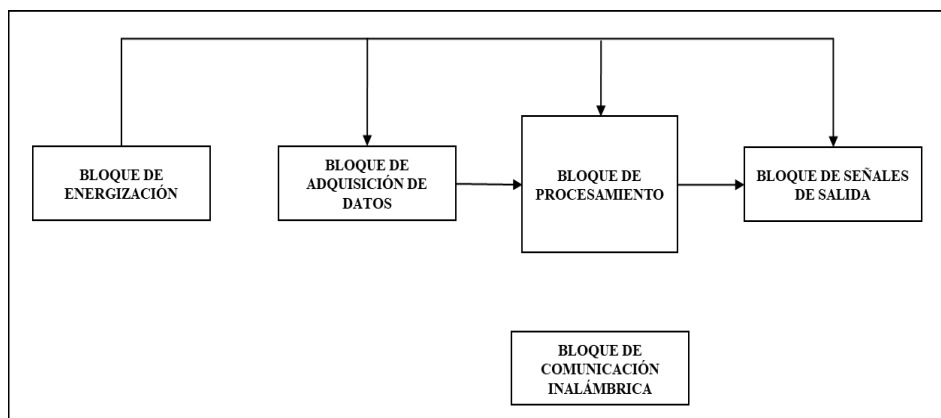


Ilustración 3-2: Arquitectura general del prototipo

Realizado por: Guamán Lessly, López Kevin, 2023

A continuación, se describe cada bloque de la arquitectura del prototipo:

- **Bloque de energización:** este bloque hace referencia a la fuente de suministro de energía que requieren los componentes eléctricos y electrónicos del prototipo para su funcionamiento.
- **Bloque de adquisición de datos:** corresponde a la adquisición de información proveniente de: el sensor de adquisición de imágenes, el dispositivo que recepta el posicionamiento global del prototipo y el sensor que detecta cuando se intenta acceder a un contenedor bloqueado. Cabe recalcar que, al existir una comunicación bidireccional con la interfaz gráfica remota, también se reciben datos provenientes de esta.
- **Bloque de procesamiento:** aquí se gestionan las señales receptadas por el sensor de adquisición de imágenes y mediante la ejecución de un algoritmo de visión artificial se determina el nivel de llenado del contenedor de basura. De igual manera gestiona las señales provenientes del dispositivo de posicionamiento global y del sensor ubicado en la puerta, del análisis de estas señales se obtiene la ubicación del prototipo y se generan señales para el accionamiento del actuador que bloquea el basurero y para la generación de una alarma. Además, este bloque contiene las sentencias de programación de una interfaz gráfica remota que monitorea las variables del prototipo correspondientes al nivel de llenado y ubicación, determina cual es el contenedor lleno más cercano y le permite al usuario bloquear o desbloquear el acceso al contenedor de basura.
- **Bloque de señales de salida:** luego del procesamiento de información de los componentes del hardware y de la interfaz gráfica remota se generan señales para controlar el acceso al contenedor de basura y para la activación de una alarma.
- **Bloque de comunicación inalámbrica:** Este bloque es el encargado de recibir y enviar datos y señales del prototipo mediante el uso de una red local de WiFi. Para la comunicación entre el contenedor de basura y la interfaz gráfica remota se empleó el Protocolo de Transferencia de Hipertexto (HTTP) que permite el intercambio de información a través de internet.

3.4 Diseño CAD

Para el diseño del prototipo se utilizó el software SolidWorks que ofrece diferentes herramientas y bibliotecas para crear un prototipo que satisfaga los requerimientos establecidos.

3.4.1 *Diseño de la estructura*

Para la estructura se consideró utilizar un ángulo doblado de 20x2 mm, porque tiene una gran resistencia, la unión de las partes de la estructura es por medio de soldadura, utilizando el electrodo 6013.

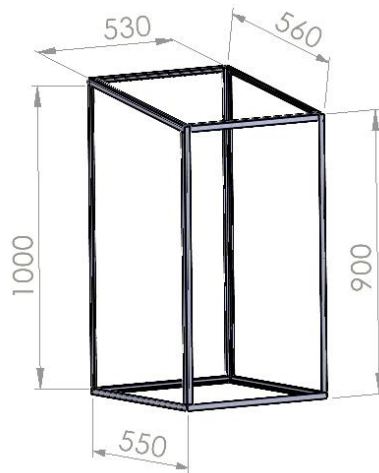


Ilustración 3-3: Dimensiones de la estructura

Realizado por: Guamán Lessly, López Kevin, 2023

3.4.2 *Diseño de la cubierta*

La cubierta se diseñó en tol galvanizada de 0.9 mm de espesor, material seleccionado por sus propiedades de resistencia a la corrosión, este elemento va unida a la estructura mediante soldadura.

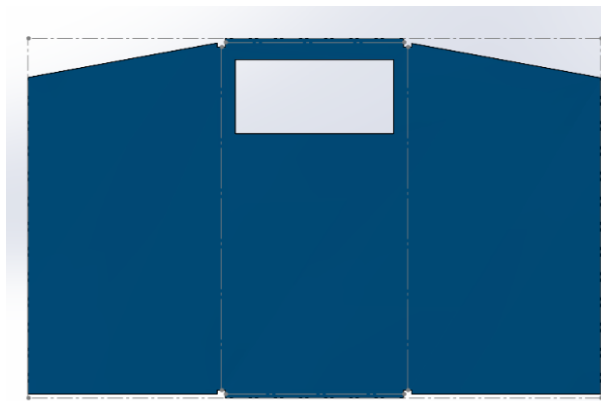


Ilustración 3-4: Diseño CAD de la cubierta

Realizado por: Guamán Lessly, López Kevin, 2023

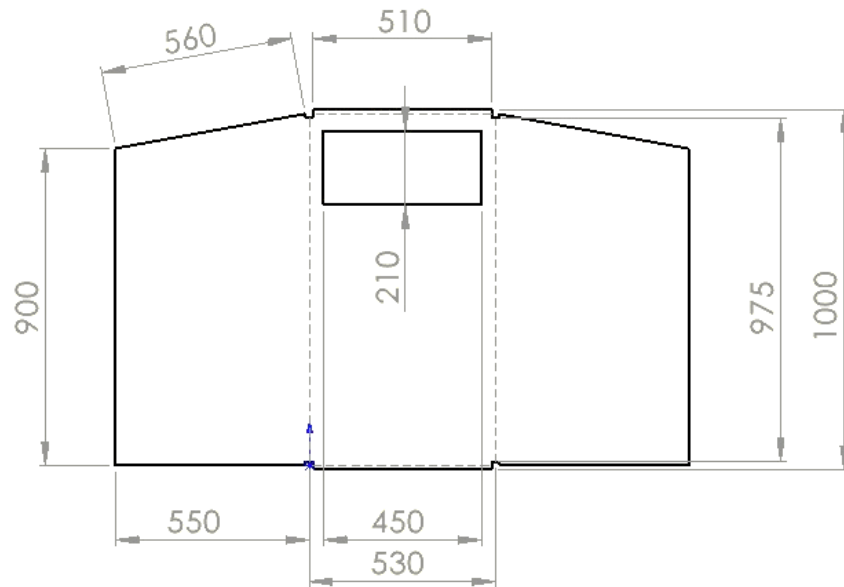


Ilustración 3-5: Dimensiones de la cubierta

Realizado por: Guamán Lessly, López Kevin, 2023

3.4.3 *Diseño de puerta trasera*

La estructura de la puerta trasera se diseñó en platina de 25x3 mm que está unida por medio de soldadura, y la puerta es de tol galvanizado de 0,9 mm que se ensambla a la estructura con remaches.

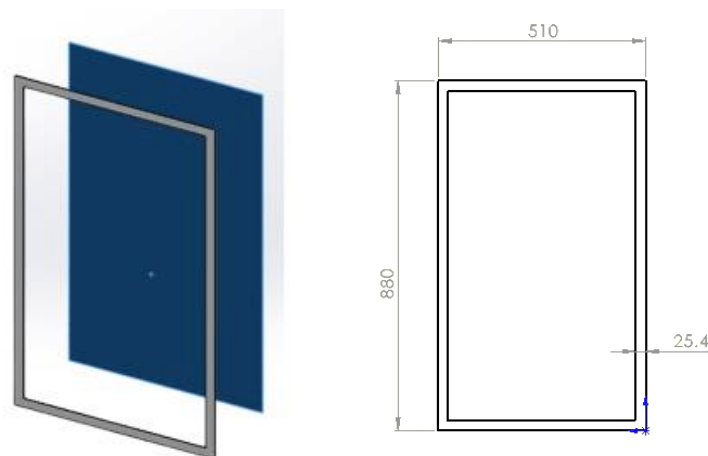


Ilustración 3-6: Diseño CAD y dimensiones de la puerta trasera

Realizado por: Guamán Lessly, López Kevin, 2023

3.4.4 *Diseño del contenedor de basura*

Para el diseño del contenedor se ensamblan todos los componentes formando una estructura sólida, a la que se le agregó pegatinas que ayudan a prevenir la corrosión y tiene un diseño atractivo que sugiere que se trata de un dispositivo inteligente.



Ilustración 3-7: Diseño CAD del contenedor de basura

Realizado por: Guamán Lessly, López Kevin, 2023

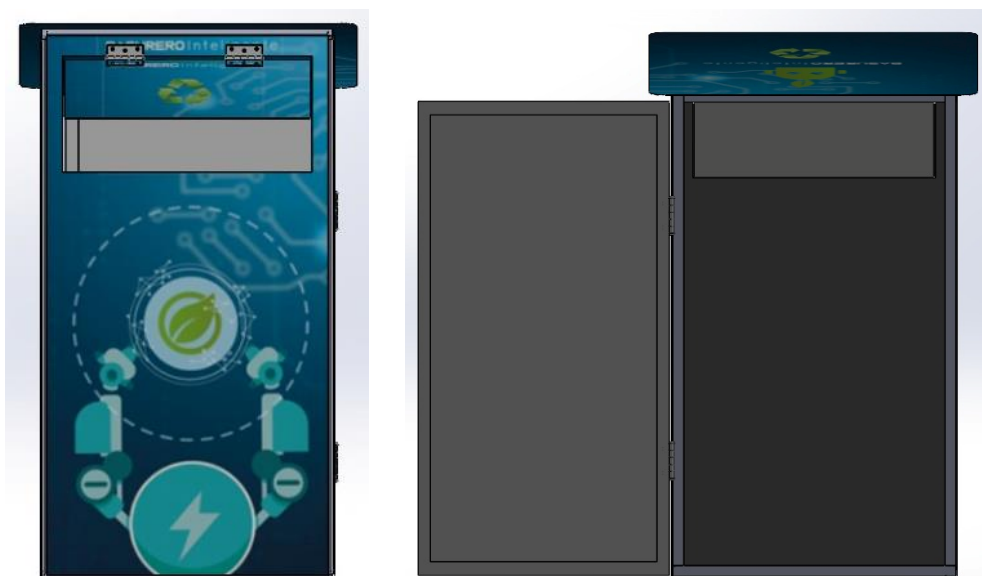

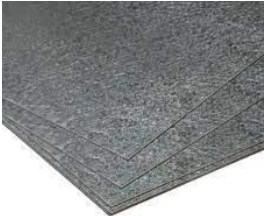






Ilustración 3-8: Vista frontal y posterior del contenedor de basura

Realizado por: Guamán Lessly, López Kevin, 2023

Tabla 3-1: Elementos utilizados en el contenedor de basura

Definición	Elemento	Cantidad
Angulo doblado (20x2) mm		9 m
Tol galvanizado (0.9) mm		2,60 m ²
Platina (25x3) mm		2,90 m
Bisagras		4 u
Remaches		55 u
Electrodos (6013)		7 u

Realizado por: Guamán Lessly, López Kevin, 2023

3.5 Hardware



En este apartado se realiza la comparación y selección de todos los componentes físicos necesarios para desarrollar el prototipo.

3.5.1 Hardware para la visión artificial

Uno de los componentes del sistema de visión artificial es un sensor que capture las imágenes del nivel de llenado de los contenedores de basura, en la tabla 3-2 se muestran las características de

una cámara web y una cámara nocturna para realizar una comparación y posterior selección de la mejor opción para el prototipo.

Tabla 3-2: Comparación de cámaras

	CÁMARA WEB	CÁMARA NOCTURNA
Imagen		
Sensor de imagen	2 MP CMOS	OV5647
Consumo	Max. 1.2 W	96 mA
Tasa máxima de transferencia de imagen	1920 x 1080 @ 5 fps 1280 x 960 @ 5 fps 1280 x 720 @ 5 fps 640 x 480 @ 30 fps 640 x 360 @ 30 fps	2592x1944: 15 fps 1080 p: 30 fps 960p: 45 fps 720p: 60 fps 640x480:90 fps
Resolución	1080p	1080p
Lente	3.6 mm	1/4"
Campo de visión	Horizontal FOV: 86° Vertical FOV: 55° Diagonal FOV: 95°	24°
Montura de Lente	M8	No
Ajuste de ángulo	Panorámica: 360° Inclinación: -15° a 15° Rotación: 0°	70 °
Salida de video	USB 2.0	Datos RGB RAW de 8/10 bits
Sistema Operativo	Windows 7/8/10/11 Android 8.1 and above version Linux 4.14 and above version macOS 10.12 and above version Chrome OS	Windows Linux macOS
Condiciones de operación	Temperatura: -10°C a 45°C Humedad: 90%	-30°C a 70°C
Fuente de alimentación	DC 5V ± 5% (USB)	3.3V
Dimensión	80.16 mm × 48.6 mm × 42.18 mm	25mm x 24mm
Peso	89 g	3 g
Precio	\$ 10.00	\$ 25.00

Fuente: (Hikvision Digital Technology, 2020); (Unit Electronics, 2022)





Realizado por: Guamán Lessly, López Kevin, 2023

Las imágenes capturadas por el sistema deben ser de una calidad capaz de reconocer los patrones utilizados para identificar el nivel de llenado, pero no excesivamente alta, ya que ralentizaría el procesamiento de imágenes, tomando esto en consideración se seleccionó la cámara web, que tiene un sensor de 2 megapíxeles, cuenta con montura la cual permite manipular el dispositivo según las necesidades, posee un ángulo de visión bastante amplio y su costo es bajo. Adicionalmente, para la obtención de imágenes fue necesario acondicionar el ambiente con un foco led para interiores de 9W.

3.5.2 Hardware para la comunicación inalámbrica

Un requerimiento indispensable del prototipo es conocer la ubicación de los contenedores de basura para brindar el servicio de recolección de desechos sólidos, por lo que se requiere la utilización de un módulo GPS que proporcione las coordenadas de los basureros, dependiendo del módulo la velocidad y la precisión de los datos variarán, en la tabla 3-3 se analizan varios módulos GPS disponibles en el mercado y se selecciona el más adecuado.

Tabla 3-3: Comparación de módulos GPS.

	NEO 6M	BEITIAN GPS BN-220 DUAL GLONASS	QUECTEL L80	EM-406
Imagen				
Alimentación	2.7-3.6 VDC	3.5-5.5 VDC	3.0 - 4.3 VDC	4.5 – 6.5 VDC
Corriente	47mA	50mA	25mA	44mA
Frecuencia receptora	1575.42 Mhz	1575.42 Mhz	1575.42 Mhz	1575,42 MHz
Conjunto de chips	-	u-blox M8030-KT	-	SiRF StarIII
Canales	50	72	66	20
Sensibilidad (Inicio, reaquisición y tracking)	-156 dBm -160 dBm -161 dBm	-156 dBm -160 dBm -167 dBm	-148 dBm -160 dBm -165 dBm	-159 dBm
Tasa de baudios	600 bps	4800bps ~ 115200bps	9600 bps	4800 bps
Pines Sensor	24		9	6
Presentación	Módulo GPS Neo 6M	Beitian GPS BN-220 Dual Glonass	Módulo GPS3 CLICK Shield M2M	Solo
Pines Modulo	4	4	16 y 2	6
Velocidad	515 m/s	516 m/s	516 m/s	515 m/s

Protocolo de datos	NMEA	NMEA 0183 UBX	NMEA 0183 PMTK	NMEA 0183
Temperatura de funcionamiento	-40 °C a +85 °C	-40 °C a +85 °C	-40 °C a +85 °C	-40 °C a +85 °C
Precisión de posicionamiento	2,5 m	1,5 m	0,5 m	10 m
Tiempo de inicio	48s	26s	35s	38s
Dimensiones sensor	6 x 12.2 mm	22 X 20 X 6 mm	16 x 16 x 6.45 mm	30 x 30 x 10.5 mm
Dimensiones antena	22 x 22 mm	10 x 15 mm	15 x 15 x 4 mm	25.20 x 25.20 mm
Disponibilidad en el país	Sí	No	No	No
Precio	\$15	\$50	\$45,3	\$12

Fuente: (Quectel, 2022)

Realizado por: Guamán Lessly, López Kevin, 2023

Posterior al análisis de las características de cada módulo GPS se seleccionó el módulo NEO 6 M porque tienen una gran precisión de posicionamiento de 2,5 m con relación a su bajo costo de \$15, además que está disponible en el mercado local a diferencia de las otras opciones que no se encuentran disponibles en el país.

3.5.3 Hardware para el bloqueo de la puerta

El prototipo debe contar con un sistema de bloqueo en la puerta que se activa cuando el basurero alcanza su capacidad máxima, por tal razón se debe seleccionar un actuador que realice esta tarea. Según la fuente de alimentación, los actuadores comúnmente se clasifican en: neumáticos, hidráulicos y eléctricos. Tanto los actuadores neumáticos como hidráulicos requieren de una fuente de alimentación externa para funcionar, tienen una instalación compleja, y al ser capaces de generar una gran potencia son utilizados en aplicaciones de trabajos pesados. Por otro lado, los actuadores eléctricos se accionan con un motor, son poco ruidosos, de instalación y mantenimiento sencillo y se emplean frecuentemente en sistemas automatizados.

En base al análisis de los distintos tipos de actuadores, se concluye que los actuadores eléctricos son los apropiados para el prototipo, en la tabla 3-4 se muestran las características de los dispositivos más usados en proyectos electrónicos.

Tabla 3-4: Comparación de actuadores eléctricos

	SERVOMOTOR MG996R	MOTOR PASO A PASO NEMA 17	MICRO DC MOTOR MM28
Imagen			
Torque	9,4 Kgf(4,8V);11 Kgf (6V)	26Ncm	20 g-cm
Peso	55 g	230 g	42 g
Dimensiones	40,7 x 19,7 x 42,9 mm	42 x 42 mm	25 x 15 x 20 mm
Velocidad de funcionamiento	0,17 s/60° (4,8V); 0,14 s/60°(6V)	200 pasos/ revolución	9600 rpm
Controla el giro	Sí	Sí	No
Tensión de funcionamiento	4,8 V - 7,2 V	4,8 V	3-6 V
Corriente de funcionamiento	500 mA - 900mA(6V)	0,84 A	0,22 A
Precio	\$ 9.00	\$ 35.00	\$ 2.00

Fuente: (Infineon, 2016) ; (Electroniclab, 2022); (Multicomp, 2012)

Realizado por: Guamán Lessly, López Kevin, 2023


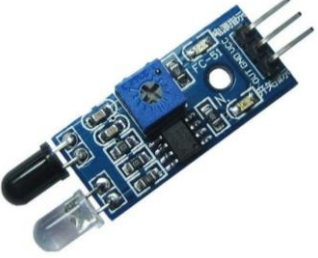
El micro motor DC MM28 presenta una ventaja importante en cuanto a su reducido tamaño y su bajo costo, sin embargo, no es posible controlar su ángulo de giro, siendo poco preciso en comparación a las otras opciones presentadas. De manera similar el motor paso a paso Nema 17 es de un tamaño ventajoso para el proyecto y de precio asequible, no obstante, necesita de Drivers para trabajar, por lo que la programación para obtener el ángulo de giro es extenuante. Por otro lado, el servomotor MG996R es de un tamaño compacto, bajo costo y permite controlar el ángulo de giro sin necesidad de Drivers. Con el análisis realizado se decidió seleccionar el servomotor MG996R debido a sus evidentes ventajas en costo, tamaño, fácil programación y por la gran cantidad de librerías disponibles en la web que sirven de soporte para el proceso.

3.5.4 Hardware para el sistema de alerta

El prototipo debe contar con un sistema que informe al usuario que el basurero alcanzó su capacidad máxima, por lo que al momento que el usuario intente depositar su basura se encontrará con la puerta bloqueada e instantáneamente se activará el sistema de alerta, para la activación de

este sistema se consideraron sensores, pues son fáciles de adquirir en el mercado y no tienen un costo elevado, los sensores analizados fueron el sensor ultrasónico HC - SR04 y el sensor infrarrojo fc-51 cuyas características se encuentran detalladas en la tabla 3-5.

Tabla 3-5: Comparación de sensores

	SENSOR ULTRASÓNICO HC-SR04	SENSOR INFRARROJO FC-51
Imagen		
Voltaje de Operación:	5V DC	3-5 V
Corriente de reposo:	< 2 mA	
Corriente de trabajo:	15 mA	43 mA (5V)
Rango de medición:	2cm a 450 cm	2-30 cm
Precisión:	± 3 mm	-
Angulo de apertura:	15°	35°
mide distancia	SI	NO
Frecuencia de ultrasonido:	40 KHz	-
Duración mínima del pulso de disparo TRIG(nivel TTL):	10 μs	-
Duración del pulso ECO de salida (nivel TTL):	100 - 25000 μs	-
Dimensiones:	45x20x15	31x15x7 mm
Tiempo mínimo de espera entre medidas:	20 ms	-

Fuente: (OSAKA ELECTRONICS, 2020); (ELECfreaks, 2020)

Realizado por: Guamán Lessly, López Kevin, 2023

El sensor ultrasónico emite una onda ultrasónica y en función del tiempo que tarda en regresar se determina la distancia a la que se encuentra cierto objeto por lo que es útil si se requiere precisión; por su parte el sensor infrarrojo emite energía infrarroja de manera constante y cuando existe la presencia de un obstáculo la detecta mediante la reflexión de luz que este ocasiona. En cuanto a su programación la diferencia entre el sensor ultrasónico y el infrarrojo es que el primero se basa en distancias, mientras que el infrarrojo no ya que el módulo regula esta magnitud.

Debido a que el sensor a seleccionar debe detectar que la puerta de un contenedor bloqueado se está abriendo, tarea de poca precisión, se seleccionó el sensor infrarrojo FC-51 que cumple la función requerida, es de fácil programación y sirve para activar el sistema de alarma.

3.5.5 *Hardware para alerta de audio*

Como el prototipo posee una alerta de audio es necesario contar con una bocina y un amplificador de audio, cuyas características se encuentran detalladas en la Tabla 3-6 y Tabla 3-7 respectivamente.



Ilustración 3-9: Parlante

Fuente: (C&D TecHNologia, 2023)

Tabla 3-6: Características del parlante

Referencia	BE3L4C2 SPSE7L2
Salida de audio	Dual
Potencia de salida	10W
Impedancia nominal	8 Ohm
Canales	2
Batería	No
Comunicación	Wireless+Wired
Tamaño	3 in

Fuente: (C&D TecHNologia, 2023)

Realizado por: Guamán Lessly, López Kevin, 2023

Se seleccionó el parlante BE3L4C2 SPSE7L2 que se muestra en la ilustración 3-9, porque al tener dos salidas de audio proporciona un sonido claro y adecuado para usar en exteriores. Además, al tener una impedancia nominal de 8 ohmios es compatible con la mayoría de los amplificadores y receptores de audio estándar.



Ilustración 3-10: Módulo amplificador de audio mono TDA2030A

Fuente: (Tecnopura, 2022)

Tabla 3-7: Características amplificador de audio mono TDA2030A

Referencia	TDA2030A
Voltaje de operación	6~12VDC
Potencia de salida	18W por 1 canal
Impedancia de salida	4~8 Ohm
Control de volumen	Incorporado
Disipador de calor	Incorporado
Peso	20 g
Dimensiones	3,2 × 2,7 × 2,2 cm

Fuente: (Tecnopura, 2022)

Realizado por: Guamán Lessly, López Kevin, 2023



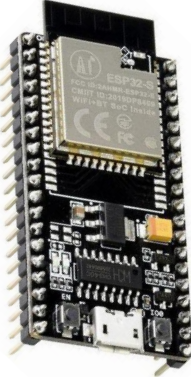
El módulo amplificador de audio mono TDA2030A de la ilustración 3-10 fue el seleccionado para el prototipo, considerando que:

- La potencia (aproximadamente 14W a 4 ohmios) es suficiente para manejar la potencia de salida del parlante (10W). Esto significa que el amplificador puede proporcionar suficiente energía para que el parlante funcione de manera óptima sin riesgo de daño por falta de potencia.
- Requiere conexiones estándar, por lo que no se requiere un diseño de circuito complicado, simplificando la tarea de construir un sistema de audio.

3.5.6 Hardware de procesamiento

A continuación, en la Tabla 3-8 se compara las tres tarjetas de desarrollo más reconocidas en la actualidad, para seleccionar la adecuada para el prototipo.

Tabla 3-8: Comparación de ordenadores

	ARDUINO MEGA 2560	RASPBERRY PI 4 Model B	ESP 32
Imagen			
Núcleo	-	Cortex-A72 (ARM v8) 64-bit	-
Nº de núcleos	-	4	2
Memoria (RAM)	8 KB	1/2/4/8GB LPDDR4	512 KB
Velocidad del reloj	16 MHz	1.5GHz	160 MHz
Conectividad red	No	RJ45 Ethernet 10/100/1000	No
Wireless	No	WiFi 802.11 b/g/n/ac Doble banda	WiFi 802.11b/g/n HT 40
		Bluetooth® 5.0 BLE	Bluetooth 4.2 y BLE
Sistema Operativo	Ninguno	Distribuciones Linux	Ninguno
Pines de E/S	70	40	34
Almacenamiento	256 KB	microSD hasta 32GB	4 MB
Salida Audio	No	Analógico por jack 3.5mm	No
Salida Video	No	2x microHDMI 4K Video analógico compartido por jack	No
Procesa imágenes	No	Sí	Sí
Periféricos	No	Sí	No
Puertos USB	No	4 (2x USB3.0 + 2x USB2.0)	No
Interfaces	I2C, SPI, Serial	I2C, SPI, Serial, CSI, DSI	I2C, SPI, CAN, PWM,
Voltaje entrada	7-12 V	5V	2.3 V - 3.6 V
Alimentación	USB-B / Conector jack	USB-C	USB-C
Lenguajes usados	C/C++	Python, HTML 5 y JavaScript	C/C++
Dimensiones	101.52x53.3 mm ²	85.6x56.5x11 mm	51 x 23 x 8 mm

Fuente: (Raspberry Pi Ltd, 2022); (Arduino, 2022); (García D. , 2021)




Realizado por: Guamán Lessly, López Kevin, 2023

De la tabla 3-8 se seleccionó la Raspberry Pi ya que satisface los requerimientos del sistema, permite la comunicación mediante WiFi y el almacenamiento de datos en la nube, cuenta con entradas de periféricos que se pueden conectar en puertos USB y HDMI y se puede programar en diversos lenguajes de programación. Además, entre las opciones comparadas es la que posee mejores recursos de memoria y almacenamiento, siendo la tarjeta más ágil.

3.5.7 Fuente de alimentación.

El prototipo debe tener un funcionamiento autónomo, sin intervención de personas, por lo tanto, se requiere de una fuente de alimentación de energía de corriente continua. Existen múltiples fuentes de alimentación, por lo que para seleccionar la adecuada para el sistema se realizó un análisis de las más comunes en la Tabla 3-9.

Tabla 3-9: Comparación fuentes de alimentación

	BATERÍA 12V / 9 Ah	DURACELL RECARGABLE 9 V	POWER BANK
Imagen			
Voltaje nominal	12V	9 V	5-9V
Capacidad nominal	9.0 Ah	170 mAh	10000mAh
Tiempo de vida	5 años	5 años	-
Números de celdas	6	-	-
Recargable	Sí	Sí	Sí
Temperatura funcionamiento	0 a 40 °C -20 a 55 °C -15 a 40°C	-18 a 55°C	0-40 °C
Corriente de carga máxima	2,8 A	1,70 A	2 A
Auto descarga mensual	2% (20°C)	4.8 V (21°C)	5V (38°C)
Terminal	T2 (250)	Broche	USB
Material de contenedor	ABS(UL94-HB)	-	Polímero
Longitud	151 mm	48,5 mm	138 mm
Ancho	65 mm	26,5 mm	66,8 mm
Altura	100 mm	17,5 mm	15,8 mm
Peso	2,32 Kg	60 g	237 g

Fuente: (ASC Electrónica S.A, 2022) ; (Energizer, 2021); (Amazon, 2022).

Realizado por: Guamán Lessly, López Kevin, 2023

En la tabla 3-10 se muestra el consumo eléctrico de los componentes eléctricos y electrónicos a utilizar en el prototipo.

Tabla 3-10: Consumo eléctrico del prototipo

ELEMENTO	CONSUMO
Cámara	0,24 A
Foco Led	0,75 A
Módulo GPS	0,046 A
Servomotor	0,5 A
Sensor	0,5 A
Bocina	2 A
Raspberry	3 A
TOTAL	7,036 A

Realizado por: Guamán Lessly, López Kevin, 2023

Considerando que el consumo eléctrico total del prototipo es de 7,036 A se seleccionó la batería de 12 V y 9Ah que satisface este consumo, además debido a que el prototipo estará situado en ambientes externos se optó por esta batería ya que su temperatura de funcionamiento es bastante flexible y el material del contenedor es apto para las condiciones de funcionamiento.

3.5.8 Diagramas de conexión del hardware

Para realizar las conexiones del hardware previamente se diseñó y simuló en el software Proteus, por las siguientes razones:

- Validación del diseño: verificar el funcionamiento correcto antes de implementarlo en la realidad. Esto evita errores costosos y tiempo perdido en la construcción de un circuito que no funciona como se espera.
- Depuración y detección de problemas: si el circuito no funciona correctamente durante la simulación, es más fácil detectar y corregir los problemas antes de llevar el diseño a la realidad.
- Seguridad y prevención de daños: Al simular conexiones, se asegura que no haya cortocircuitos u otros problemas que podrían dañar los componentes o poner en riesgo la seguridad del usuario.
- Documentación y presentación: Las simulaciones proporcionan una forma visual y clara de presentar y documentar el diseño del circuito.

En la ilustración 3-11 se muestra el diagrama de conexión del hardware implementado en el prototipo.

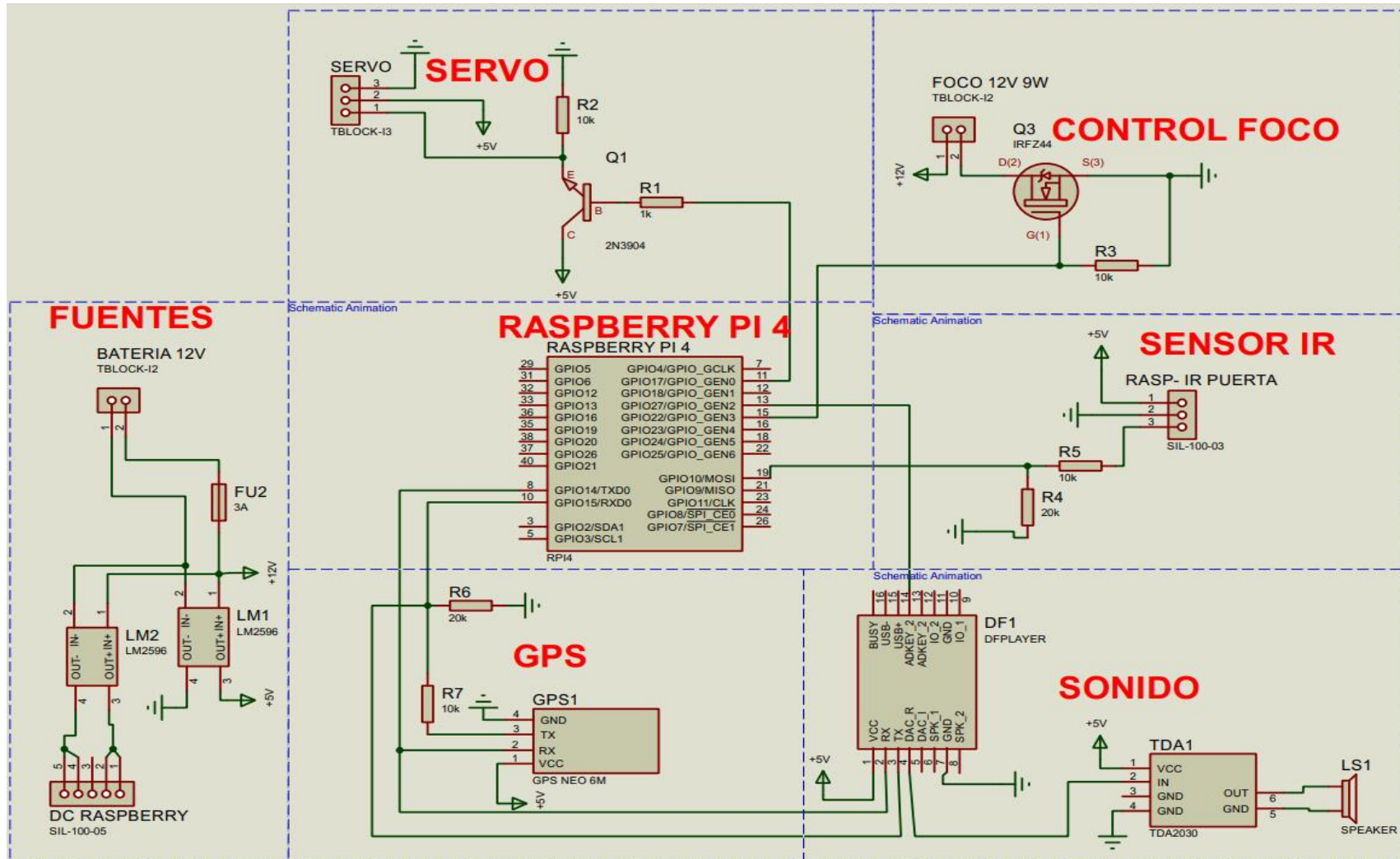


Ilustración 3-11: Esquema de conexión completo del hardware

Realizado por: Guamán Lessly, López Kevin, 2023

El esquema de conexiones del sistema fue desarrollado por bloques para facilitar su comprensión y evitar errores en el desarrollo de las conexiones de los componentes del hardware.

- Con la finalidad de proteger el hardware de procesamiento, se energizó el sistema en dos partes, con un módulo LM2596 se suministró energía específicamente a la Raspberry, y con otro módulo LM2596 se abasteció de energía al resto del circuito, estos módulos reducen el voltaje de entrada de la batería de 12 V a 5V, asegurando que los componentes operen con el voltaje adecuado. Adicionalmente se incorporó un fusible para controlar el flujo de corriente y prevenir posibles daños.
- Para conectar el GPS a la Raspberry, ambos deben estar alimentados, y los pines Tx y Rx del GPS se conectan a los pines físicos 8 y 10 de la Raspberry, que representan Txd y Rxd respectivamente, tomando en consideración que Tx del GPS se conecta al Rxd de la Raspberry, mientras que Rx del GPS se conecta al Txd de la Raspberry. Además, se conectaron resistencias para garantizar una comunicación efectiva entre ambos dispositivos, variando la señal entre 0 V en su nivel más bajo y 3.3 V en su nivel más alto.
- La conexión correspondiente al foco led utiliza un Mosfet que actúa como interruptor para activar el foco, este componente se activará mediante una señal emitida por la Raspberry, así mismo se incorporaron resistencias que eliminan ruidos y aseguran que la señal recibida sea precisa.

Los dispositivos mencionados están en constante funcionamiento, sin embargo, hay otras conexiones que empiezan a operar cuando el nivel de llenado del contenedor alcanza su nivel de llenado máximo, como es el caso de:

- El servomotor que se activa bloqueando la puerta del basurero, este dispositivo está conectado a la Raspberry mediante el pin físico 11.
- El sensor infrarrojo que está conectado a la Raspberry a través del pin físico 19, detecta cualquier intento de ingreso al contenedor bloqueado, y si se detecta, emite una señal para activar la alerta de audio mediante el DFPlayer Mini MP3 que está conectado a la Raspberry mediante el pin físico 13, cuando la señal es recibida, tanto el módulo TDA2030 como el parlante se activan simultáneamente, emitiendo un mensaje de voz fuerte y claro.

En la Tabla 3-11 se muestra de forma detallada las conexiones del sistema.

Tabla 3-11: Detalles de conexión del hardware

FUENTE											
EMISOR					RECEPTOR						
Etiqueta	Código	ENTRADAS		SALIDAS		Etiqueta	Código	ENTRADAS		SALIDAS	
Batería de 12 V	TBLOCK-12	-	-	1	GND	LM1	LM2596	1	GND	3	5V
		-	-	2	12V			2	12V	4	GND
LM2	LM2596	1	GND	3	3.3V	DC Raspberry	SIL-100-50	1	GND	3	3.3V
								2	12V	4	GND
		2	12V	4	GND			3	3.3V	1-2	PIN (1-17)
								4	GND	4-5	PIN (6-9-14-20-25-30-34-39)

MÓDULO GPS											
EMISOR						RECEPTOR					
Etiqueta	Código	ENTRADAS		SALIDAS		Etiqueta	Código	ENTRADAS		SALIDAS	
GPS1	GPS NEO 6M	1	5V	2	RX	RASPBERY PI 4	RPI4	PIN 8	TX	-	-
		4	GND	3	TX			PIN 10	RX	-	-
SERVO MOTOR											
EMISOR						RECEPTOR					
Etiqueta	Código	ENTRADAS		SALIDAS		Etiqueta	Código	ENTRADAS		SALIDAS	
RASPBERY PI 4	RPI4	-	-	PIN 11 GPIO 17	PWM	Q1	2N3904	B	PWM	E	5V
		-	-					C		5V	-
Q1	2N3904	B	PWM	E	5V	SERVO	TBLOCK-13	1	5V	-	-
		C	5V	-	-			2	5V	-	-
								3	GND	-	-
CONTROL FOCO											
EMISOR						RECEPTOR					
Etiqueta	Código	ENTRADAS		SALIDAS		Etiqueta	Código	ENTRADAS		SALIDAS	
RASPBERY PI 4	RPI4	-	-	PIN 15 GPIO 22	5V	Q3	IRFZ44	1 (G)	5V	2 (D)	GND
		-	-					3 (S)		GND	-
Q3	IRFZ44	PIN 15 1 (G)	5V	2 (D)	GND	FOCO 12V 9W	TBLOCK-12	1	12V	-	-
		3 (S)	GND	-	-			2	GND	-	-
								2	2 (D)	GND	-
SONIDO											
EMISOR						RECEPTOR					
Etiqueta	Código	ENTRADAS		SALIDAS		Etiqueta	Código	ENTRADAS		SALIDAS	
RASPBERY PI 4	RPI4	PIN 8	TX	PIN 13 GPIO 27	5V	DF1	DFPLAYER	13	ADKEY_1	2	RX
		PIN 10	RX					PIN 13 GPIO 27		5V	3
								7	GND	4	DAC_R
DF1	DFPLAYER	13	ADKEY_1	2	RX	TDA	TDA2030	1	5V	5	GND
		1						5V	3	TX	
		7	GND	4	DAC_R			2	DAC_R	6	OUT
TDA	TDA2030	1	5V	5	GND	LS1	SPEAKER	5	GND	"MENSAJE DE VOZ"	
		2	DAC_R	6	OUT			6	OUT		
		4	GND								

Realizado por: Guamán Lessly, López Kevin, 2023

3.6 Software

En esta sección se selecciona el lenguaje de programación a emplear para la codificación de los algoritmos del sistema y se muestra la plataforma IoT a utilizar para la creación de la interfaz gráfica remota.

3.6.1 Lenguajes de programación

Entre los lenguajes de programación más utilizados se encuentra C ++, C #, Visual Basic, Java y Python. En la Tabla 3-12 se realiza una comparación de las características de los lenguajes de programación mencionados.

Tabla 3-12: Comparación de Lenguajes de Programación.

	C++	C#	VISUAL BASIC	JAVA	PYTHON
Programación orientada a objetos	Sí	Sí	No	Sí	Sí
Cuenta con librerías	Sí	Sí	Sí	Sí	Sí
Apto para creación de páginas web	No	Sí	Sí	Sí	Sí
Visualmente atractivo	No	No	Sí	Sí	Sí
Sistemas operativos	Todos los sistemas operativos, con su respectiva versión para dicho sistema	Unix Linux Solaris Windows Mac	Microsoft Windows Ms-dos	Todos los sistemas operativos	Todos los sistemas operativos
OpenCV	Sí	Sí	No	Sí	Sí
Información disponible	Normal	Mucha	Normal	Mucha	Mucha
Nivel de lenguaje	Bajo	Alto	Alto	Alto	Alto
Facilidad de aprender	Difícil	Normal	Fácil	Fácil	Muy fácil

Fuente: (Challenger et al., 2014).

Realizado por: Guamán Lessly, López Kevin, 2023

Se seleccionó el lenguaje Python, porque tiene librerías directas para visión artificial, como OpenCV, es de código abierto, compacto y de fácil entendimiento, permite crear interfaces gráficas remotas, existe mucha información disponible en la web que sirve de guía para el desarrollo de la codificación del algoritmo para el prototipo.

3.6.2 Plataforma IoT

En base a las características de las plataformas IoT indagadas en el apartado 2.2.2.3 se seleccionó la plataforma Blynk IoT, al ser capaz de enviar y recibir información que se puede recopilar y almacenar, cuenta con librerías disponibles para la Raspberry Pi, es sencilla de usar y permite crear interfaces gráficas amigable con el usuario.

3.7 Desarrollo de la programación

Para crear el sistema que integra la identificación del nivel de llenado de los contenedores de basura mediante visión artificial, la comunicación inalámbrica con el hardware seleccionado y una interfaz gráfica remota, se codificaron diferentes algoritmos que se detallan en este apartado.

3.7.1 Visión artificial

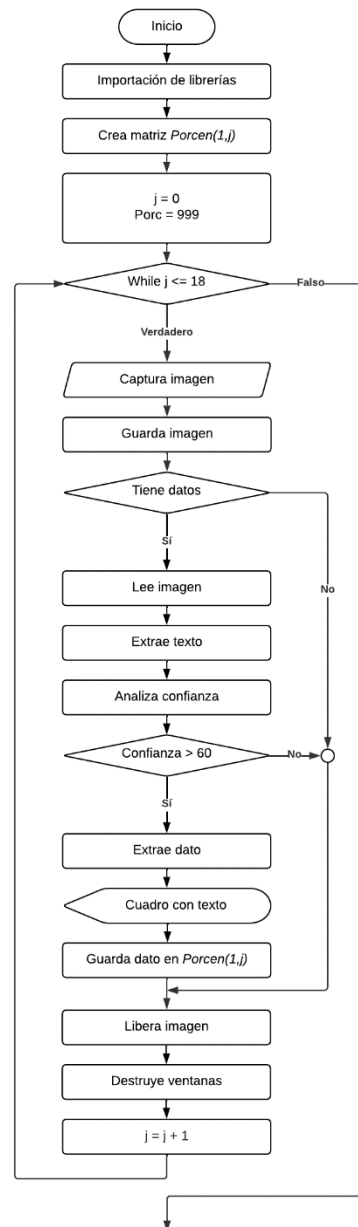


Ilustración 3-12: Recolección de datos

Realizado por: Guamán Lessly, López Kevin, 2023

La ilustración 3-12 muestra el funcionamiento del algoritmo de visión artificial, el proceso inicia con la importación de librerías para procesamiento de imágenes y la creación de la matriz $Porcen(1,j)$, luego se captura una imagen con la cámara web, si la imagen obtenida contiene caracteres correspondiente al patrón del nivel de llenado se extrae esta información y se analiza su confianza, si la confianza es mayor al 60% se la almacena en la matriz, cuando finaliza este proceso elimina la captura e inicia nuevamente el ciclo desde la captura de la imagen, este ciclo se repite por 18 veces. Cuando se realizan las 18 repeticiones se realizan las acciones mostradas en la ilustración 3-13.

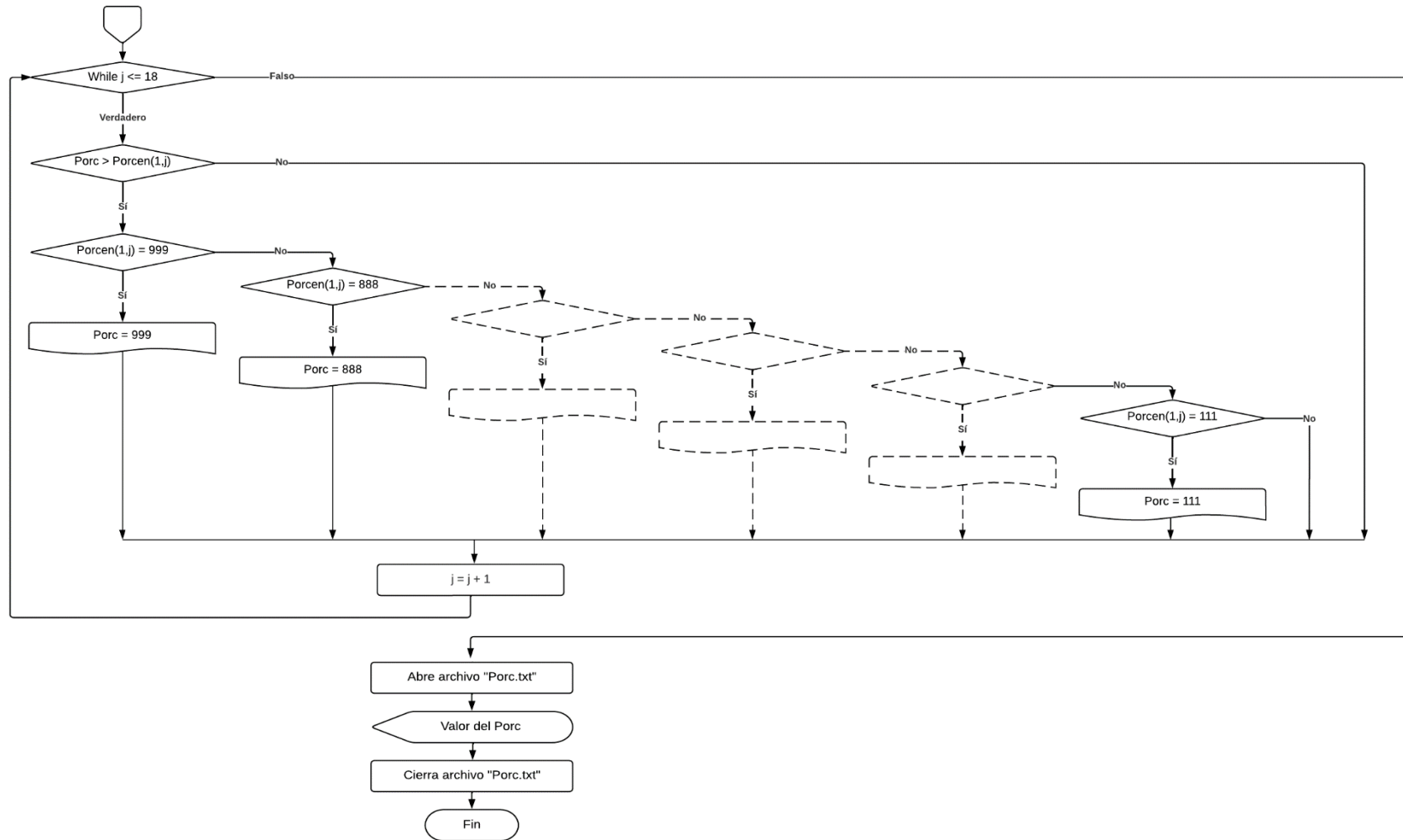


Ilustración 3-13: Análisis y almacenamiento de datos

Realizado por: Guamán Lessly, López Kevin, 2023

La ilustración 3-13 es la continuación del funcionamiento del algoritmo de visión artificial, los datos almacenamiento en la matriz inicial $Porcen(1,j)$, corresponden a varias capturas realizadas por la cámara web, por lo que es necesario compararlos e identificar el valor menor, el cual se guarda en un archivo de texto denominado *Porc.txt*.



Ilustración 3-14: Detección del nivel de llenado del basurero cuando está al 50%

Realizado por: Guamán Lessly, López Kevin, 2023

La ilustración 3-14 corresponde a una imagen capturada por el algoritmo de visión artificial de los caracteres 5 del nivel de llenado, que se enmarcó en un recuadro verde, y el dato extraído apareció en color rojo, este valor es guardado y comparado como se explicó previamente.

3.7.2 Codificación para la ubicación de los contenedores de basura

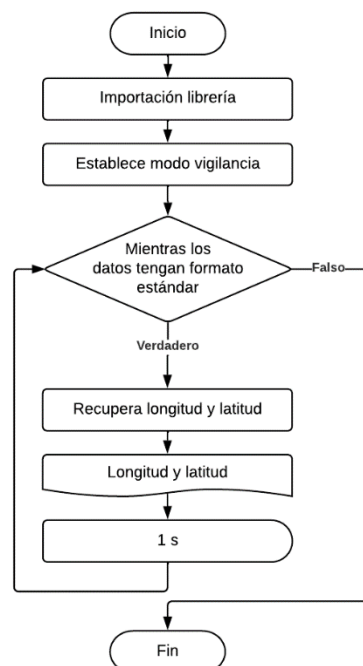


Ilustración 3-15: Obtención de la ubicación de los basureros

Realizado por: Guamán Lessly, López Kevin, 2023

En la ilustración 3-15 se describe el funcionamiento del algoritmo que obtiene la ubicación del prototipo, el proceso inicia con la importación de la librería *gps* que recopila información del módulo GPS conectado a la Raspberry y se extrae únicamente la longitud y latitud que se muestran en la interfaz gráfica, este proceso de obtención de coordenadas geográficas se realiza constantemente cada segundo.

3.7.3 Codificación para la activación del servo motor

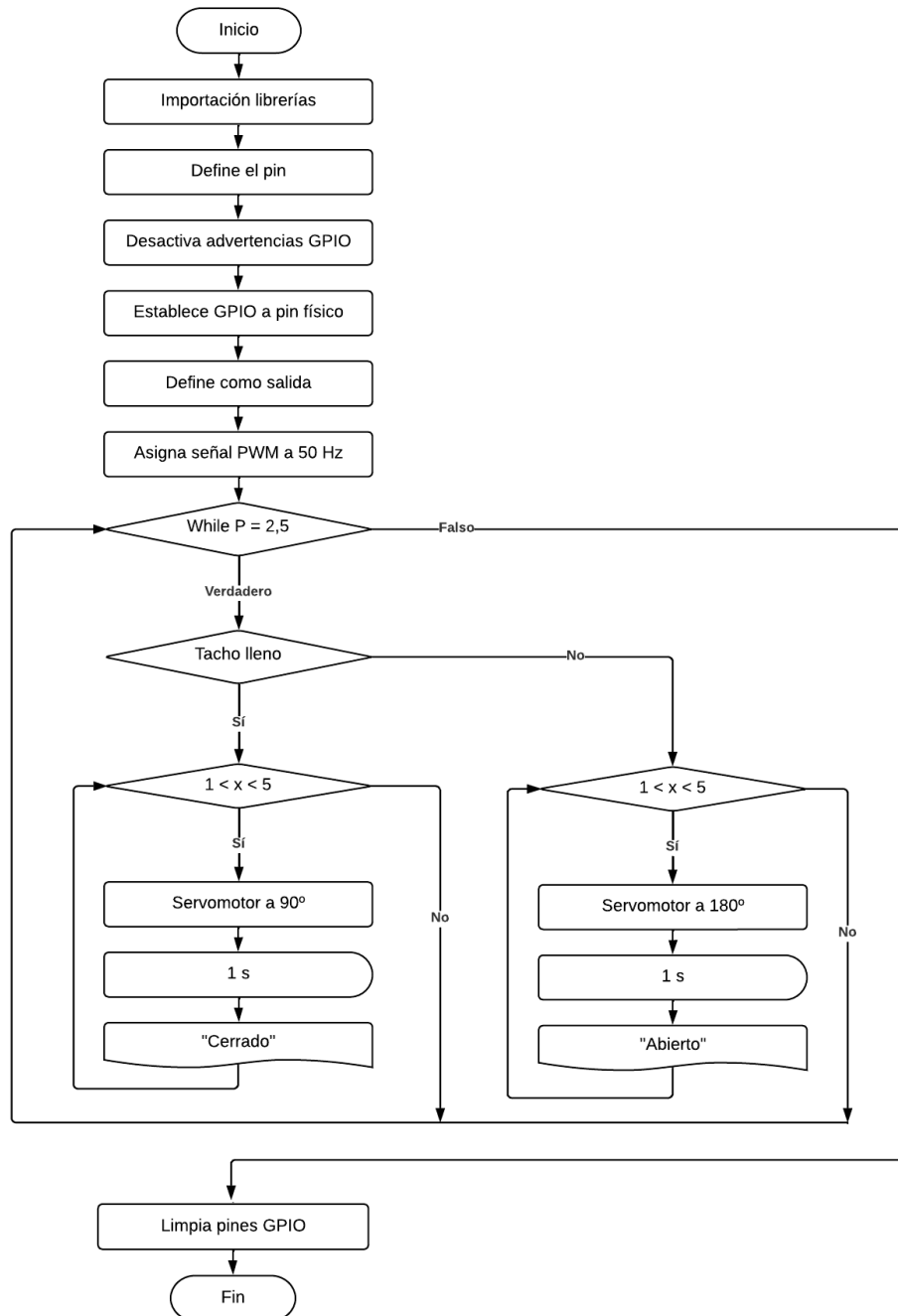


Ilustración 3-16: Procedimiento para el bloqueo del contenedor de basura

Realizado por: Guamán Lessly, López Kevin, 2023

En la ilustración 3-16 se detalla el funcionamiento del algoritmo que controla el servomotor, la codificación inicia con la importación de la librería *GPIO* que permite controlar digitalmente el actuador, se definió el pin a utilizar *servoPin = 11* y se desactivaron los mensajes de advertencia producidos durante la ejecución del código a fin de mejorar el rendimiento del proceso. Luego se configuraron los pines *GPIO* como salidas digitales con la función *GPIO.setup(servoPIN, GPIO.OUT)* y se asignó una señal PWM (modulación por ancho de pulso) de 50 Hz, es decir que la señal cambiará 50 veces por segundo, que inicia con un valor de 2,5.

Se creó un bucle con 5 iteraciones, que compara si el voltaje es menor a 5, en caso de serlo el servomotor se activa y después de 1 segundo muestra en la interfaz gráfica la palabra “Cerrado”. En caso de ser un voltaje mayor a 5, ocurre lo contrario, se desactiva el servomotor y se imprime la palabra “Abierto”.

3.7.4 Codificación para la interfaz gráfica remota

Se descargó la aplicación Blynk IoT disponible en la app store y se creó una cuenta para la creación de la interfaz gráfica remota, para conseguir el diseño deseado se utilizaron los elementos prediseñados que ofrece la aplicación y para su codificación y conexión con el sistema se instaló el paquete pip de Python con los comandos:

```
sudo apt install python3-pip  
python3 -m pip install requests
```

Se insertó la función *write()* que permite a la Raspberry Pi escribir un valor en cualquier pin virtual de la plataforma Blink, con esto se obtiene de entrada el valor token, el número del pin virtual y el valor a escribir. Después se ingresó la función *read* que lee un valor widget y la regresa al programa Python, para lo cual requiere únicamente el token y un valor pin. Posteriormente se genera un pin virtual con la línea *write(token, "v1", "150")*.

Luego se leyó el pin virtual generado y se ejecutó el programa, obteniendo:

```
pi@raspberrypi:~ $ python3 restblynk.py  
Value successfully updated  
0
```

La primera línea muestra que se cargó exitosamente el valor “150” y el “0” es el valor leído desde el botón (*OFF="0" and "ON"=1*).

3.7.5 Codificación general del sistema

Una vez codificados los algoritmos de visión artificial, de la ubicación de los contenedores, el bloqueo de la puerta y la interfaz gráfica se realiza un programa final que integra los algoritmos mencionados y el control de todos los elementos que conforman el prototipo.

v

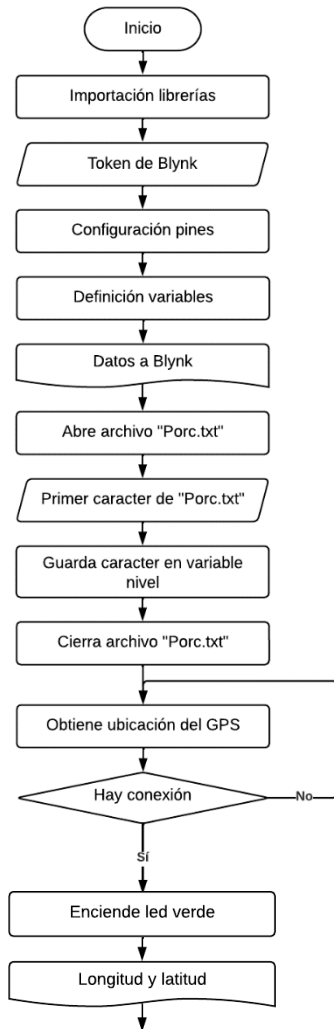


Ilustración 3-17: Diagrama de flujo del funcionamiento del sistema

Realizado por: Guamán Lessly, López Kevin, 2023

Se importaron todas las librerías, resaltando *request* que actualiza la API de Blynk y se creó una variable denominada *token* que identifica el usuario en la aplicación, luego se configuraron los pines, es decir se estableció los números de pines a utilizar con el código *GPIO.setmode(GPIO.BOARD)*, lo que significa que la numeración es igual a como se encuentra en la placa física de la Raspberry. Posteriormente se definieron las variables a utilizar para obtener la ubicación y el control del servomotor y se enviaron los datos obtenidos a Blynk, para el análisis de los datos adquiridos mediante visión artificial se guardó la información del archivo *Porc.txt* en una variable denominada *nivel=archi.read(1)* que lee únicamente el primer caracter.

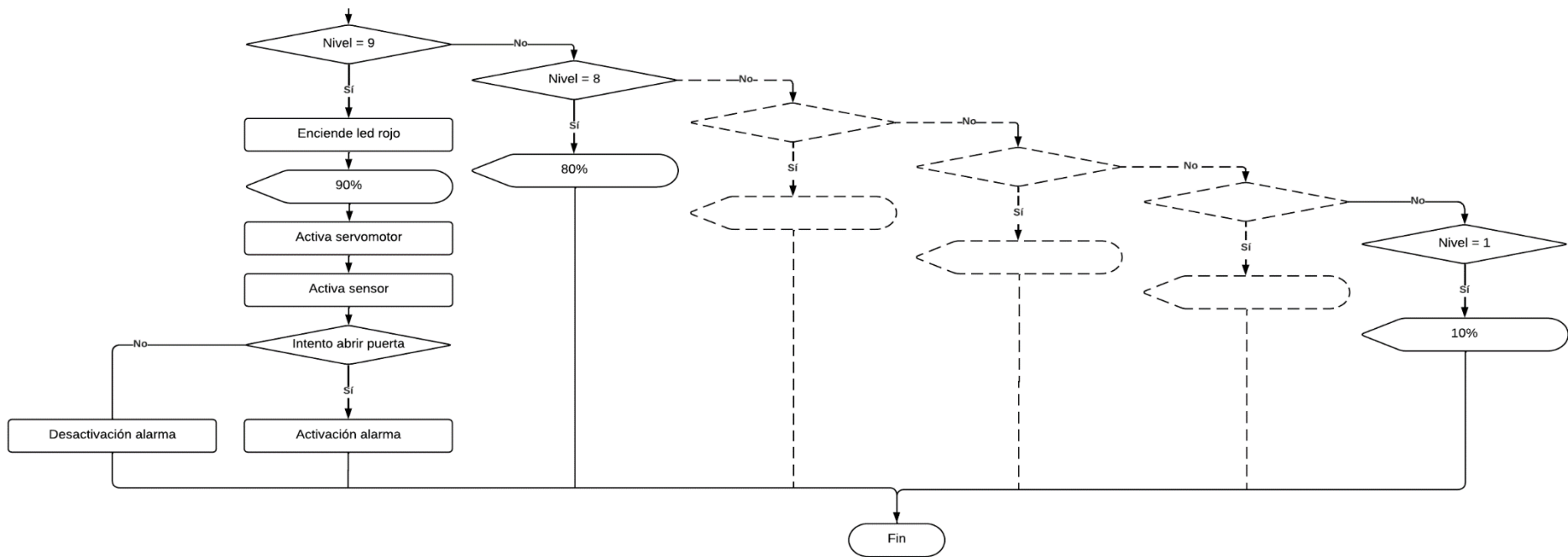


Ilustración 3-18: Diagrama de flujo del funcionamiento del sistema

Realizado por: Guamán Lessly, López Kevin, 2023

La ilustración 3-18 continúa el proceso explicado en la ilustración 3-17, en donde una vez obtenido las coordenadas del contenedor de basura se envían a la interfaz gráfica mediante un http y se enciende un indicativo en forma de led verde que significa que hay conexión con el módulo GPS. Finalmente, se envía el porcentaje de llenado del contenedor de basura de la variable *nivel* a la interfaz gráfica en un http, en caso de ser igual al 90% se ejecutan las siguientes tareas: activación del servomotor que bloquea la puerta y del sensor infrarrojo que se encarga de activar la alerta de voz si existe un intento de acceso, a su vez enciende un indicativo en forma de led rojo en la interfaz que señala que el basurero está lleno; en caso de ser un valor diferente a 90% no se ejecutan las tareas anteriores y se apaga el led.

CAPÍTULO IV

4 MARCO DE ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

4.1 Prototipo

El prototipo finalizado integra los recursos tanto del hardware como del software especificados en el capítulo III.



Ilustración 4-1: Prototipo finalizado

Realizado por: Guamán Lessly, López Kevin, 2023

En la ilustración 4-1 se muestra la estructura del prototipo finalizado, construido según el diseño CAD realizado en SolidWorks que se detalló en el capítulo anterior, externamente está cubierto por pegatinas que, a más de proteger la estructura de factores externos, dan al prototipo una imagen agradable que aluden tecnología. Internamente el prototipo tiene acoplado los componentes eléctricos y electrónicos, que están protegidos con cajas plásticas verdes que fueron impresas en 3D, de igual manera las conexiones están cubiertas por canaletas plásticas blancas, esto se puede apreciar de mejor manera en la ilustración 4-2 que muestra la parte interna del prototipo.

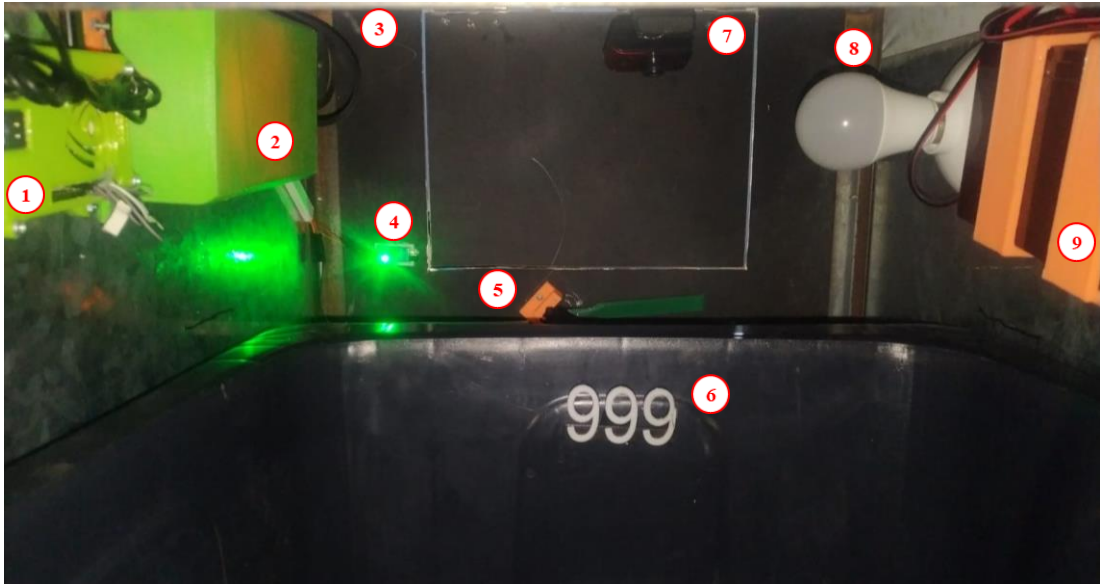


Ilustración 4-2: Parte interna del prototipo

Realizado por: Guamán Lessly, López Kevin, 2023

En el interior del contenedor se encuentran los siguientes elementos:

1. Raspberry Pi.
2. Placa PCB con las conexiones del módulo GPS, MP3 player, módulo amplificador de audio, reguladores de voltaje y componentes como resistencias y transistores.
3. Bocina
4. Sensor infrarrojo
5. Servomotor
6. Patrones del nivel de llenado
7. Cámara web
8. Foco LED
9. Fuente de alimentación

El funcionamiento del prototipo es el siguiente, mediante visión artificial el sistema detecta el nivel de llenado del contenedor de basura, información capturada por la cámara web y procesada por la Raspberry Pi, cuando el basurero alcanza un nivel de llenado del 90% la puerta del contenedor se bloquea automáticamente con el accionamiento de un servomotor y si alguien intenta acceder a este, se activa un mensaje de voz que avisa al usuario sobre el estado del contenedor. El prototipo está vinculado a una interfaz gráfica remota en la aplicación web de Blynk IoT que muestra la información del nivel de llenado, la ubicación de los contenedores e indica que contenedor lleno es el más cercano, además le permite al usuario bloquear o desbloquear remotamente la puerta del basurero.

4.2 Visión artificial

Con la utilización de la librería OpenCv y la metodología detallada en el capítulo anterior, se obtuvo el algoritmo de visión artificial en el lenguaje Python que detecta el nivel de llenado de los contenedores de basura.

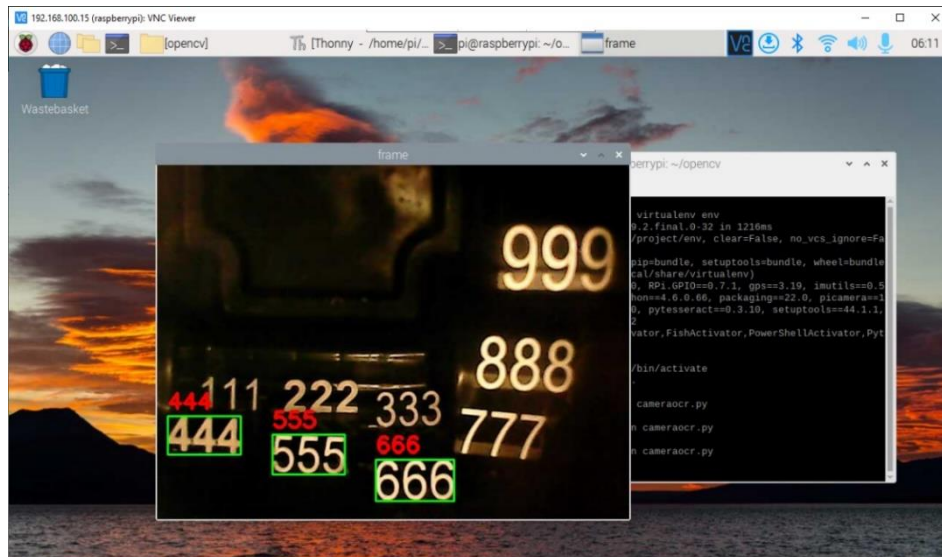


Ilustración 4-3: Funcionamiento del algoritmo de visión artificial.

Realizado por: Guamán Lessly, López Kevin, 2023

El algoritmo de visión artificial detecta el nivel de llenado del contenedor de basura mediante el reconocimiento de números que fueron colocados en el interior del contenedor. Como se puede observar en la ilustración 4-3 el algoritmo captura varias imágenes que marca en un rectángulo verde y al reconocer los caracteres de su interior los muestra en color rojo. De los caracteres identificados toma el valor menor que corresponde al nivel de llenado y lo muestra en la interfaz gráfica remota.

4.2.1 Eficiencia de la visión artificial

El cálculo de la eficiencia en sistemas de visión artificial se realiza con la utilización de la matriz de confusión, la cual evalúa el nivel de confusión de modelos inteligentes al clasificar los aciertos y desaciertos en cuatro categorías:

- Verdadero Positivo (VP): un dato “normal” clasificado correctamente como “normal”.
- Verdadero Negativo (VN): un dato “anormal” clasificado correctamente como “anormal”.
- Falso Negativo (FN): un dato “normal” clasificado incorrectamente como “anormal”.
- Falso Positivo (FP): un dato “anormal” clasificado incorrectamente como “normal”.

En la tabla 4-1 se muestra la matriz de confusión empleada para validar el algoritmo de visión artificial del prototipo.

Tabla 4-1: Matriz de confusión

		ESTIMADO	
		CARACTERES	NO-CARACTERES
REAL	CARACTERES	Verdaderos “positivos”	Falso “negativos”
	NO-CARACTERES	Falsos “positivos”	Verdaderos “negativos”

Realizado por: Guamán Lessly, López Kevin, 2023

En donde:

- Verdaderos positivos (VP): reconoce todos y solamente los caracteres en la imagen
- Verdaderos negativos (VN): reconoce todos los caracteres y objetos adicionales en la imagen.
- Falsos positivos (FP): no reconoce ningún carácter en la imagen.
- Falsos negativos (FN): no reconoce por completo el número de caracteres en la imagen.

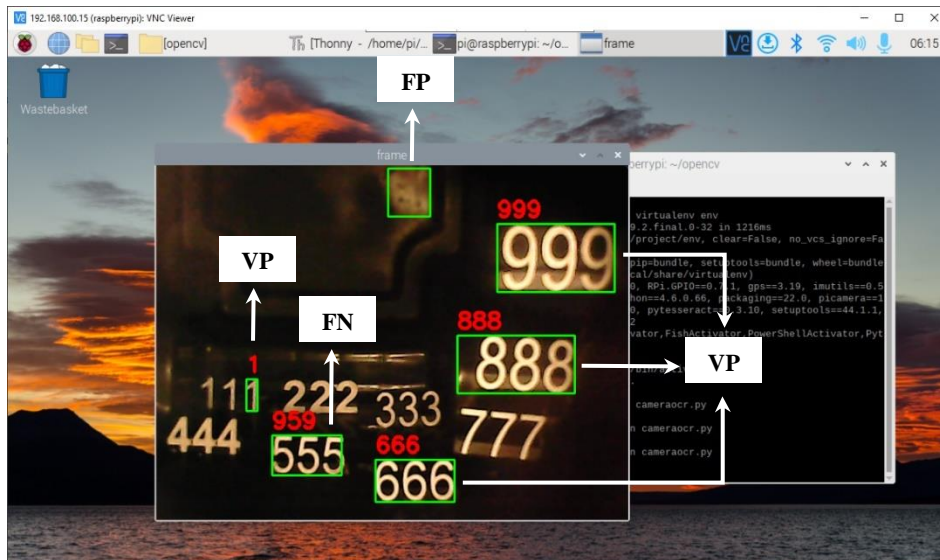


Ilustración 4-4: VP, FP y FN en las imágenes capturadas.

Realizado por: Guamán Lessly, López Kevin, 2023

La ilustración 4-4 muestra una captura realizada durante la validación del algoritmo de visión artificial, en la ilustración se observa que el algoritmo capturó varios verdaderos positivos (VP), correspondientes a los caracteres 1, 6, 8 y 9 de la escala de medición, un falso positivo (FP) en donde la imagen no contenía ningún carácter y un falso negativo (FN) en donde de los tres caracteres 5, únicamente reconoció uno y los demás detectó erróneamente como 9.

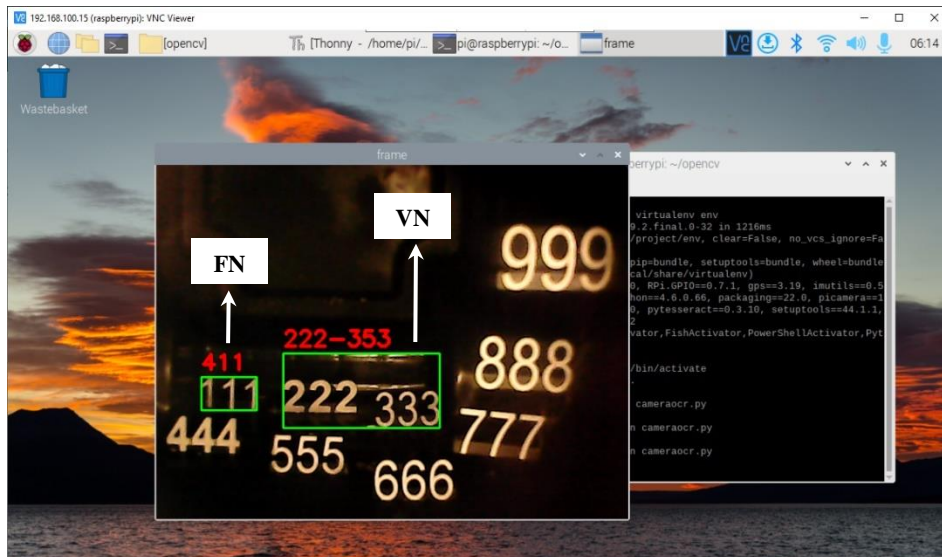


Ilustración 4-5: VN y FN en las imágenes capturadas

Realizado por: Guamán Lessly, López Kevin, 2023

En la ilustración 4-5 se observa como el algoritmo capturó un falso negativo (FN) en donde de los tres caracteres 1, solamente reconoció dos y el restante detectó erróneamente como 4, también capturó un verdadero negativo (VN) al hacer una sola captura de los caracteres 2 y 3, en donde reconoció por completo los caracteres 2, solamente dos caracteres 3 y objetos adicionales.

Para determinar la eficiencia del algoritmo de visión artificial se realizaron varias pruebas que consistieron en llenar el contenedor de basura con diferentes desechos, se grabó el proceso durante un tiempo determinado y con los datos observados se realizó el cálculo correspondiente con la siguiente fórmula:

$$E = \frac{VP + VN}{VP + VN + FN + FP} * 100 \quad (1)$$

4.2.1.1 Prueba 1

Tabla 4-2: Datos de la prueba 1

INFORMACIÓN DEL VIDEO	
Material	Papel
Duración	5:11 minutos
Imágenes capturadas	89

Realizado por: Guamán Lessly, López Kevin, 2023

En la prueba 1 se llenó el contenedor de basura con desechos de papel durante 5,11 minutos, la cantidad de imágenes capturadas por el algoritmo fue de 89.

Tabla 4-3: Matriz de confusión de la prueba 1

		ESTIMADO	
		CARACTERES	NO-CARACTERES
REAL	CARACTERES	68	2
	NO-CARACTERES	17	2

Realizado por: Guamán Lessly, López Kevin, 2023

$$E = \frac{68 + 2}{68 + 2 + 2 + 17} * 100 = 78,65 \% \quad (2)$$

Durante la realización de la prueba se observó que cuando el nivel de basura superaba el 40% de llenado la cantidad de imágenes tomadas disminuía y los falsos positivos aumentaban, esto debido a que por el color claro de los desechos la reflexión de la luz era alta dificultando al algoritmo reconocer los caracteres del nivel de llenado que son de color blanco. Como resultado de la prueba se obtuvo una eficiencia del 78,65%.

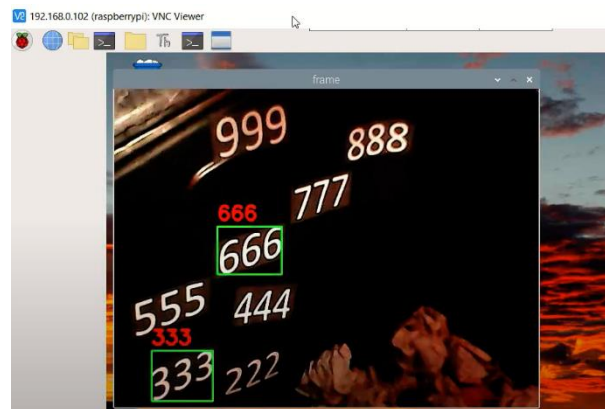


Ilustración 4-6: Prueba 1 del algoritmo de visión artificial

Realizado por: Guamán Lessly, López Kevin, 2023

En la ilustración 4-6 se observa como el algoritmo captura verdaderos positivos (VP) correspondientes a los caracteres 3 y 6 de la escala de medición.

4.2.1.2 Prueba 2

Tabla 4-4: Datos de la prueba 2

INFORMACIÓN DEL VIDEO	
Material	Desechos en fundas negras
Duración	5:08 minutos
Imágenes capturadas	132

Realizado por: Guamán Lessly, López Kevin, 2023

Con la eficiencia poco satisfactoria de la prueba 1, para la prueba 2 se depositaron desechos en fundas plásticas de color negro en un periodo de tiempo de 5,08 minutos, en esta prueba la cantidad de imágenes capturadas fue de 132.

Tabla 4-5: Matriz de confusión de la prueba 2

		ESTIMADO	
		CARACTERES	NO-CARACTERES
REAL	CARACTERES	111	16
	NO-CARACTERES	2	3

Realizado por: Guamán Lessly, López Kevin, 2023

$$E = \frac{111 + 3}{111 + 3 + 16 + 2} * 100 = 86,36\% \quad (3)$$

En la prueba 2 la reflexión de luz fue baja con respecto a la prueba 1, por lo que la cantidad de imágenes capturadas fue mayor de 132, de las cuales únicamente 2 fueron falsos positivos, mientras que los verdaderos positivos y los falsos negativos fueron altos de 111 y 16 respectivamente, obteniendo una eficiencia aceptable del algoritmo de visión artificial del 86,36%.

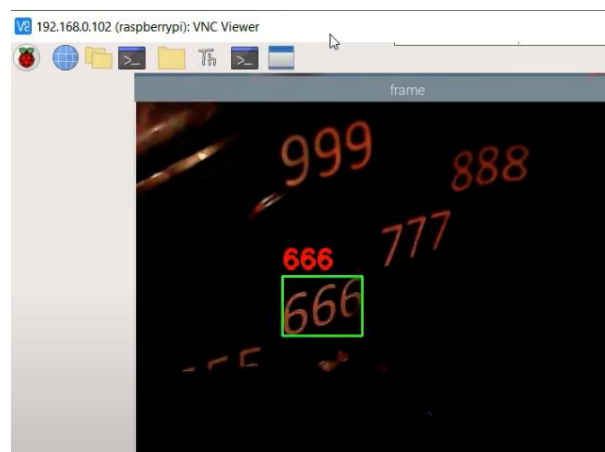


Ilustración 4-7: Prueba 2 del algoritmo de visión artificial

Realizado por: Guamán Lessly, López Kevin, 2023

En la ilustración 4-7 se muestra un verdadero positivo (VP) capturado por el algoritmo de visión del carácter 6 de la escala de medición del interior del basurero.

4.2.1.3 Prueba 3

Tabla 4-6: Datos de la prueba 3

INFORMACIÓN DEL VIDEO	
Material	Plásticos
Duración	5:06 minutos
Imágenes capturadas	160

Realizado por: Guamán Lessly, López Kevin, 2023

En la prueba 3 se llenó el contenedor de basura con diferentes desechos plásticos, como fundas, botellas, envases, entre otros, durante un lapso de 5,06 minutos; la cantidad de imágenes capturadas por el algoritmo fue de 160.

Tabla 4-7: Matriz de confusión de la prueba 3

		ESTIMADO	
		CARACTERES	NO-CARACTERES
REAL	CARACTERES	35	6
	NO-CARACTERES	44	75

Realizado por: Guamán Lessly, López Kevin, 2023

$$E = \frac{35 + 75}{35 + 75 + 6 + 44} * 100 = 68,75\% \quad (4)$$

En esta prueba los desechos plásticos depositados en el contenedor de basura contenían números, letras y figuras, por lo que la cantidad de imágenes capturadas es mucho mayor respecto a las pruebas anteriores de 160, sin embargo, se obtuvieron 75 verdaderos negativos, debido a que el algoritmo a más de capturar imágenes de los caracteres de la escala de medición capturó imágenes de los desechos, dando como resultado una eficiencia baja del 68,75%.

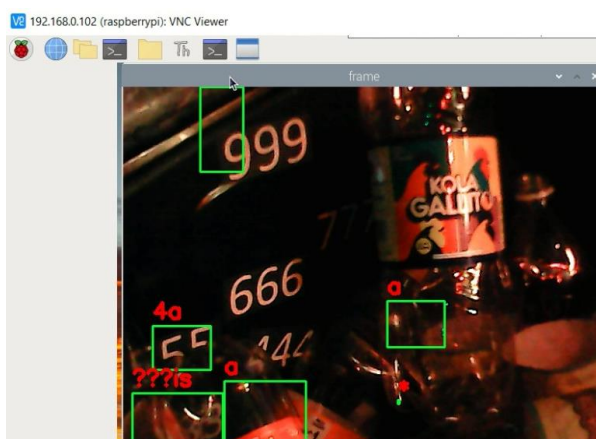


Ilustración 4-8: Prueba 3 del algoritmo de visión artificial

Realizado por: Guamán Lessly, López Kevin, 2023

Durante la prueba 3 el algoritmo de visión artificial se confundió con los desechos y capturó imágenes que no corresponden a la escala de medición empleada en el sistema, es decir falsos positivos como se muestra en la ilustración 4-8.

4.3 Validación de la flexibilidad del sistema

La comunicación entre el prototipo y el sistema inteligente de gestión de recolección de desechos sólidos urbanos se basa en una topología tipo estrella como se muestra en la ilustración 4-9.

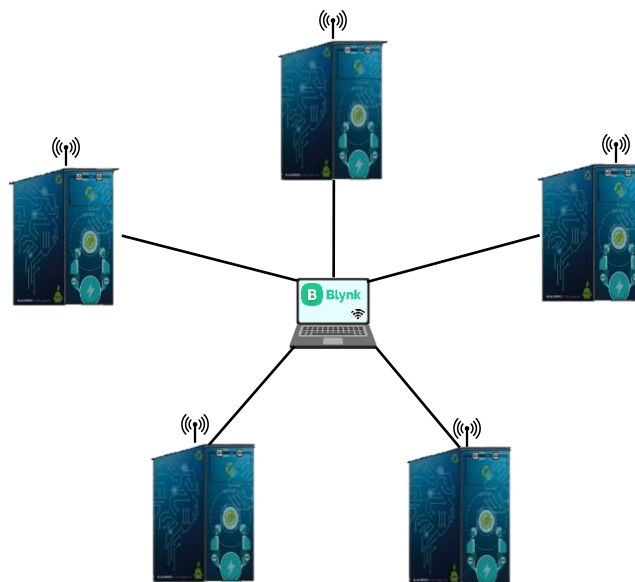


Ilustración 4-9: Comunicación del sistema.

Realizado por: Guamán Lessly, López Kevin, 2023

Para establecer la comunicación entre los basurero y la interfaz gráfica remota se empleó el protocolo http, se llamó la API de Blynk IoT con la siguiente línea:

```
api_url = "https://blynk.cloud/external/api/update?token="+token+"&"+pin+"="+value
```

Y se actualizaron las variables de la interfaz gráfica remota que almacenan la longitud y latitud.

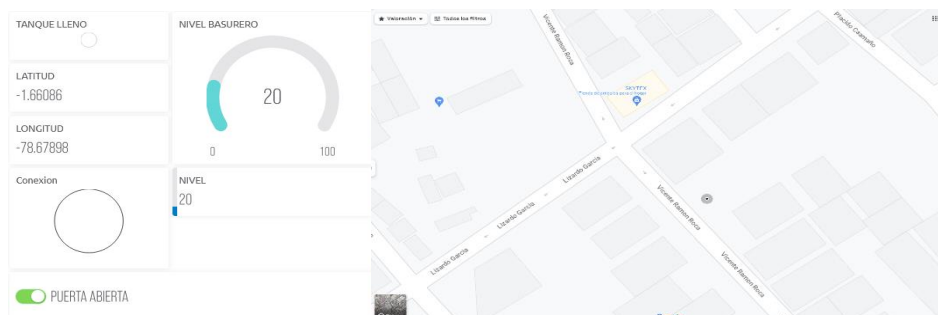


Ilustración 4-10: Desarrollo inicial de la interfaz gráfica remota

Realizado por: Guamán Lessly, López Kevin, 2023

Inicialmente se desarrolló un solo prototipo, como se muestra en la ilustración 4-10, no obstante, se desarrolló un segundo prototipo que se enlazó al sistema al modificar las variables de almacenamiento de la ubicación, con ello se demostró que el sistema es apto para incorporar más nodos (basureros). Vale recalcar que la aplicación de Blynk ofrece 8 variables para su versión gratuita, por lo que para incorporar el segundo prototipo fue necesario adquirir la versión plus.

4.4 Interfaz gráfica

La interfaz gráfica remota se realizó en la aplicación Blynk IoT, está disponible para dispositivos móviles y para computador.

4.4.1 Interfaz gráfica web

La interfaz gráfica web muestra el nivel de llenado de los contenedores de basura, su ubicación, información del estado como si la puerta está abierta o cerrada y si existe conexión con los prototipos, así mismo cuando ambos contenedores se encuentran llenos indica cual es el más cercano, considerando como punto de partida los Talleres municipales del GADM de Riobamba, que es de donde salen los camiones recolectores de basura.

Toda la información expuesta sirve para monitorear el nivel de llenado de los contenedores, realizar una recolección optimizada y llevar a cabo tareas de inspección y mantenimiento a tiempo.

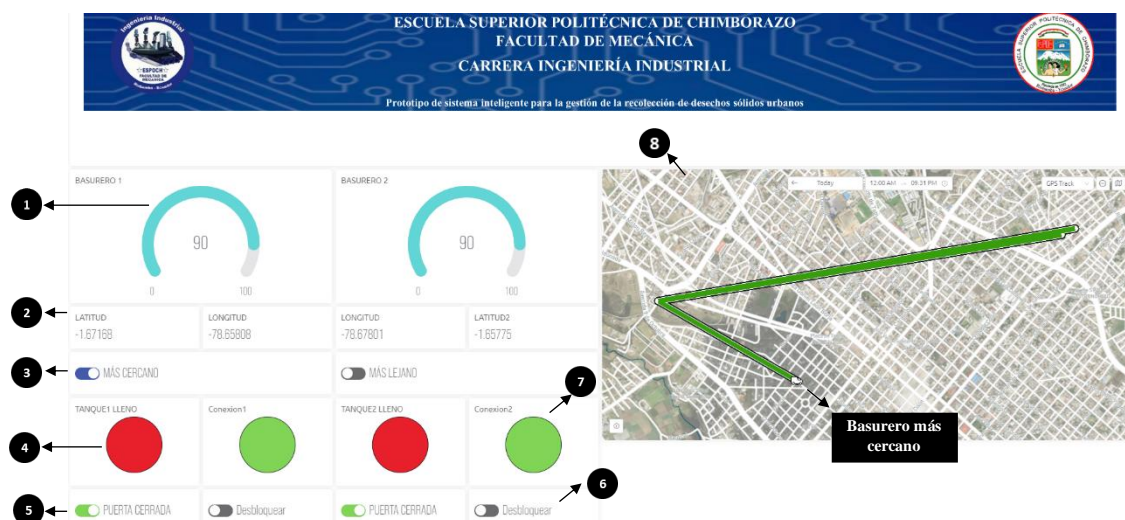


Ilustración 4-11: Interfaz gráfica web cuando los basureros están llenos.

Realizado por: Guamán Lessly, López Kevin, 2023

En la ilustración 4-11 se muestra la interfaz gráfica web, en donde:

1. Porcentaje del nivel de llenado del contenedor
2. Coordenadas geográficas del contenedor.
3. Barra deslizante que indica si es el contenedor más cercano o lejano.
4. Led rojo que se activa cuando el tanque está lleno
5. Barra deslizante que indica si la puerta está abierta o cerrada
6. Barra deslizante que permite bloquear o desbloquear la puerta del contenedor.
7. Led verde que se activa cuando existe conexión con el contenedor.
8. Basurero lleno más cercano.

Se observa que cuando el basurero ha alcanzado su capacidad máxima de llenado, el led de tanque lleno se marca de color rojo, la barra deslizante cambia de estado a “puerta cerrada”, el cual se puede cambiar mediante la barra deslizante “Desbloquear”. También se visualiza que el led verde de conexión se encuentra encendido, lo que señala una correcta comunicación entre el prototipo y la interfaz gráfica. En el mapa se marca la ubicación de los dos basureros y en el basurero lleno más cercano se muestra un indicador en forma de camión recolector blanco.

4.4.2 Interfaz gráfica desde la aplicación móvil

De manera análoga a la interfaz gráfica web, la interfaz que se visualiza desde la aplicación móvil posee elementos que indican las coordenadas del basurero, un led que señala si la puerta está cerrada o abierta, el nivel de llenado y un pulsador que permite bloquear o desbloquear el prototipo.

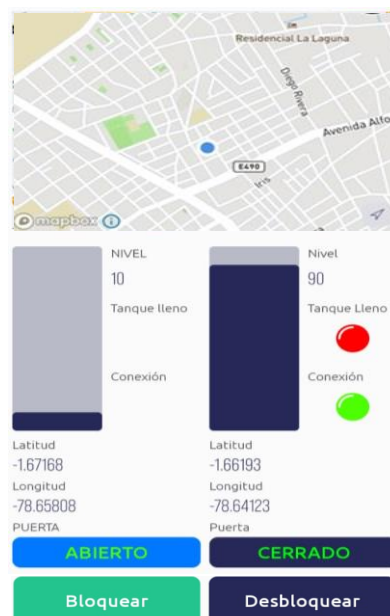


Ilustración 4-12: Interfaz gráfica desde la app móvil.

Realizado por: Guamán Lessly, López Kevin, 2023

En la ilustración 4-12 se muestra la interfaz gráfica desde la app móvil, en esta captura se evidencia las diferencias en la interfaz gráfica cuando uno de los basureros alcanza su nivel de llenado máximo, en el basurero 2 se observa que se activó un led rojo denominado “tanque lleno”, y el indicador inferior cambia de estado a “Cerrado”, así mismo el pulsador cambia de estado a “Desbloquear”.

4.5 Estabilidad de los datos de ubicación del prototipo

En este apartado se analiza la variación de los datos de latitud y longitud obtenidas por el sistema, para realizar la prueba se colocaron los contenedores en lugares de afluencia que generan grandes cantidades de desechos sólidos, como son zonas aledañas a los mercados, en donde se presentan problemas de inoportuna recolección de basura y por consiguiente desbordamiento de los contenedores.



Ilustración 4-13: Desbordamiento de contenedor de basura

Realizado por: Guamán Lessly, López Kevin, 2023

En la ilustración 4-13 se muestra el desbordamiento de un contenedor de basura ubicado a una cuadra del Centro Comercial Popular - La Condamine en la calle Carabobo entre Barón de Carondelet y Bocayá en la ciudad de Riobamba.

Las pruebas de precisión de ubicación se realizaron comparando los datos emitidos por el GPS NEO 6M, instalado en el prototipo, con dos Smartphones, uno con sistema operativo Android y otro con el sistema operativo iOS, en diferentes días y horas con el objetivo de garantizar la confiabilidad de los datos.

4.5.1 Ubicación 1

La ubicación 1 seleccionada fue en la calle Vicente Ramón Roca entre Lizardo García y Marcelo Suárez a media cuadra del Mercado “La Esperanza 1”, los datos obtenidos en esta prueba se visualizan en la Tabla 4-8.

Tabla 4-8: Datos emitidos por el prototipo en la ubicación 1

		MERCADO “LA ESPERANZA”						
		Android		Contenedor (GPS Neo 6M)		iOS		
	Fecha	Hora	Latitud	Longitud	Latitud	Longitud	Latitud	Longitud
1	22/5/2023	6:00:00	-1,661943	-78,640813	-1,661683348	-78,640935385	-1,661988	-78,640960
2	22/5/2023	9:00:00	-1,661931	-78,640844	-1,661683348	-78,640935385	-1,661537	-78,640843
3	22/5/2023	12:00:00	-1,661943	-78,640824	-1,662285864	-78,640536609	-1,661814	-78,640798
4	22/5/2023	15:00:00	-1,661933	-78,640821	-1,662285864	-78,640536609	-1,661546	-78,640814
5	22/5/2023	18:00:00	-1,662130	-78,640639	-1,661683348	-78,640935385	-1,661852	-78,640735
6	22/5/2023	21:00:00	-1,661954	-78,640822	-1,662285864	-78,640536609	-1,661656	-78,640878
7	23/5/2023	0:00:00	-1,661970	-78,640852	-1,662014895	-78,640937589	-1,661266	-78,640709
8	23/5/2023	1:00:00	-1,661971	-78,640840	-1,662014895	-78,640937589	-1,661784	-78,640933
9	23/5/2023	6:00:00	-1,662056	-78,640732	-1,662014895	-78,640937589	-1,661643	-78,640892
10	23/5/2023	9:00:00	-1,661965	-78,640815	-1,662285864	-78,640536609	-1,661663	-78,640844
11	23/5/2023	12:00:00	-1,661975	-78,640769	-1,661683348	-78,640935385	-1,661684	-78,640794
12	23/5/2023	15:00:00	-1,661875	-78,640737	-1,662285864	-78,640536609	-1,661535	-78,640748
13	23/5/2023	18:00:00	-1,661896	-78,640750	-1,661683348	-78,640935385	-1,661609	-78,640916
14	23/5/2023	21:00:00	-1,661898	-78,640809	-1,662285864	-78,640536609	-1,661765	-78,640929
15	24/5/2023	0:00:00	-1,661920	-78,640803	-1,662285864	-78,640536609	-1,661681	-78,640857
16	24/5/2023	1:00:00	-1,661914	-78,640813	-1,661683348	-78,640935385	-1,669514	-78,640801
17	24/5/2023	6:00:00	-1,661883	-78,640716	-1,661683348	-78,640935385	-1,661750	-78,640882
18	24/5/2023	9:00:00	-1,661960	-78,640813	-1,662285864	-78,640536609	-1,661728	-78,640893
19	24/5/2023	12:00:00	-1,661990	-78,640772	-1,661683348	-78,640935385	-1,661457	-78,640722
20	24/5/2023	15:00:00	-1,661960	-78,640788	-1,662285864	-78,640536609	-1,661965	-78,640800

21	24/5/2023	18:00:00	-1,661985	-78,640783	-1,661683348	-78,640935385	-1,661521	-78,640969
22	24/5/2023	21:00:00	-1,661977	-78,640772	-1,661683348	-78,640935385	-1,661529	-78,640907
23	25/5/2023	0:00:00	-1,662008	-78,640751	-1,662014895	-78,640937589	-1,661515	-78,640809
24	25/5/2023	1:00:00	-1,661874	-78,640745	-1,662014895	-78,640937589	-1,661536	-78,640838
25	25/5/2023	6:00:00	-1,662038	-78,640797	-1,662285864	-78,640536609	-1,661732	-78,640872
26	25/5/2023	9:00:00	-1,661978	-78,640820	-1,661683348	-78,640935385	-1,661617	-78,640943
27	25/5/2023	12:00:00	-1,662008	-78,640797	-1,662285864	-78,640536609	-1,661945	-78,640962
28	25/5/2023	15:00:00	-1,661881	-78,640739	-1,661683348	-78,640935385	-1,661656	-78,640793
29	25/5/2023	18:00:00	-1,663035	-78,640946	-1,661683348	-78,640935385	-1,661986	-78,640834
30	25/5/2023	21:00:00	-1,662066	-78,640792	-1,661683348	-78,640935385	-1,661979	-78,640853
Promedio:			-1,661997	-78,640790	-1,661959528	-78,640789534	-1,6619484	-78,640851
Varianza:			4,20E-08	2,89E-09	7,77E-08	3,83E-08	2,07E-06	5,22E-09
Desviación Estándar:			2,05E-04	5,38E-05	2,79E-04	1,96E-04	1,44E-03	7,23E-05
Coefficiente de Variación:			-1,23E-04	-6,84E-07	-1,68E-04	-2,49E-06	-8,66E-04	-9,19E-07

Realizado por: Guamán Lessly, López Kevin, 2023

Se observa que los valores de latitud y longitud proporcionados por los teléfonos smartphones y el módulo GPS en la ubicación 1 son similares, lo que indica que la ubicación proporcionada por el sistema es correcta. De igual manera se observa que tanto la varianza como la desviación estándar de los datos emitidos por el módulo GPS son cercanos a cero, demostrando una precisión en los datos de ubicación.

4.5.2 Ubicación 2

La ubicación 2 es en las calles Esmeraldas y Francia, a tres cuadras del Centro Comercial Popular–La Condamine, los datos obtenidos en esta prueba se encuentran en la Tabla 4-9.

Tabla 4-9: Datos emitidos por el prototipo en la ubicación 2

		MERCADO “CONDAMINE”						
		Android		Contenedor (GPS Neo 6M)		iOS		
	Fecha	Hora	Latitud	Longitud	Latitud	Longitud	Latitud	Longitud
1	18/5/2023	6:00:00	-1,671628	-78,657977	-1,671573518	-78,658065385	-1,671657	-78,657794
2	18/5/2023	9:00:00	-1,671681	-78,657979	-1,671987327	-78,657782162	-1,671798	-78,657628
3	18/5/2023	12:00:00	-1,671657	-78,657983	-1,671993742	-78,657709539	-1,671721	-78,65754
4	18/5/2023	15:00:00	-1,671643	-78,657989	-1,671993742	-78,657709539	-1,671675	-78,65767

5	18/5/2023	18:00:00	-1,671659	-78,657945	-1,671987327	-78,657782162	-1,671733	-78,657829
6	18/5/2023	21:00:00	-1,671775	-78,657883	-1,671987327	-78,657782162	-1,671669	-78,657573
7	19/5/2023	0:00:00	-1,671678	-78,657902	-1,671723753	-78,657597394	-1,671709	-78,657803
8	19/5/2023	1:00:00	-1,671811	-78,657785	-1,671723753	-78,657597394	-1,671859	-78,657832
9	19/5/2023	6:00:00	-1,671747	-78,657838	-1,671723753	-78,657597394	-1,67177	-78,657544
10	19/5/2023	9:00:00	-1,671698	-78,657919	-1,671573518	-78,658065385	-1,67183	-78,657729
11	19/5/2023	12:00:00	-1,671678	-78,657914	-1,671993742	-78,657709539	-1,67176	-78,657631
12	19/5/2023	15:00:00	-1,671534	-78,658063	-1,671987327	-78,657782162	-1,671776	-78,657782
13	19/5/2023	18:00:00	-1,671494	-78,658080	-1,671573518	-78,658065385	-1,671674	-78,657675
14	19/5/2023	21:00:00	-1,671665	-78,657931	-1,671993742	-78,657709539	-1,67167	-78,657615
15	20/5/2023	0:00:00	-1,671658	-78,657967	-1,671723753	-78,657597394	-1,671795	-78,657767
16	20/5/2023	1:00:00	-1,671625	-78,657906	-1,671723753	-78,657597394	-1,671705	-78,65779
17	20/5/2023	6:00:00	-1,671699	-78,657952	-1,671993742	-78,657709539	-1,671768	-78,657833
18	20/5/2023	9:00:00	-1,671638	-78,657988	-1,671573518	-78,658065385	-1,671748	-78,65782
19	20/5/2023	12:00:00	-1,671629	-78,657938	-1,671987327	-78,657782162	-1,67181	-78,657675
20	20/5/2023	15:00:00	-1,671693	-78,657812	-1,671987327	-78,657782162	-1,671691	-78,6578
21	20/5/2023	18:00:00	-1,671708	-78,657892	-1,671573518	-78,658065385	-1,671741	-78,657834
22	20/5/2023	21:00:00	-1,671722	-78,657910	-1,671987327	-78,657782162	-1,671654	-78,657731
23	21/5/2023	0:00:00	-1,671728	-78,657915	-1,671723753	-78,657597394	-1,671706	-78,657762
24	21/5/2023	1:00:00	-1,671731	-78,657915	-1,671573518	-78,658065385	-1,671772	-78,657851
25	21/5/2023	6:00:00	-1,671713	-78,657937	-1,671987327	-78,657782162	-1,671804	-78,657758
26	21/5/2023	9:00:00	-1,671729	-78,657940	-1,671993742	-78,657709539	-1,671667	-78,657584
27	21/5/2023	12:00:00	-1,671751	-78,657931	-1,671993742	-78,657709539	-1,671825	-78,657593
28	21/5/2023	15:00:00	-1,671766	-78,657906	-1,671573518	-78,658065385	-1,671799	-78,657747
29	21/5/2023	18:00:00	-1,671714	-78,657923	-1,671987327	-78,657782162	-1,671699	-78,657747
30	21/5/2023	21:00:00	-1,671747	-78,657906	-1,671987327	-78,657782162	-1,671854	-78,657813
Promedio:			-1,671687	-78,657931	-1,671839554	-78,657794348	-1,671745	-78,657725
Varianza:			4,42E-09	3,71E-09	3,31E-08	2,75E-08	3,83E-09	9,43E-09
Desviación Estándar:			6,65E-05	6,09E-05	1,82E-04	1,66E-04	6,19E-05	9,71E-05
Coefficiente de Variación:			-3,98E-05	-7,74E-07	-1,09E-04	-2,11E-06	-3,70E-05	-1,23E-06

Realizado por: Guamán Lessly, López Kevin, 2023

De la misma manera que en la prueba anterior, se observa que las ubicaciones proporcionadas por los teléfonos smartphones y el módulo GPS en la ubicación 2 son similares, lo que indica que los valores emitidos por el sistema son correctos. Así mismo, la varianza y la desviación estándar de los datos emitidos por el módulo GPS son cercanos a cero, demostrando precisión.

Observando los valores de la variación y desviación estándar se puede concluir que los datos emitidos por el módulo GPS son muy precisos pues se tienen valores cercanos a cero tanto en la ubicación 1 correspondiente al mercado “La Esperanza” como en la ubicación 2 del mercado “Condamine”, lo que significa que el lugar en donde se coloquen los contenedores no afecta en el envío y recepción de datos del sistema. En cuanto a los promedios latitud y longitud se puede observar que el módulo GPS NEO 6M del prototipo emite coordenadas similares a los dispositivos Smartphones, pero con un mayor número de decimales, por lo que su precisión es mayor.

4.6 Validación punto más cercano

El algoritmo de la interfaz gráfica remota determina que prototipo es el más cercano, esta función se activa cuando tanto el prototipo 1 como el prototipo 2 se encuentran llenos. Para corroborar que el basurero proporcionado por la interfaz gráfica remota es el más cercano se realizaron varias pruebas que consistieron en colocar los basureros llenos en diferentes lugares de la ciudad de Riobamba y con la utilización de Google Maps se trazó la ruta más corta para llegar a cada basurero, de esta manera se determinó cuál era el más cercano y si coincidía con el proporcionado por la interfaz gráfica.

4.6.1 Prueba 1

Tabla 4-10: Datos de la prueba 1 en la interfaz gráfica

INFORMACIÓN DE LA INTERFAZ GRÁFICA		
Punto de partida	Latitud	-1.6661581152835396
	Longitud	-78.66725877008285
Basurero 1	Latitud	-1.660695182783868
	Longitud	-78.67993610632683
Basurero 2	Latitud	-1.6699937191979612
	Longitud	-78.67558399823871
Punto más cercano	Basurero 2	

Realizado por: Guamán Lessly, López Kevin, 2023

En esta prueba se colocó el basurero 1 en la Avenida Pedro Vicente y Juan Romualdo Navarro, el basurero 2 en la calle Diego de Rodríguez y calle SN diagonal a la Hacienda Turística “Las Manolas” y el punto de partida fue entre la Avenida Pedro Vicente y Avenida 9 de Octubre en los Talleres Municipales del GAD RIOBAMBA, cuyos valores de latitud y longitud se encuentran en la Tabla 4-10.



Ilustración 4-14: Prueba 1 del punto más cercano.

Realizado por: Guamán Lessly, López Kevin, 2023

La interfaz gráfica en la prueba 1 se muestra en la ilustración 4-14, en donde marca como el contenedor lleno más cercano al basurero 2. Para corroborar esta información con la API de Google Maps denominado Directions se trazó la ruta más eficiente en vehículo desde el punto de partida, los Talleres Municipales del GADM Riobamba hacia el basurero 1 y basurero 2 como se muestran en la ilustración 4-15 e ilustración 4-16 respectivamente.

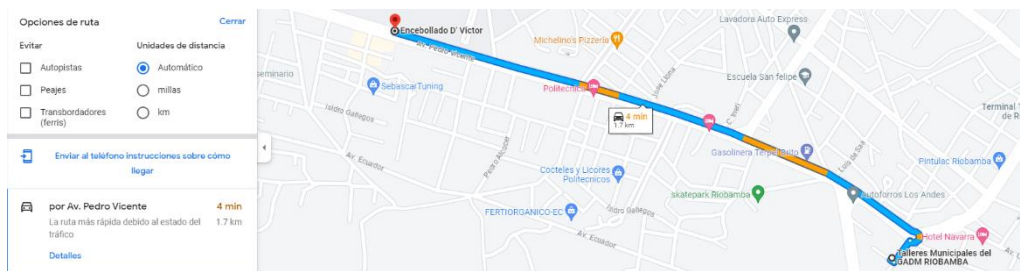


Ilustración 4-15: Ruta más corta al basurero 1 en la prueba 1.

Fuente: (Google Maps, 2023)

La ruta más corta entre el punto de partida y el basurero 1 se muestra en la ilustración 4-15, la distancia a recorrer es de 1,7 km.

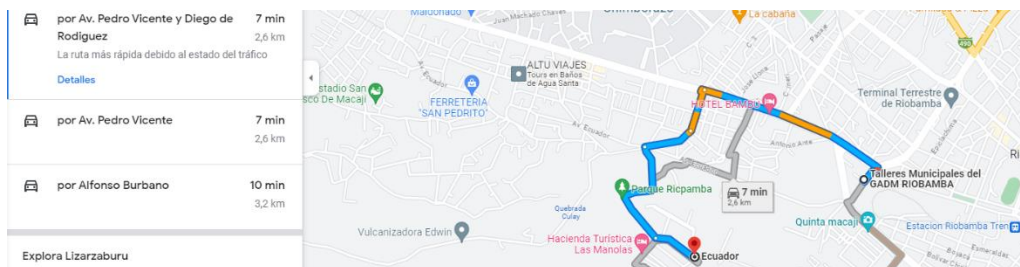


Ilustración 4-16: Ruta más corta al basurero 2 en la prueba 1.

Fuente: (Google Maps, 2023)

La ruta más corta entre el punto de partida y el basurero 2 se muestra en la ilustración 4-16, la distancia a recorrer es de 2,6 km

Tabla 4-11: Datos de la ruta más corta en la prueba 1

INFORMACIÓN DE GOOGLE MAPS	
Distancia basurero 1	1,7 km
Distancia basurero 2	2,6 km
Punto más cercano	Basurero 1

Realizado por: Guamán Lessly, López Kevin, 2023

Según la información proporcionada por la API de Google Maps que se encuentra resumida en la Tabla 4-11, el basurero más cercano es el 1 que se encuentra a una distancia de 1,7 km, este punto no coincide con el proporcionado por la interfaz gráfica remota que marca como el punto más cercano al basurero 2. Por lo tanto, el resultado de esta prueba es desfavorable debido a que la ubicación seleccionada para el basurero 2 se encuentra en un punto donde no existe una calle directa desde el punto de partida ya que se encuentra la Quinta Macají.

4.6.2 Prueba 2

Tabla 4-12: Datos de la prueba 2 en la interfaz gráfica

INFORMACIÓN DE LA INTERFAZ GRÁFICA		
Punto de partida	Latitud	-1.6661581152835396
	Longitud	-78.66725877008285
Basurero 1	Latitud	-1.6840781495193413
	Longitud	-78.65310047376244
Basurero 2	Latitud	-1.6508202901972668
	Longitud	-78.6597940997395
Punto más cercano	Basurero 2	

Realizado por: Guamán Lessly, López Kevin, 2023

En la prueba 2 se colocó el basurero 1 en la Avenida 9 de Octubre entre Antonio Flor y Alvarado, el basurero 2 en la Prolongación de la Avenida Gonzalo Dávalos entre José María Egas y la calle adyacente al canal de riego y el punto de partida fue entre la Avenida Pedro Vicente y Avenida 9 de Octubre en los Talleres Municipales del GAD RIOBAMBA, cuyos valores de latitud y longitud se encuentran en la Tabla 4-12.

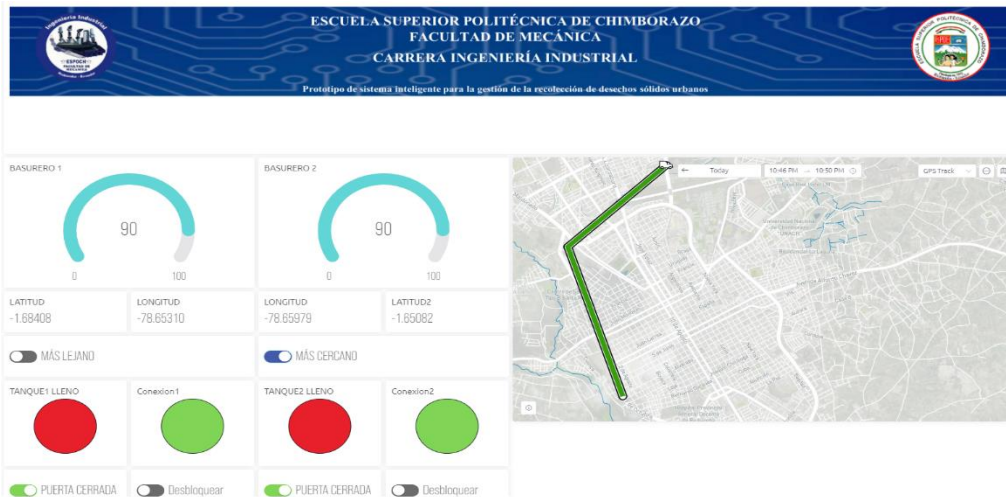


Ilustración 4-17: Prueba 2 del punto más cercano.

Realizado por: Guamán Lessly, López Kevin, 2023

La interfaz gráfica en la prueba 2 se muestra en la ilustración 4-17, en donde señala como el contenedor lleno más cercano al basurero 2. Para verificar que esta información es correcta, se trazó la ruta más eficiente con la API de Google Maps denominado Directions desde el punto de partida, los Talleres Municipales del GADM Riobamba hacia el basurero 1 y basurero 2 como se muestran en la ilustración 4-18 e ilustración 4-19 respectivamente.

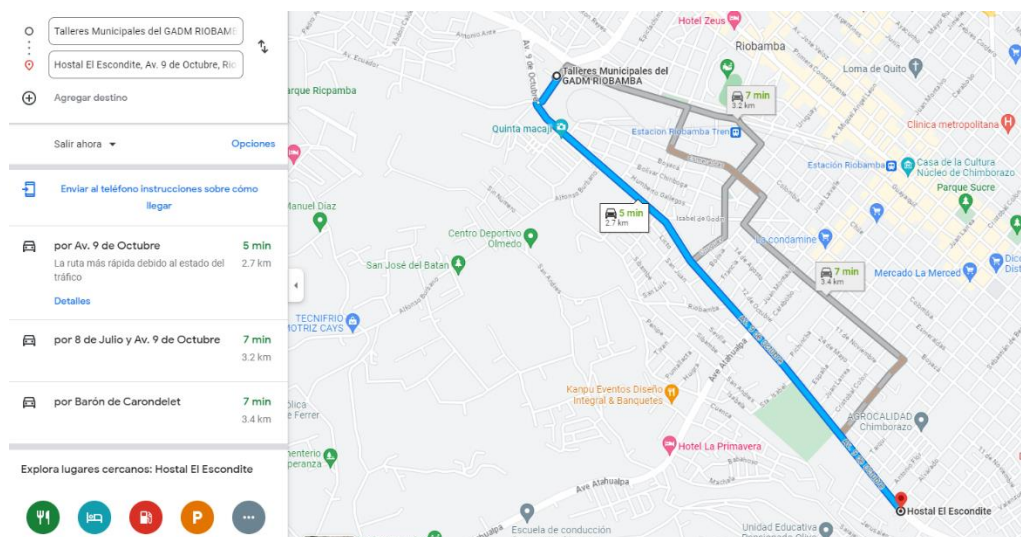


Ilustración 4-18: Ruta más corta al basurero 1 en la prueba 2.

Fuente: (Google Maps, 2023)

La ruta más corta entre el punto de partida y el basurero 1 en la prueba 2 se muestra en la ilustración 4-18, la distancia a recorrer es de 2,7 km.

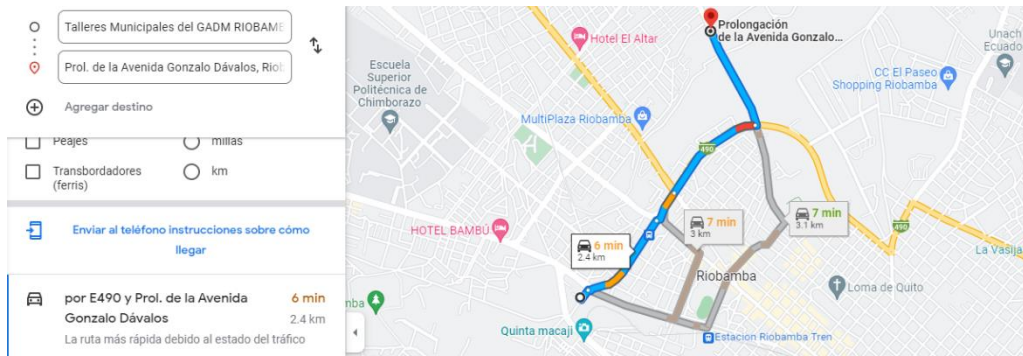


Ilustración 4-19: Ruta más corta al basurero 2 en la prueba 2.

Fuente: (Google Maps, 2023)

La ruta más corta entre el punto de partida y el basurero 2 en la prueba 2 se muestra en la ilustración 4-19, la distancia a recorrer es de 2,4 km.

Tabla 4-13: Datos de la ruta más corta en la prueba 2

INFORMACIÓN DE GOOGLE MAPS	
Distancia basurero 1	2,7 km
Distancia basurero 2	2,4 km
Punto más cercano	Basurero 2

Realizado por: Guamán Lessly, López Kevin, 2023

Las distancias entre cada basurero y el punto de partida se muestran en la Tabla 4-13, observando los valores se determina que el basurero 2 es el más cercano que se encuentra a 2,4 km, este punto coincide con el proporcionado con la interfaz gráfica remota. El resultado de esta prueba es favorable debido a que las ubicaciones para ambos prototipos fueron en zonas residenciales de la ciudad de Riobamba a la que se puede acceder por múltiples rutas.

4.6.3 Prueba 3

Tabla 4-14: Datos de la prueba 3 en la interfaz gráfica

INFORMACIÓN DE LA INTERFAZ GRÁFICA		
Punto de partida	Latitud	-1.6661581152835396
	Longitud	-78.66725877008285
Basurero 1	Latitud	-1.6736415964520457
	Longitud	-78.65346666258841
Basurero 2	Latitud	-1.6690136604535797
	Longitud	-78.64962104144489
Punto más cercano	Basurero 1	

Realizado por: Guamán Lessly, López Kevin, 2023

En la prueba 3 se colocó el basurero 1 en las calles Chile Y García Moreno, el basurero 2 en la Argentinos y García Moreno y el punto de partida al igual que en las pruebas anteriores fue desde los Talleres Municipales del GAD RIOBAMBA, cuyos valores de latitud y longitud se encuentran en la Tabla 4-14.



Ilustración 4-20: Prueba 3 del punto más cercano.

Realizado por: Guamán Lessly, López Kevin, 2023

La interfaz gráfica en la prueba 3 se muestra en la ilustración 4-20, en donde marca como el contenedor lleno más cercano al basurero 1. Para corroborar esta información con la API de Google Maps denominado Directions se trazó la ruta más eficiente en vehículo desde el punto de partida, los Talleres Municipales del GADM Riobamba hacia el basurero 1 y basurero 2 como se muestran en la ilustración 4-21 e ilustración 4-22 respectivamente.

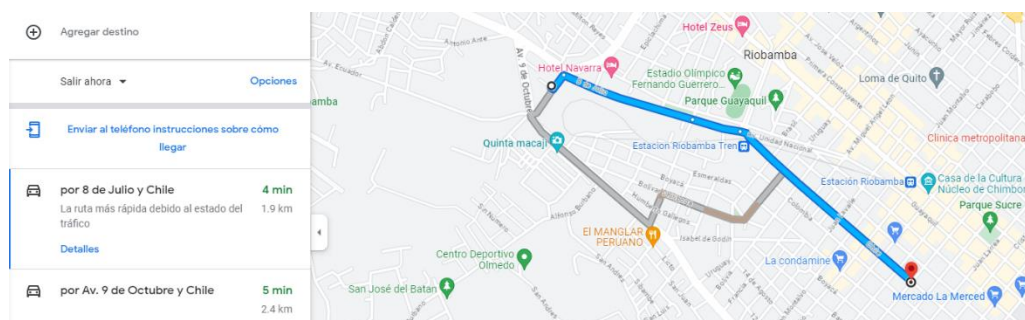


Ilustración 4-21: Ruta más corta al basurero 1 en la prueba 3.

Fuente: (Google Maps, 2023)

La ruta más corta entre el punto de partida y el basurero 1 en la prueba 3 se muestra en la ilustración 4-21, la distancia a recorrer es de 1,9 km.

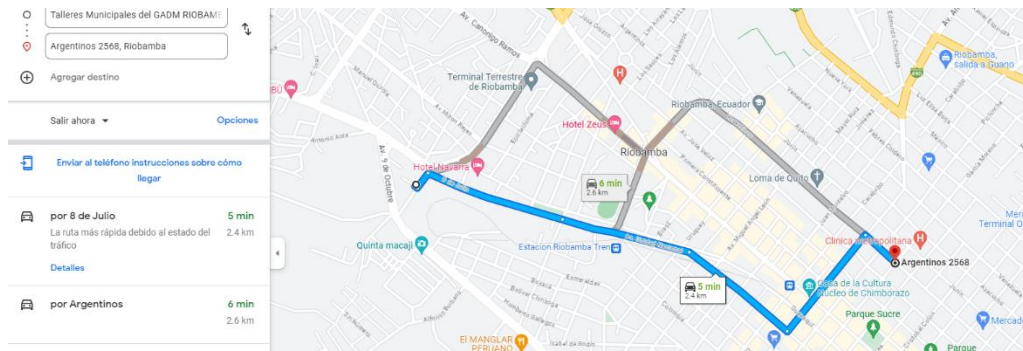


Ilustración 4-22: Ruta más corta al basurero 2 en la prueba 3.

Fuente: (Google Maps, 2023)

La ruta más corta entre el punto de partida y el basurero 2 en la prueba 3 se muestra en la ilustración 4-22, la distancia a recorrer es de 2,4 km.

Tabla 4-15: Datos de la ruta más corta en la prueba 3

INFORMACIÓN DE GOOGLE MAPS	
Distancia basurero 1	1,9 km
Distancia basurero 2	2,4 km
Punto más cercano	Basurero 1

Realizado por: Guamán Lessly, López Kevin, 2023

Según las distancias proporcionadas por la API de Google Maps que se encuentran en la Tabla 4-15, el basurero 1 es el más cercano que se encuentra a una distancia de 1,9 km, este punto coincide con el indicado por la interfaz gráfica remota. Por lo tanto, el resultado de esta prueba es favorable debido a que las locaciones para ambos basureros fueron en zonas céntricas en donde la distribución de las cuerdas es homogénea y son de fácil acceso.

4.7 Validación del sistema de bloqueo de la puerta

En la tabla 4-16 se muestra la toma del tiempo de latencia desde que la interfaz gráfica remota indica que el contenedor se ha llenado hasta la activación del servomotor que bloquea la puerta.

Tabla 4-16: Tiempos de respuesta del sistema de bloqueo

PRUEBA	TIEMPO (s)
1	29,14
2	27,04
3	28,59
4	27,62

5	27,72
6	28,02
7	30,01
8	26,95
9	29,68
10	26,97
11	28,67
12	28,79
13	28,75
14	27,29
15	29,68
16	27,23
17	27,57
18	27,89
19	29,28
20	28,53
21	28,95
22	27,03
23	26,99
24	28,75
25	27,58
Promedio	28,19
Máximo	30,01
Mínimo	26,95
Varianza	0,93
Desv. Estándar	0,96

Realizado por: Guamán Lessly, López Kevin, 2023

El tiempo promedio de activación del servomotor es de 28,19 segundos, un valor bastante aceptable considerando que no es urgente el bloqueo de la puerta, además la desviación estándar de los datos es menor a 1 lo que indica que los valores tomados no varían excesivamente entre ellos.

4.8 Validación del sistema de alarma

Se planteó como prueba medir el tiempo que tarda el sensor infrarrojo en detectar que la puerta se está abriendo. En la tabla 4-17 se muestra los datos de la toma de tiempos de esta prueba.

Tabla 4-17: Tiempos de respuesta del sistema de alarma

PRUEBA	TIEMPO (s)
1	25,03
2	17,12

3	24,34
4	25,26
5	18,84
6	17,37
7	21,2
8	19,93
9	21,22
10	18,48
11	18,37
12	18,11
13	22,75
14	15,46
15	22,28
16	18,04
17	15,83
18	22,99
19	15,09
20	23,32
21	21,84
22	18,43
23	19,58
24	23,43
25	18,47
Promedio	20,11
Máximo	25,26
Mínimo	15,09
Varianza	9,10
Desv. Estándar	3,02

Realizado por: Guamán Lessly, López Kevin, 2023

Del análisis de la tabla se puede determinar que el tiempo promedio de activación de la alerta de voz es de 20,11 segundos, se observa que la desviación estándar de los datos es bastante alta de 3,02 segundos esto se debe a que el tiempo que la persona presiona la puerta intentando acceder al basurero es variado.

4.9 Costos

En la tabla 4-18 se detallan los costos de los componentes y materiales empleados para la construcción del prototipo, mientras que en la tabla 4-19 se muestran los costos generados por actividades ajenas al proceso de fabricación del prototipo.

Tabla 4-18: Costos directos del prototipo

CANTIDAD	DESCRIPCIÓN	V. UNITARIO	V. TOTAL
3	Angulo doblado 20x2 mm	7,00	21,00
2	Tol galvanizado 0.9 mm	22,00	44,00
1	Platina 25x3 mm	6,00	6,00
4	Bisagras	0,25	1,00
55	Remaches	0,05	2,75
7	Electrodos 6013	0,20	1,40
1	Contenedor de basura	30,00	30,00
1	Pegatinas	15,00	15,00
1	Cámara web	10,00	10,00
1	Modulo GPS NEO 6M	12,00	12,00
1	Servomotor MG996R	8,00	8,00
1	Sensor infrarrojo FC-51	1,00	1,00
1	Parlante	5,00	5,00
1	Módulo amplificador de audio TDA2030A	1,25	1,25
1	Raspberry Pi 4 Model B	220,00	220,00
1	Batería 12 V	20,00	20,00
2	Reguladores de voltaje	3,00	6,00
1	Foco led de 12V	0,80	0,80
1	DFplayer mini reproductor mp3	3,50	3,50
TOTAL			\$ 408,7

Realizado por: Guamán Lessly, López Kevin, 2023

Tabla 4-19: Costos indirectos del prototipo

DESCRIPCIÓN	V. TOTAL
Transporte	20,00
Imprevisto	10,00
TOTAL	\$ 30,00

Realizado por: Guamán Lessly, López Kevin, 2023

La tabla 4-20 expone el costo total de la obtención del prototipo, que incluye tanto los costos directos como indirectos.

Tabla 4-20: Costos totales del prototipo

DESCRIPCIÓN	V. TOTAL
Costos directos	408,70
Costos indirectos	30,00
TOTAL	\$ 438,70

Realizado por: Guamán Lessly, López Kevin, 2023

Como se elaboraron dos prototipos, el costo total para la culminación del presente trabajo de titulación es de \$847,40.

CONCLUSIONES

- Se desarrolló un sistema inteligente basado en comunicación inalámbrica que mediante visión artificial detecta el nivel de llenado de los contenedores de basura, esta información se expone, registra y almacena en una interfaz gráfica remota que le permite al usuario monitorear los prototipos y controlar el acceso a los mismos. También de forma local los prototipos se bloquean cuando alcanzan un nivel de llenado máximo y emiten una alerta en caso de intento de acceso.
- Se seleccionó como hardware de procesamiento la Raspberry Pi 4 Model B ya que permite la comunicación mediante WiFi, cuenta con múltiples entradas de periféricos y es capaz de albergar un algoritmo de visión artificial, además es compatible con el lenguaje de programación Python que fue el seleccionado para el sistema debido a que cuenta con librerías que permiten la codificación de un algoritmo de visión artificial.
- Se determinó que el servomotor MG996R tarda un tiempo promedio de 28,19 segundos en activarse luego de que la interfaz gráfica remota indica que se ha alcanzado un nivel de llenado máximo, por su parte el sistema de alerta de voz tarda un tiempo promedio de 20,11 segundos en activarse después de que el sensor infrarrojo FC-51 detecta que se está intentando acceder a un contenedor bloqueado.
- Se implementó una red inalámbrica de sensores y actuadores con topología tipo estrella que se vinculan al sistema inteligente mediante conexión Wi-Fi, posibilitando el envío y recepción de información entre la interfaz gráfica remota y los prototipos a través de la nube.
- Se creó una interfaz gráfica remota en la plataforma Blynk IoT que monitorea el nivel de llenado de los contenedores de basura, su ubicación e información de estado, además le permite al usuario controlar el acceso a los prototipos.
- Se programó un algoritmo en la interfaz gráfica remota que determina cuál de los contenedores llenos es el más cercano, esta función es útil si se utiliza en zonas residenciales o céntricas en donde la distribución de cuadras es homogénea, ya que si se emplea en lugares aledaños a parques, haciendas o sitios donde no existan calles que atraviesen el lugar, el punto indicado como más cercano por la interfaz gráfica remota es erróneo.
- Se determinó que el sistema de visión artificial tiene una eficiencia óptima del 86,36% si se depositan en el prototipo desechos en fundas plásticas de color negro, esta eficiencia se ve afectada según los desperdicios que se coloquen, en el caso de desechos de colores claros la eficiencia obtenida es del 78,65% y en caso de depositarse desechos que incluyan letras, números o figuras la eficiencia resultante del sistema es de tan solo el 68,75%.

RECOMENDACIONES

- Mantener constantes factores externos como el tipo de iluminación y su ángulo de incidencia, para no influir en la calidad de imagen que la visión artificial debe procesar.
- Se recomienda utilizar el prototipo con desechos sólidos uniformes a fin de evitar que se cubra por completo la escala de medición y que el sistema detecte un nivel de llenado erróneo.
- Analizar otras opciones para la escala de medición del nivel de llenado de basura, como es el uso de colores, que permitan mejorar la eficiencia del algoritmo de visión artificial.
- Se recomienda que en el prototipo se depositen desechos sólidos en fundas de color negro u oscuros que no contengan números, letra o figuras, pues la eficiencia del sistema es alta con este tipo de desechos.
- Se sugiere reemplazar las baterías convencionales que requieren ser cargadas, por la implementación de un sistema fotovoltaico basado en paneles solares, para hacer de los prototipos autosustentables.
- Para utilizar el prototipo se recomienda tener una conexión a internet estable que no afecte en el tiempo de respuesta del sistema.
- Se recomienda la implementación de un módem USB en los prototipos para asegurar la conectividad al sistema incluso en áreas donde no existan redes WiFi disponibles.

BIBLIOGRAFÍA

- 1. ADAFRUIT INDUSTRIES.** *Adafruit IO* [blog]. 2022. [Consulta: 22 octubre 2022]. Disponible en: <https://io.adafruit.com/>
- 2. ADELL, Jordi; & BERNABÉ, Iolanda.** "Software libre en educación". *Tecnología educativa* [en línea], 2007, (España), pp 173-195. [Consulta: 14 octubre 2022]. Disponible en: https://www.researchgate.net/profile/Jordi-Adell/publication/216393192_Software_libre_en_educacion/links/0912f51366175dd62d000000/Software-libre-en-educacion.pdf
- 3. ALLOZA, Jesús.** *Montaje de componentes y periféricos microinformáticos* [en línea]. IC Editorial, 2021. [Consulta: 5 diciembre 2022]. Disponible en: <https://books.google.es/books?hl=es&lr=&id=435KEAAAQBAJ&oi=fnd&pg=PT6&dq=montaje+de+componentes+y+perifericos+microinformaticos&ots=DPb0Avz5I-&sig=BikuIwIwczBUBi6CP7dQsajDCE#v=onepage&q=montaje%20de%20componentes%20y%20perifericos%20microinformaticos&f=f>
- 4. AMAZON.** *Power Bank 30000mAh Batería Externa Cargador Móvil Portátil* [blog]. 2022. [Consulta: 05 diciembre 2022]. Disponible en: <https://www.amazon.es/30000mAh-Cargador-Port%C3%A1til-Capacidad-Smartphones/dp/B089136WW8?th=>
- 5. ARDUINO.** *What is Arduino?* [blog]. 2022. [Consulta: 10 noviembre 2022]. Disponible en: <https://www.arduino.cc/en/Guide/Introduction>.
- 6. ASAMBLEA NACIONAL DE LA REPÚBLICA DEL ECUADOR.** *Constitución de la República del Ecuador* [en línea]. Montecristi-Ecuador, 2008. [Consulta: 12 octubre 2022]. Disponible en: https://www.asambleanacional.gob.ec/sites/default/files/documents/old/constitucion_de_bolsillo.pdf
- 7. ASC ELECTRÓNICA S.A.** *Batería 12V/7Ah* [blog]. 2022. [Consulta: 12 diciembre 2022]. Disponible en: <https://magomelectronica.com/asc/producto/bateria-12v-7ah/>
- 8. BARRÓN, Mariano.** *Uso didáctico del software de ayuda al diseño electrónico "PROTEUS"* [en línea]. País Vasco: Departamento de Ingeniería de Sistemas y Automática, 2000. [Consulta: 27 octubre 2022]. Disponible en: <http://e-spacio.uned.es/fez/eserv/taee:congreso-2004-1034/S1F05.pdf>
- 9. BLYNK INTERNATIONAL.** *Plataforma Blynk IoT: para empresas y desarrolladores* [blog]. 2023. [Consulta: 23 octubre 2022]. Disponible en: <https://blynk.io/>.
- 10. BNAMERICAS.** *Los planes de Ecuador para promover la transformación digital y la industria 4.0.* [blog]. 2021. [Consulta: 02 octubre 2022]. Disponible en:

<https://www.bnamericas.com/es/entrevistas/los-planes-de-ecuador-para-promover-la-transformacion-digital-y-la-industria-40>.

11. BOADI, Kwasi Owusu; & KUITUNEN, Markku. "Municipal solid waste management in the Accra Metropolitan Area, Ghana". *The Environmentalist* [en línea], 2003, 23, pp. 211-218. [Consulta: 2 octubre 2022]. Disponible en: <https://doi.org/10.1023/B:ENVR.0000017283.09117.20>

12. BOROZDUKHIN, Andrei; et al. "Approach to the garbage collection in the "Smart Clean City" project". *IEEE* [en línea], 2016, pp. 918-922. [Consulta: 5 octubre 2022]. Disponible en: <https://ieeexplore.ieee.org/document/7805019>

13. BURGOS, José; & PRIETO, Antonio. "Aplicación de las TIG para la optimización de los servicios de recogida y gestión de los residuos sólidos urbanos del municipio de Aspe (Alicante)". *Asociación de Geógrafos Españoles* [en línea], 2014, (España), pp. 807-813. [Consulta: 28 septiembre 2022]. ISBN 978-84-940784-4-6. Disponible en: <http://hdl.handle.net/10045/46761>

14. C&D TECHNOLOGIA. *Parlante 8 Ohms 10W* [blog]. 2023. [Consulta: 10 diciembre 2022]. Disponible en: <https://cdtecnologia.net/otros/861-parlante-8-ohms-10w.html>.

15. CAMPO, Diego; et al. "Diseño de un prototipo para evaluar el color y matiz del café tostado y su relación con sus propiedades organolépticas". *Institución Universitaria Tecnológica de Comfacauca* [en línea], 2018. [Consulta: 14 octubre 2022]. Disponible en: https://www.researchgate.net/publication/332653745_Color_Index_System_for_Coffee

16. CHALLENGER, Ivet; et al. "El lenguaje de programación Python". *Centro de Información y Gestión Tecnológica de Santiago de Cuba* [en línea], 2014, (Cuba) 20(2), pp. 1-13. [Consulta: 3 noviembre 2022]. ISSN 1027-2127. Disponible en: <https://www.redalyc.org/pdf/1815/181531232001.pdf>

17. CNC. *Informe sobre mapeo de actores generadores de información a nivel territorial e identificación de fuentes de información de la competencia de desechos sólidos* [en línea]. Quito-Ecuador: Consejo Nacional de Competencias, 2019. [Consulta: 10 octubre 2022]. Disponible en: <http://www2.competencias.gob.ec/wp-content/uploads/2021/03/03-Manejo-desechos-solidos-2.pdf>

18. COMPOLOGY. *About Compology* [blog]. 2022. [Consulta: 06 octubre 2022]. Disponible en: <https://www.crunchbase.com/organization/compology>.

19. CORONA, Leonel; et al. *Sensores y Actuadores* [en línea]. México D.F.-México: Grupo Editorial Patria, 2014. [Consulta: 19 octubre 2022]. Disponible en: https://books.google.es/books?hl=es&lr=&id=wMm3BgAAQBAJ&oi=fnd&pg=PP1&dq=actuadores+eléctricos&ots=6O9lix901B&sig=ynoRRI_mhmaqsgHv1LTgc2MC2eI#v=onepage&q=actuadores%20eléctricos&f=false

20. DASSAULT SYSTEMES. *SolidWorks* [blog]. 2022. [Consulta: 18 octubre 2022]. Disponible en: <https://www.solidworks.com/es>.

- 21. DE MORI, Paulo; et al.** "Urban solid waste management in Caxias do Sul/Brazil: practices and challenges". *Journal of Urban and Environmental Engineering* [en línea], 2014, (Brasil) 10(1), pp. 50-56. [Consulta: 18 octubre 2022]. ISSN 1982-3932. Disponible en: <https://doi.org/10.4090/juee.2016.v10n1.50-56>
- 22. DESONGLES, Juan; et al.** *Técnicos de Soporte Informático de la Comunidad de Castilla Y Leon* [en línea]. Sevilla-España: MAD, 2006. [Consulta: 11 octubre 2022]. Disponible en: https://books.google.es/books?id=SUjFswQk1_4C&pg=PA312&dq=lenguaje++programaci%C3%B3n+es&hl=es&sa=X&ved=0ahUKEwiq6rWskdjlAhUJkhQKHWzbCGQQ6AEINjAC#v=onepage&q=lenguaje%20%20programaci%C3%B3n%20es&f=false
- 23. ECUBE LABS.** *Soluciones inteligentes de gestión de residuos* [blog]. 2022. [Consulta: 03 octubre 2022]. Disponible en: <https://www.ecubelabs.com/es/>.
- 24. ELECFreaks.** *Ultrasonic Ranging Module HC - SR04*. 2020 [blog]. 2022. [Consulta: 11 diciembre 2022]. Disponible en: <https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf>
- 25. ELECTRONICLAB.** *Motor Paso A Paso Nema 17*. 2022 [blog]. 2022. [Consulta: 12 diciembre 2022]. Disponible en: <https://electronilab.co/tienda/motor-paso-a-paso-nema-17-25-oz-in-200-pasosvuelta/>
- 26. EMAC.** *Recolección* [blog]. 2019. [Consulta: 16 octubre 2022]. Disponible en: <https://emac.gob.ec/servicios/recoleccion/>.
- 27. ENDARA, Clelio; & MAIGUA, Emerson.** Desarrollo de un algoritmo de trayectoria para un robot seguidor de línea destreza de competencia mediante visión e inteligencia artificial [en línea] (Trabajo de titulación) (Ingeniería). Universidad Politécnica Salesiana, Quito -Ecuador. 2021. [Consulta: 10 octubre 2022]. Disponible en: <http://dspace.ups.edu.ec/handle/123456789/19880>
- 28. ENERGIZER.** *Duracell* [blog]. 2021. [Consulta: 13 noviembre 2022]. Disponible en: <https://www.amazon.es/Pilas-Duracell-Recargables-Ultra-paquete/dp/B0002FQXJK>
- 29. ENERGIZER.** *ENERGIZER 522* [blog]. 2021. [Consulta: 13 noviembre 2022]. Disponible en: <https://www.electronicoscaldas.com/datasheet/522-Energizer.pdf>
- 30. ESCALANTE, David; & VARGAS, Daniel.** *Raspberry pi: la tecnología reducida en placa* [blog]. 2019. [Consulta: 08 noviembre 2022]. Disponible en: <https://repository.usc.edu.co/bitstream/handle/20.500.12421/4250/RASPBERRY%20PI.pdf?sequence=3&isAllowed=y>
- 31. FRAILE, Juan.** Introducción al control remoto de servomotores industriales [en línea] (Trabajo Fin de Máster) (Maestría). Universidad de Sevilla, Sevilla-España. 2021. [Consulta: 09 octubre 2022]. Disponible en: <https://hdl.handle.net/11441/127920>

- 32. FUJDIAK, Radek; et al.** "Using genetic algorithm for advanced municipal waste collection in Smart City". IEEE [en línea], 2016, (Chequia), pp. 1-6 [Consulta: 20 octubre 2022]. Disponible en: <https://ieeexplore.ieee.org/document/7574016>
- 33. GARCÍA, David.** Comparativa entre xips wifi de baix cost: esp32 vs esp8266. Aplicació a una xarxa domòtica [en línea] (Trabajo fin de grado) (Ingeniería). Universitat Politècnica de Catalunya, Barcelona-España. 2021. [Consulta: 14 octubre 2022]. Disponible en: https://upcommons.upc.edu/bitstream/handle/2117/350947/TFG_Garc%c3%ada_David.pdf?sequence=1&isAllowed=y
- 34. GARCÍA, Francisco.** Sistema de Gestión de Residuos con Contenedores Inteligentes [en línea] (Tesis) (Licenciatura). Universidad Empresarial Siglo 21, Córdoba-Argentina. 2021. [Consulta: 14 octubre 2022]. Disponible en: <https://repositorio.uesiglo21.edu.ar/handle/ues21/21556>
- 35. GILLIS, Alexander.** *Web Server* [blog]. 2020. [Consulta: 01 noviembre 2022]. Disponible en: <https://www.techtarget.com/whatis/definition/Web-server>.
- 36. GÓMEZ, Marcello; et al.** "ThingSpeak – an API and Web Service for the Internet of Things". World Wide Web [en línea], 2014, 25, pp. 1-4 [Consulta: 24 octubre 2022]. Disponible en: https://staas.home.xs4all.nl/t/swtr/documents/wt2014_thingspeak.pdf
- 37. GOOGLE MAPS.** *Google Maps* [en línea]. 2023. [Consulta: 28 enero 2023]. Disponible en: <https://www.google.com/maps/place/1%C2%B039'43.4%22S+78%C2%B038'28.0%22W/@-1.6622648,-78.6419999,18z/data=!4m4!3m3!8m2!3d-1.662061!4d-78.641104?hl=es>.
- 38. GOOGLE MAPS.** *Google Maps* [en línea]. 2023. [Consulta: 28 enero 2023]. Disponible en: <https://www.google.com/maps/place/1%C2%B040'18.3%22S+78%C2%B039'28.9%22W/@-1.6723389,-78.6577248,18z/data=!4m4!3m3!8m2!3d-1.671754!4d-78.658031?hl=es>.
- 39. GRUESO, Jhon.** Implementación de una interfaz HMI para la operación de módulo control de procesos del laboratorio de robótica [en línea] (Trabajo de grado) (Ingeniería). Universidad de la Salle, Bogotá-Colombia. 2008. [Consulta: 09 octubre 2022]. Disponible en: https://ciencia.lasalle.edu.co/cgi/viewcontent.cgi?article=1054&context=ing_automatizacion
- 40. GUERRERO, Jehicob.** Utilización de contenedores inteligentes como alternativa para la recolección de residuos domiciliarios [en línea] (Tesis) (Ingeniería). Universidad de Guayaquil, Guayaquil-Ecuador. 2019. [Consulta: 13 octubre 2022]. Disponible en: <http://repositorio.ug.edu.ec/handle/redug/40607>
- 41. HERTZ, Daniel.** *How to Use a GPS Receiver With Raspberry Pi 4* [blog]. 2020. [Consulta: 01 diciembre 2022]. Disponible en: <https://maker.pro/raspberry-pi/tutorial/how-to-use-a-gps-receiver-with-raspberry-pi-4>.
- 42. HIKVISION DIGITAL TECHNOLOGY.** *DS-U02 2 MP Web Camera* [blog]. 2020. [Consulta: 17 noviembre 2022]. Disponible en:

- https://www.hikvision.com/content/dam/hikvision/products/S000000001/S000000132/S000000146/S000004538/OFR006820/M000043043/Data_Sheet/Datasheet-of-DS-U02_20220906.pdf
- 43. IGUAL, Raúl; & MEDRANO, Carlos.** *Tutorial de OpenCV* [blog]. 2008. [Consulta: 17 noviembre 2022]. Disponible en: https://josbram.delifrut.cl/files/openCV/tutorial_opencv.pdf
- 44. INFINEON.** *Servomotor MG996R* [blog]. 2016. [Consulta: 14 diciembre 2022]. Disponible en: <https://pdf1.alldatasheet.com/datasheet-pdf/view/1131873/ETC2/MG996R.html>
- 45. INSTITUTO NACIONAL DE ESTADÍSTICAS Y CENSOS; et al.** *Estadística de Información Ambiental Económica en Gobiernos Autónomos Descentralizados Municipales* [en línea]. Quito-Ecuador, 2021. [Consulta: 09 octubre 2022]. Disponible en: https://www.ecuadorencifras.gob.ec/documentos/web-inec/Encuestas_Ambientales/Municipios_2020/Residuos_solidos_2020/Boletin_Tecnico_Residuos_2020.pdf
- 46. JÁCOME, Diego.** Manejo de un GPS con NIOS II [en línea] (Tesina) (Ingeniería). Escuela Superior Del Litoral, Guayaquil-Ecuador. 2012. [Consulta: 02 diciembre 2022]. Disponible en: https://www.dspace.espol.edu.ec/bitstream/123456789/21573/1/Manejo%20de%20un%20GPS%20con%20NIOS%20II%20_0.pdf
- 47. JIMÉNEZ, Nancy.** "La gestión integral de residuos sólidos urbanos en México: entre la intención y la realidad". *Revista Latinoamericana De Estudios Socioambientales* [en línea], 2015, (México) (17), pp. 29-56. [Consulta: 13 octubre 2022]. Disponible en: <https://revistas.flacsoandes.edu.ec/letrasverdes/article/view/1419>
- 48. LIGO, George.** *Servo Motor* [blog]. 2020. [Consulta: 07 noviembre 2022]. Disponible en: <https://electrosome.com/servo-motor/>.
- 49. MAROTTA, Raquel; et al.** "Municipal solid waste in Brazil: A review". *Waste Management & Research* [en línea], 2017, (Brasil) 35(12), pp. 1195-1209. [Consulta: 20 octubre 2022]. Disponible en: <https://doi.org/10.1177/0734242X17735375>
- 50. MATHWORKS.** *ThingSpeak* [blog]. 2023. [Consulta: 22 octubre 2022]. Disponible en: <https://thingspeak.com/>.
- 51. MINISTERIO DEL AMBIENTE, AGUA Y TRANSICIÓN ECOLÓGICA.** *Programa Nacional para la Gestión Integral de Desechos Sólidos – PNGIDS ECUADOR* [en línea]. 2020. [Consulta: 11 octubre 2022]. Disponible en: <https://www.ambiente.gob.ec/programa-pngids-ecuador/>.
- 52. MOISÉS, Andrés.** *Internet de las cosas* [en línea]. Madrid-España: REUS, 2020. [Consulta: 14 octubre 2022]. Disponible en: <https://books.google.es/books?hl=es&lr=&id=0BE5EAAAQBAJ&oi=fnd&pg=PA9&dq=internet+de+las+cosas&ots=JK82QSzVF3&sig=vHmHjP-j9cP1oQDHa9UsFKamM1I#v=onepage&q=internet%20de%20las%20cosas&f=false>

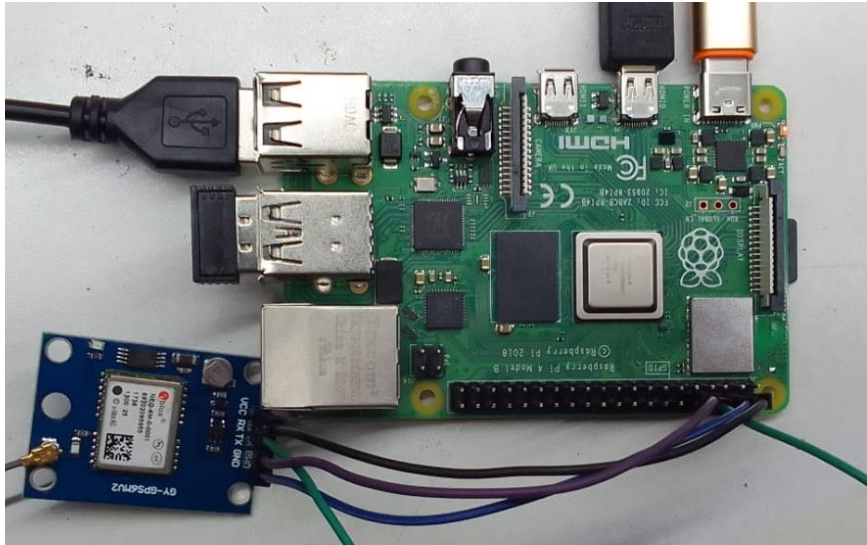
- 53. MOROCHO, Vanessa; et al.** "Sistema multimodal que permita monitorear y controlar la recolección de desechos sólidos en tiempo real para Smart Cities". *Revista de Tecnologías de la Informática y las Comunicaciones* [en línea], 2018, 2(1), pp. 60-73. [Consulta: 14 octubre 2022]. ISSN 2550-6730. Disponible en: <https://doi.org/10.33936/isrtic.v2i1.1137>
- 54. MUBASHIR, Ali; & MAHNOOR, Paracha.** "An iot based approach for monitoring solar power consumption with adafruit cloud". *International Journal of Engineering Applied Sciences and Technology* [en línea], 2020, (Pakistán) 4(9), pp. 335-341. [Consulta: 16 septiembre 2022]. ISSN 2455-2143. Disponible en: <http://www.ijeast.com/papers/335-341,Tesma409,IJEAST.pdf>
- 55. MULTICOMP.** *Miniature motors* [blog]. 2012. [Consulta: 26 octubre 2022]. Disponible en: <https://manualzz.com/doc/14969661/datasheet-for-compact-8-000-rpm-hobby-motor-3v---6v-dc>
- 56. MUNICIPIO DEL DISTRITO METROPOLITANO DE QUITO; et al.** *Plan Maestro de Gestión Integral de Residuos Sólidos* [en línea]. Quito-Ecuador, 2016. [Consulta: 20 octubre 2022]. Disponible en: <https://gobiernoabierto.quito.gob.ec/wp-content/uploads/documentos/planificaabierta/Archivos/planesmaestros/Plan%20Maestro%20Gestion%20Integral%20de%20Residuos.pdf>
- 57. NICOLAU, Mariano; et al.** *Diseño de nodos de control inteligentes con integración a un sistema de gestión de la energía* [en línea]. Santa Fe: Universidad Tecnológica Nacional Facultad Regional Santa Fe, 2019. [Consulta: 27 noviembre 2022]. Disponible en: https://www.editores-srl.com.ar/sites/default/files/ie341_utn_generacion_distribuida.pdf
- 58. OSAKA ELECTRONICS.** *Módulo Sensor Obstáculos Infrarrojo FC-51* [blog]. 2020. [Consulta: 23 noviembre 2022]. Disponible en: <https://osakaelectronicsltda.com/modulos-robotica/sensores-distancia/modulo-sensor-obstaculos-infrarrojo-fc-51.html>
- 59. PC COMPONENTES Y MULTIMEDIA.** *Raspberry Pi 4 Model B* [blog]. 2022. [Consulta: 20 octubre 2022]. Disponible en: <https://www.pccomponentes.com/caracteristicas-raspberry-pi-4>
- 60. POZO-RUZ, Ana; et al.** "Sistema de posicionamiento global (gps): descripción, análisis de errores, aplicaciones y futuro". *E.T.S Ingenieros de Telecomunicaciones* [en línea], 2000, (España), pp. 2-3. [Consulta: 30 octubre 2022]. Disponible en: http://www.oocities.org/es/foro_gps/infografia/gps5.pdf
- 61. PRIETO, Josep.** *Introducción a los sistemas de comunicación inalámbricos* [en línea]. Cataluña-España: Universitat Oberta de Catalunya, 2011. [Consulta: 23 octubre 2022]. Disponible en: <https://goo.su/OVIO7>
- 62. PYTHON.ORG.** *The Python Logo* [blog]. 2022. [Consulta: 15 octubre 2022]. Disponible en: <https://www.python.org>.
- 63. QUECTEL.** *Quectel L80* [blog]. 2022. [Consulta: 26 noviembre 2022]. Disponible en: https://www.quectel.com/wp-content/uploads/pdfupload/Quectel_L80_GPS_Specification_V1.3.pdf

- 64. QUIÑONEZ, Oswaldo.** *Internet de las Cosas (IoT)* [en línea]. Ibukku LLC, 2019. [Consulta: 28 octubre 2022]. Disponible en: https://books.google.es/books?hl=es&lr=lang_es&id=vnnEDwAAQBAJ&oi=fnd&pg=PT2&dq=i-ot&ots=oRsvR_pruq&sig=udqsVHg3NITzF_1-D5z3A0IubnU#v=onepage&q=i-ot&f=false
- 65. RASPBERRY PI LTD.** *About Raspberry Pi* [blog]. 2022. [Consulta: 14 octubre 2022]. Disponible en: <https://www.raspberrypi.org/about/>.
- 66. RASPBERRY PI LTD.** *Raspberry Pi 3 Model B+* [blog]. 2022. [Consulta: 14 octubre 2022]. Disponible en: <https://www.raspberrypi.com/products/raspberry-pi-3-model-b-plus/>.
- 67. RASPBERRY PI LTD.** *Raspberry Pi 4 Model B* [blog]. 2022. [Consulta: 14 octubre 2022]. Disponible en: <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/>.
- 68. REPÚBLICA DEL ECUADOR.** *Texto Unificado de Legislación Secundaria de Medio Ambiente* [en línea]. Quito-Ecuador: Asamblea Nacional de la República del Ecuador, 2015. [Consulta: 16 octubre 2022]. Disponible en: https://www.gob.ec/sites/default/files/regulations/2018-09/Documento_acuerdo-ministerial-061.pdf
- 69. REPÚBLICA DEL ECUADOR.** *Código Orgánico del Ambiente* [en línea]. Quito-Ecuador: Asamblea Nacional de la República del Ecuador, 2017. [Consulta: 16 octubre 2022]. Disponible en: https://www.ambiente.gob.ec/wp-content/uploads/downloads/2018/01/CODIGO_ORGANICO_AMBIENTE.pdf
- 70. REPÚBLICA DEL ECUADOR.** *Código Orgánico de Organización Territorial, Autonomía y Descentralización* [en línea]. Quito-Ecuador: Asamblea Nacional de la República del Ecuador, 2019. [Consulta: 16 octubre 2022]. Disponible en: <https://www.gob.ec/sites/default/files/regulations/2020-10/CODIGO-ORGANICO-DE-ORGANIZACION-TERRITORIAL-COOTAD.pdf>
- 71. REVELO, Boris; & HENAO, Silvia.** "Aplicación de herramientas gratuitas para la medición en línea de variables electromagnéticas". Simposio de Investigación USTAMED [en línea], 2017, (Colombia) 4, p. 280. [Consulta: 28 octubre 2022]. ISSN 2463-0489. Disponible en: <https://goo.su/lqdCaB>
- 72. SALAZAR, Jordi.** *Redes Inalámbricas* [en línea]. Erasmus+, 2016. [Consulta: 18 octubre 2022]. Disponible en: https://upcommons.upc.edu/bitstream/handle/2117/100918/LM01_R_ES.pdf
- 73. SANTACRUZ, Karen.** Evaluación del sistema de recolección de residuos sólidos Urbanos del Cantón Riobamba mediante el uso de herramientas SIG para la mejora del modelo de gestión [en línea] (Trabajo de titulación) (Ingeniería). Universidad Nacional de Chimborazo, Carrera de Ingeniería Ambiental, Riobamba-Ecuador. 2021. [Consulta: 19 octubre 2022]. Disponible en: <http://dspace.unach.edu.ec/handle/51000/8022>

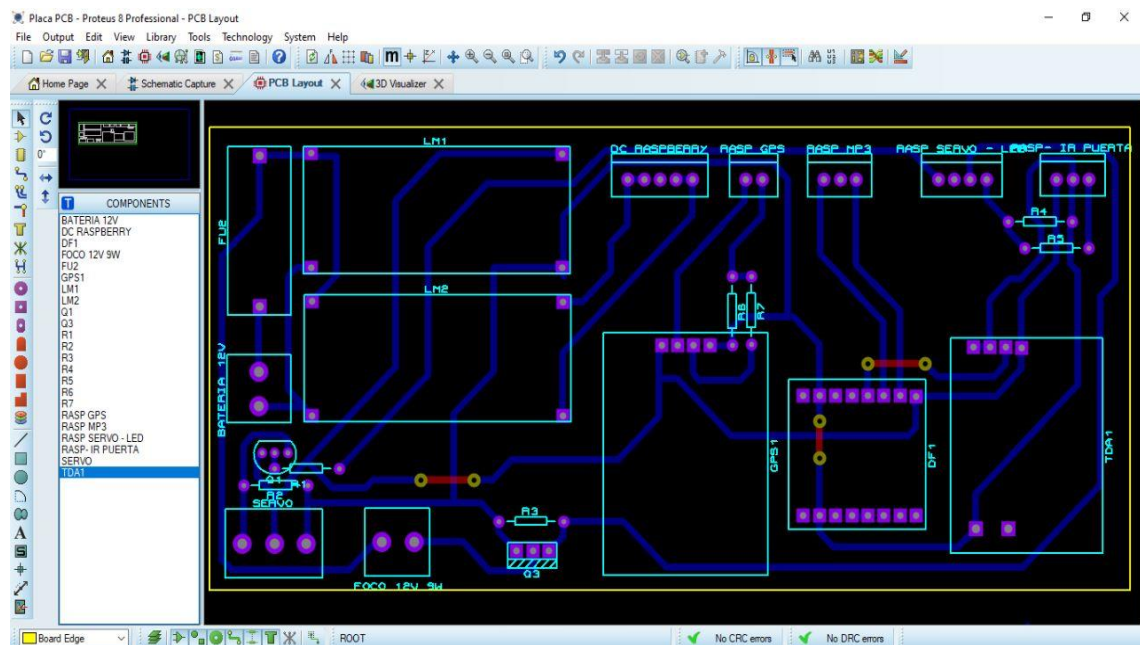
- 74. SK.** *Raspberry Pi OS* [blog]. 2022. [Consulta: 20 noviembre 2022]. Disponible en: <https://www.raspberrypi.com/software/>.
- 75. SOUZA, Fernando.** *How To Use a Camera in Raspberry Pi With OpenCV* [blog]. 2021. [Consulta: 20 noviembre 2022]. Disponible en: <https://medium.com/vacatronics/how-to-use-a-camera-in-raspberry-pi-with-opencv-bb6cf42650da>.
- 76. TECNOPURA.** *Módulo amplificador para audio monofónico TDA2030A 18W 6~12VDC* [blog]. 2022. [Consulta: 02 noviembre 2022]. Disponible en: <https://www.tecnopura.com/producto/modulo-amplificador-para-audio-monofonico-tda2030a-18w-612vdc/>
- 77. TOLEDO, Jessica; & QUINTERO, Carlos.** "Gestión de residuos sólidos urbanos en México: un caso de estudio desde la perspectiva organizacional". *RAE-Revista de Administração de Empresas* [en línea], 2021, (México) 62(3), pp. 2-4. [Consulta: 23 octubre 2022]. ISSN 2178-938X. Disponible en: <http://dx.doi.org/10.1590/S0034-759020220302>
- 78. TOMALÁ, Erick; & SANTANA, Solange.** Prototipo electrónico de seguridad vehicular utilizando la plataforma BLYNK para generar alertas, alarma, bloqueo y encendido del automotor mediante comando de voz y sistema biométrico facial [en línea] (Tesis) (Ingeniería). Universidad de Guayaquil, Carrera de Ingeniería en Networking y Telecomunicaciones, Guayaquil-Ecuador. 2022. [Consulta: 22 octubre 2022]. Disponible en: <http://repositorio.ug.edu.ec/handle/redug/64525>
- 79. TORRENTE-VELÁSQUEZ, Jorge; et al.** "Robust information for effective municipal solid waste policies: Identifying behaviour of waste generation across spatial levels of organization". *Waste Management* [en línea], 2020, 103, pp. 208-217. [Consulta: 07 octubre 2022]. ISSN 0956-053X. Disponible en: <https://www.sciencedirect.com/science/article/pii/S0956053X1930786X>
- 80. TRIANO, Moisés.** *Smart Cities and Connected Mobility* [en línea] (Trabajo final de grado) (Ingeniería). Universitat Politècnica de Catalunya, Barcelona-España. 2020. [Consulta: 22 octubre 2022]. Disponible en: <https://upcommons.upc.edu/bitstream/handle/2117/177576/memoria.pdf?sequence=1&isAllowed=y>
- 81. UBIDOTS.** *Ubidots: IoT industrial potente pero simple* [blog]. 2023. [Consulta: 02 febrero 2023]. Disponible en: <https://ubidots.com/>.
- 82. UNIT ELECTRONICS.** *Cámara Visión Nocturna Raspberry Pi 5MP 1080p OV5647* [blog]. 2022. [Consulta: 12 noviembre 2022]. Disponible en: <https://uelectronics.com/producto/camara-vision-nocturna-raspberry-pi-5mp-1080p-ov5647/>

ANEXOS

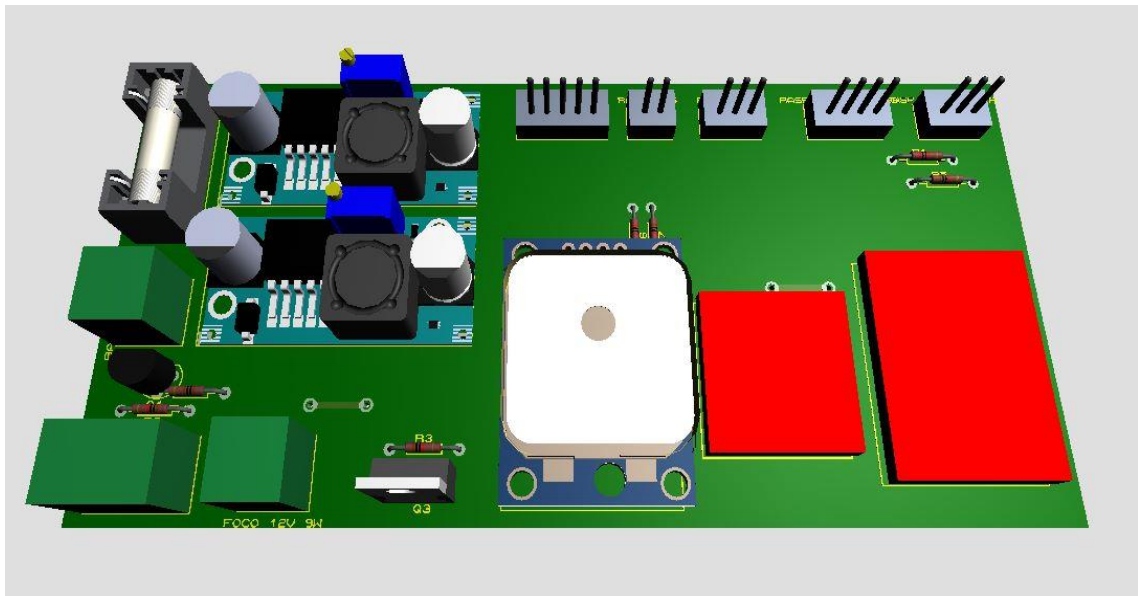
ANEXO A: PRUEBAS DE CONEXIÓN DE LA RASPBERRY PI CON EL MÓDULO GPS



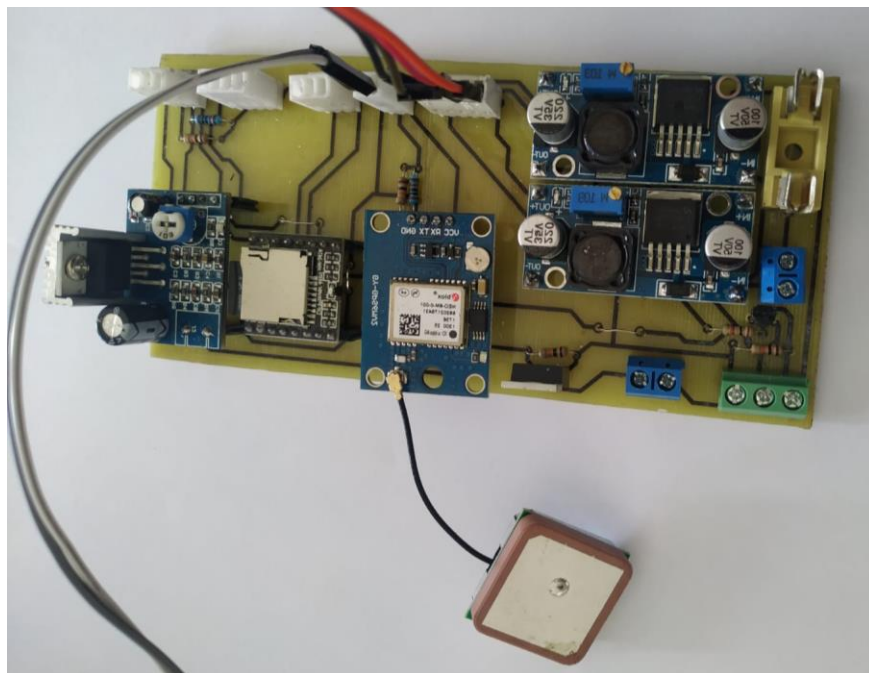
ANEXO B: DISEÑO DE LA PLACA PCB EN EL SOFTWARE PROTEUS



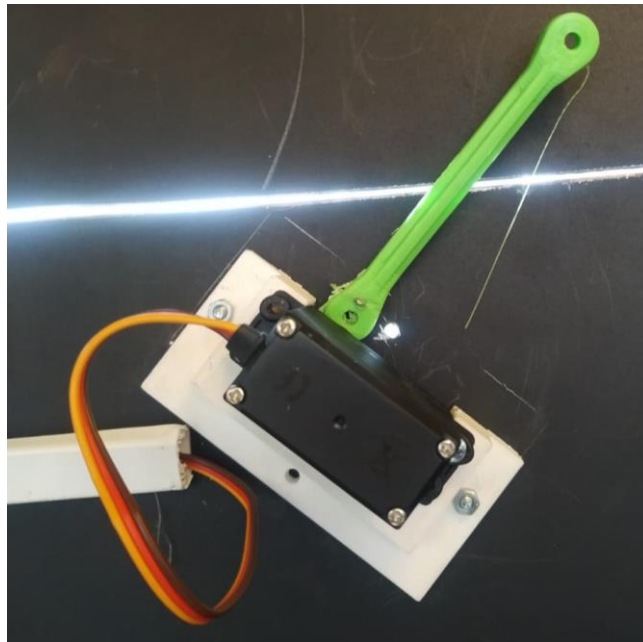
ANEXO C: VISTA 3D DE LA PLACA PCB



ANEXO D: ENSAMBLE DE COMPONENTES ELECTRÓNICOS EN LA PLACA PCB



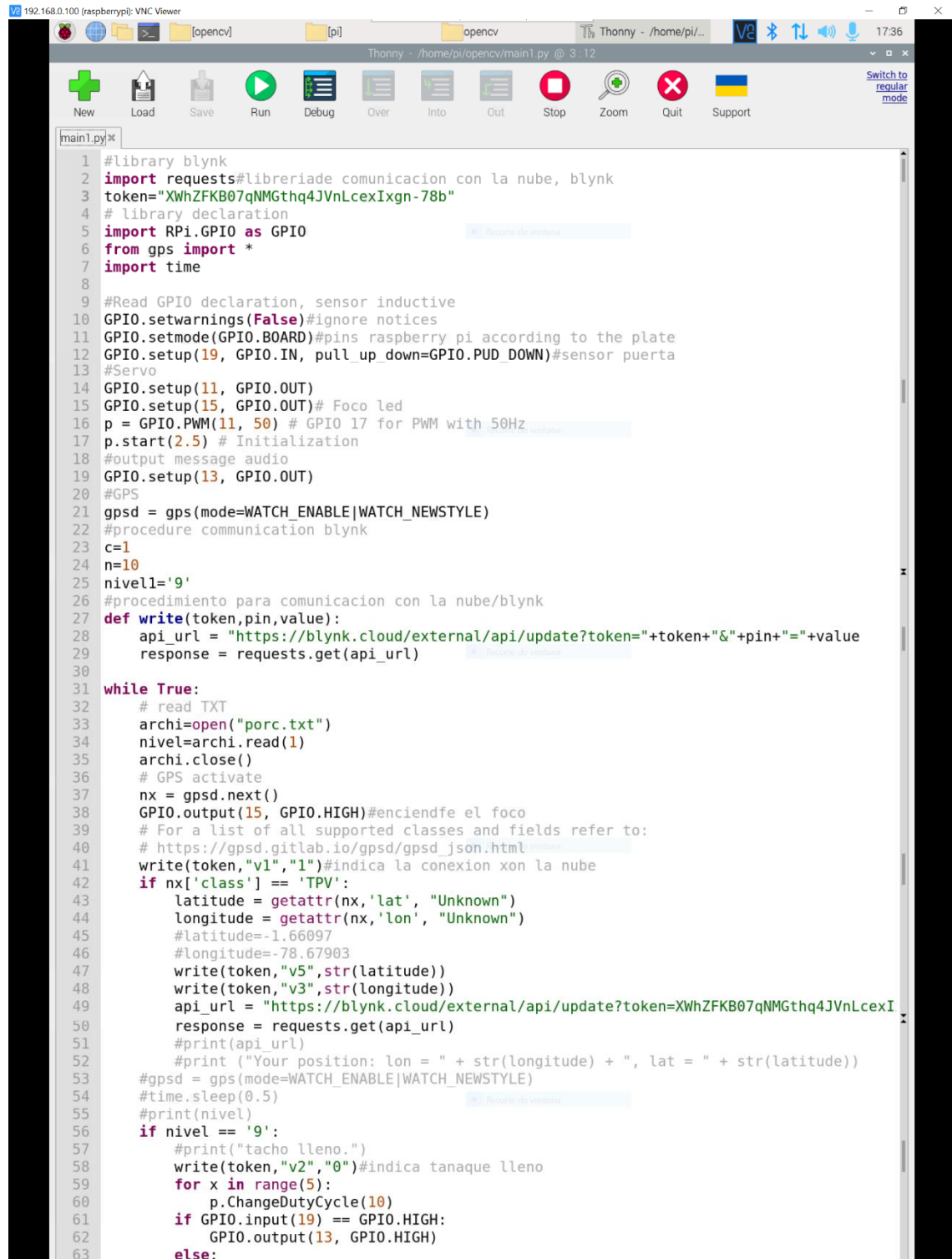
ANEXO E: SISTEMA DE BLOQUEO DE ACCESO A LA PUERTA DEL BASURERO



ANEXO F: CÁMARA EMPLEADA PARA LA VISIÓN ARTIFICIAL



ANEXO G: PROGRAMACIÓN DE VISIÓN ARTIFICIAL



```
1 #library blynk
2 import requests#libreria de comunicacion con la nube, blynk
3 token="XWhZFKB07qNMGthq4JVnLcexIxgn-78b"
4 # library declaration
5 import RPi.GPIO as GPIO
6 from gps import *
7 import time
8
9 #Read GPIO declaration, sensor inductive
10 GPIO.setwarnings(False)#ignore notices
11 GPIO.setmode(GPIO.BOARD)#pins raspberry pi according to the plate
12 GPIO.setup(19, GPIO.IN, pull_up_down=GPIO.PUD_DOWN)#sensor puerta
13 #Servo
14 GPIO.setup(11, GPIO.OUT)
15 GPIO.setup(15, GPIO.OUT)# Foco led
16 p = GPIO.PWM(11, 50) # GPIO 17 for PWM with 50Hz
17 p.start(2.5) # Initialization
18 #output message audio
19 GPIO.setup(13, GPIO.OUT)
20 #GPS
21 gpsd = gps(mode=WATCH_ENABLE|WATCH_NEWSTYLE)
22 #procedure communication blynk
23 c=1
24 n=10
25 nivel='9'
26 #procedimiento para comunicacion con la nube/blynk
27 def write(token,pin,value):
28     api_url = "https://blynk.cloud/external/api/update?token="+token+"&pin="+value
29     response = requests.get(api_url)
30
31 while True:
32     # read TXT
33     archi=open("porc.txt")
34     nivel=archi.read(1)
35     archi.close()
36     # GPS activate
37     nx = gpsd.next()
38     GPIO.output(15, GPIO.HIGH)#enciendfe el foco
39     # For a list of all supported classes and fields refer to:
40     # https://gpsd.gitlab.io/gpsd/gpsd_json.html
41     write(token,"v1","1")#indica la conexion xon la nube
42     if nx['class'] == 'TPV':
43         latitude = getattr(nx,'lat', "Unknown")
44         longitude = getattr(nx,'lon', "Unknown")
45         #latitude=-1.66097
46         #longitude=-78.67903
47         write(token,"v5",str(latitude))
48         write(token,"v3",str(longitude))
49         api_url = "https://blynk.cloud/external/api/update?token=XWhZFKB07qNMGthq4JVnLcexI
50         response = requests.get(api_url)
51         #print(api_url)
52         #print ("Your position: lon = " + str(longitude) + ", lat = " + str(latitude))
53     #gpsd = gps(mode=WATCH_ENABLE|WATCH_NEWSTYLE)
54     #time.sleep(0.5)
55     #print(nivel)
56     if nivel == '9':
57         #print("tacho lleno.")
58         write(token,"v2","0")#indica tanaque lleno
59         for x in range(5):
60             p.ChangeDutyCycle(10)
61         if GPIO.input(19) == GPIO.HIGH:
62             GPIO.output(13, GPIO.HIGH)
63     else:
```



```

64         GPIO.output(13, GPIO.LOW)#activa EL MODULO DE AUDIO
65         time.sleep(0.1)
66         GPIO.output(13, GPIO.HIGH)
67
68     else:
69         #print("tacho vacio")
70         p.ChangeDutyCycle(5)
71         time.sleep(0.5)
72         write(token,"v2","1")#TANQUE AUN NO LLENO
73     if c < 10:
74         if nivel > '0':#control del menor nuenmro
75             if str(nivell) > nivel:
76                 nivell = nivel
77                 c=c+1
78     if c==10:
79         if nivell == '1':
80             n=10
81         if nivell == '2':
82             n=20
83         if nivell == '3':
84             n=30
85         if nivell == '4':
86             n=40
87         if nivell == '5':
88             n=50
89         if nivell == '6':
90             n=60
91         if nivell == '7':
92             n=70
93         if nivell == '8':
94             n=80
95         if nivell == '9':
96             n=90
97         nivell=9
98         c=1
99         #print(n)
100        write(token,"v4",str(n) )
101        write(token,"v1","0")
102        time.sleep(1)
103    p.stop()
104    GPIO.cleanup()

```

Shell

Python 3.9.2 (/usr/bin/python3)
 >>>

Local Python 3 · /usr/bin/python3

ANEXO H: CONSTRUCCIÓN DEL PROTOTIPO



ANEXO I: VIDEO DEL FUNCIONAMIENTO DEL PROTOTIPO

[Funcionamiento prototipo.mp4](#)