



UNIVERSIDAD NACIONAL AUTÓNOMA DE NICARAGUA, MANAGUA

UNAN - MANAGUA

Facultad de Ciencias e Ingeniería

Departamento de Tecnología

Tema:

Sistema de adquisición de video y medición de distancia durante el retroceso de un vehículo en un dispositivo móvil.

**Seminario de Graduación como requisito final para optar al título de
Ingeniero Electrónico.**

Autor/es:

✚ Carlos Eliud Marquez Bejarano.

Carnet: 14042460

✚ Everth Josué Méndez Cruz.

Carnet: 14044540

Tutor(a):

✚ Msc. Milcíades Delgadillo.

✚ Msc. Karen María Acevedo Mena.

Managua, Nicaragua 13 de enero de 2020.

Dedicatoria.

Dios por darme la oportunidad de vivir y estar conmigo en todo momento, en cada paso que doy permitirme a cumplir mis metas.

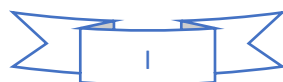
Mi madre por el apoyo incondicional que siempre estuvo presente guiándome y aconsejando en todo momento y por los valores inculcados.

A mi familia por los ánimos que me brindaron cuando lo necesitaba cuando todo parecía acabar y por la ayuda que siempre me dieron en todo momento.

A mi compañero Carlos Marquez y a todas esas personas que con sus palabras de una u otra manera han formado parte de mis logros.

A mi prometida que gracias a ella he aprendido que cada esfuerzo tiene un valor y cada obstáculo se supera con dedicación.

Everth Josué Méndez Cruz



Dedicatoria

Primeramente, a Dios por haberme permitido llegar hasta este punto y haberme dado salud, darme lo necesario para seguir adelante día a día y lograr mis metas.

A mi Familia, por apoyarme en todo momento, por la motivación constante que hicieron un hombre de bien, con valores y virtudes las cuales hacen posible que el día de hoy este aquí presente, pero más que nada, por su amor y paciencia.

A mis maestros por la sabiduría que me transmiten en el desarrollo de mi formación profesional por su gran apoyo y motivación para la culminación de nuestra investigación y por impulsar el desarrollo de nuestra formación profesional.

Carlos Eliud Marquez Bejarano



Agradecimiento.

Agradecemos a Dios por habernos dado la fuerza, sabiduría y paciencia para poder llegar a cumplir las metas propuestas.

A nuestra asesora metodológica Msc. Karen Acevedo por ser orientación y dedicación brindada para la elaboración de este documento.

A nuestro amigo Ing. Melvin Antonio Rivera Espinoza por su ayuda incondicional para la realización de este proyecto.

A nuestro amigo Ing. Julio Eduardo Hernández Mendieta por su apoyo para la realización de este proyecto.

A nuestro tutor Msc. Milcíades Delgadillo, por su orientación que nos ha brindado en la realización de este proyecto.

Finalmente, a todas aquellas personas que estuvieron presentes apoyándonos para cumplir nuestras metas.

Los Autores



Resumen.

El presente proyecto tiene por objetivo desarrollar un sistema de adquisición de video y medición de distancia durante el retroceso de un vehículo en un dispositivo móvil, que pueda ser utilizado como una herramienta para todos los conductores.

Para el desarrollo de la misma se han planteado objetivos centrales que guían el proyecto, así mismo se dispone de material teórico que brinda información de la temática que se está abordando, se plasma así mismo la herramienta utilizada (encuesta) con su debido análisis de datos para posteriormente la información ser interpretada y utilizada en el apartado de recomendaciones de ser necesario.

Para el diseño del sistema de adquisición de video y medición de distancia durante el retroceso de un vehículo en un dispositivo móvil se detallada como estará constituido el sistema propuesto, conociendo estos se procede a la elaboración del diseño electrónico a través del cual se plantean los dispositivos electrónicos a utilizar (sensores de proximidad, plataforma de control arduino uno, raspberry pi 3B+, plataforma android), y mediante el circuito de medición de proximidad se describe el datasheet de los sensores de proximidad, el micro controlador (arduino) que tendrá la lectura de dichos sensores.

Finalmente se presentaran resultados de las pruebas de funcionamiento del sistema una vez montado como un prototipo a escala con sus respectivos componentes.

Tabla de contenido

Dedicatoria.....	I
Agradecimiento.....	III
Resumen.....	IV
I. Introducción.....	8
II. Antecedentes.....	9
III. Planteamiento del problema.....	10
IV. Justificación del problema.....	11
V. Objetivos.....	12
Objetivo General.....	12
Objetivos Específicos.....	12
VI. Fundamentos Teóricos.....	13
1. Video cámara.....	13
1.1. Sensor CMOS.....	14
1.2. La imagen digital.....	16
2. Redes informáticas.....	18
2.1. Clasificación.....	19
2.2. Velocidades de conexión.....	19
2.3. Red inalámbrica.....	20
3. Plataforma Android.....	24
3.1. Características.....	24
3.2. Arquitectura de Android.....	25
4. Sensores de proximidad.....	28
4.1. Tipos de sensor de proximidad.....	28
Desarrollo	
VII. Análisis del sistema de aparcamiento actual vs sistema de aparcamiento propuesto.....	31

1.	Sistema de aparcamiento actual.	31
1.1.	Ventajas del sistema de aparcamiento	32
2.	Sistema de aparcamiento propuesto	33
3.	Sistema de aparcamiento actual vs sistema de aparcamiento propuesto. 33	
VIII.	Diseño Metodológico.	34
IX.	Hardware	43
2.	Diseño del sistema de adquisición de video y medición de distancia durante el retroceso de un vehículo en un dispositivo móvil.	43
1.1.	Diseño Electrónico.	43
1.2.	Diagrama de bloque de periféricos de entrada y salida del sistema Electrónico.	65
X.	Software	67
1.	Lenguajes de programación.	67
1.2.	Python.....	67
1.3.	Lenguaje C ++	68
1.4.	Java	69
2.	Tarjeta arduino uno.	71
2.1.	Diagrama de flujo del programa de arduino.	71
2.2.	Comunicación serial con la Raspberry pi 3 B+.....	73
3.	Software implementado en Raspberry pi 3 B+	76
3.1.	Configuración inicial de la Raspberry pi 3 B+	76
3.2.	Pasos para la instalación de NOOBS.	76
3.3.	Instalación del servidor de video.....	84
3.4.	Configuración como Access Point	88
3.5.	Programa principal.....	97
4.	Desarrollo de la aplicación para Android	100
4.1.	Creación de la aplicación	100

4.2. Activity de BackwardView.	103
4.3. Escala de proximidad.....	111
4.4. Alarmas Sonoras	112
XI. Pruebas de funcionamiento del sistema de adquisición de video y medición de distancia durante el retroceso de un vehículo en un dispositivo móvil mediante un prototipo a escala en un ambiente controlado.....	114
Paso 1.Montaje de los componentes a utilizar.....	114
Paso 2. Arrancar el emulador donde se estaran haciendo pruebas de la app.	115
Paso 3. Se empiezan a realizar pruebas de distancia.....	115
Paso 4. Prueba de funcionamiento a una distancia de 30 cm.	116
Paso 5. Prueba de funcionamiento a una distancia de 60 cm.	116
Paso 6. Prueba de funcionamiento a 80 cm.	117
Paso 7. Prueba de funcionamiento a una distancia de 101 cm.	117
XII. Presupuesto de los componentes utilizados para la elaboración del prototipo a escala del sistema de adquisición de video y medición de distancia durante el retroceso en un dispositivo móvil.....	118
XIII. Conclusiones	119
XIV. Recomendaciones.....	120
Bibliografía	121
Anexos	124
Anexo A. Modelo de Encuesta utilizado en la recolección de datos.	124

Índice de Figura

Figura 1.Sensor CMOS.....	14
Figura 2.Filtro de Bayer	15
Figura 3. Números binarios formando una imagen digital.	16
Figura 4.Proceso para la formación de una imagen digital.	17
Figura 5.Estructura de una red WLAN.....	19
Figura 6.Eschema de una red inalámbrica.....	20
Figura 7.Características de Android.....	24
Figura 8. Arquitectura Android.	27
Figura 9.Sensores de proximidad capacitivos.	28
Figura 10.Sensor de proximidad inductivo.....	29
Figura 11. Grafica N° 1 Edad.	37
Figura 12.Gráfico N° 2 Sexo.	37
Figura 13 Gráfico N°3 Pertenece.....	38
Figura 14. Gráfico N° 4 Año del vehículo que conduce.	38
Figura 15. Gráfico N° 5 Sistema de ayuda para el retroceso del vehículo.	39
Figura 16. Gráfico N° 6 Implementación de un sistema de ayuda para el momento de retroceder en su vehículo.....	39
Figura 17. Gráfico N° 7 ¿Considera necesario la implementación de un sistema de ayuda en el retroceso de un vehículo?.....	40
Figura 18. Gráfico N° 8 Un nuevo sistema de ayuda para el retroceso a través de una aplicación móvil.	40
Figura 19. Gráfico N° 9 ¿Le gustaría que este sistema sea de bajo costo?	41
Figura 20. Gráfico N° 10 Ventajas de ser implementado el sistema.	41
Figura 21. Gráfico N° 11 Desventajas de ser implementado el sistema. ...	42
Figura 22.Principales Componentes del Sistema.....	43
Figura 25.Sensor JSN-SR04T.....	44
Figura 26.Sensor de ultrasonidos HC-SR04.	46
Figura 27.Tabla de especificaciones del sensor HC-SR04.	47
Figura 28.Envío y recepción del pulso ultrasónico.....	47
Figura 29.Incertidumbre angular del sensor HSR04.	48
Figura 30.Error producido por varios sensores ultrasónico (crosstalk)... ..	49

Figura 31.Placa Electrónica Arduino Uno.	51
Figura 32.Diagrama de bloques de las partes de Arduino.	52
Figura 33.Conexiones de los Sensores Ultrasónicos.	55
Figura 34.Raspberry Pi 3 Model B+ 2017 Partes que lo conforman	58
Figura 35.Raspberry Pi 3 B+ Pin Out.	60
Figura 36.Módulo de Cámara Raspberry.	62
Figura 24.Comunicación entre los dispositivos.	64
Figura 23. Diagrama de bloque de periféricos de entrada y salida del sistema Electrónico.	65
Figura 37.Logo de Python	67
Figura 38.Lenguaje C++ logo.	68
Figura 39.Lenguaje Java logotipo.	70
Figura 40.Diagrama de Flujo de la programación de arduino.	72
Figura 41. Diagrama de bloque de envío de datos a Raspberry pi 3 B+. ...	74
Figura 42.Diagrama de flujo de selección del valor mínimo de los sensores.	75
Figura 43.Página oficial para descargar NOOBS.	77
Figura 44.Página de descarga de SD Card Formatter.	78
Figura 45. Página principal del software SD Card Formatter.	78
Figura 46.Menú de distintos sistemas operativos disponibles en Raspberry Pi.	79
Figura 47. Ventana de espera de instalación del sistema operativo seleccionado.	79
Figura 48. Ventana emergente se ha instalado correctamente.	80
Figura 49.Ventana de configuración de lenguaje.	81
Figura 50.Cambiar clave de usuario de administrador.	82
Figura 51.Activar la interface SSH.	82
Figura 52.Terminal de Raspbian comandos de actualización.	83
Figura 53.Módulo de cámara raspberry.	84
Figura 54.Comando ifconfig.	88
Figura 55.Instalación de las reglas de IPV4.	89
Figura 56. Archivo de modificación dhcpd.conf	90
Figura 57.Archivo de modificación dnsmasq.conf	91
Figura 58.Archivo de configuración dhcpd.conf	92

Figura 59. Parametros de configuración para el hostapd.....	93
Figura 60. Archivo de configuración hostapd el default.....	94
Figura 61. Archivo de enrutamiento y mascarada.....	95
Figura 62. Archivo de configuración para que se ejecute el hostapd como Daemon.....	96
Figura 63. Encapsulación de datagramas UDP.	98
Figura 64. Flujo de datos cliente-servidor.....	99
Figura 65. Welcome to Android Studio	101
Figura 66. Ventana de Choose Your Project.	101
Figura 67. Ventana de Configure Your Project	102
Figura 68. LoginActivity.xml(Pantalla de inicio)	105
Figura 69. Actividad configuración.....	108
Figura 70. Pantalla de Ayuda.....	110
Figura 71. Ensamble.	114
Figura 72. Emulador Android 7.1	115
Figura 73. Prueba de funcionamiento con un objeto a 10 cm.	115
Figura 74. Prueba de funcionamiento a 30 cm.	116
Figura 75. Prueba de funcionamiento a 60 cm.	116
Figura 76. Prueba de funcionamiento a 80 cm.	117
Figura 77. Prueba de funcionamiento a 101 cm.	117

Índice de tablas

Tabla 1.Tabla de los estándares de WiFi.....	23
Tabla 2.Ventajas del sistema de aparcamiento.	32
Tabla 3.Características del Arduino Uno.....	51
Tabla 4.Tabla de Tecnología WI-FI y Bluetooth.	61
Tabla 5.Componentes dinámicos de Login Activity.....	106
Tabla 6. Componentes dinámicos de la ConfActivity	109
Tabla 7.Componentes dinámicos de HelpActivity.....	110
Tabla 8.Relación de color - Proximidad.....	111
Tabla 9.Sonidos de escala de proximidad.	113
Tabla 10.Presupuesto de los componentes utilizados.	118

I. Introducción.

Los sistemas de adquisición de video y medición de distancia son también llamados sistemas de aparcamientos, estos comenzaron siendo un elemento de lujo en los vehículos, hoy en día son una herramienta que facilitan al conductor disminuir el índice de colisiones en el fascia (parte trasera) del automóvil.

El presente proyecto tiene la finalidad de crear un sistema que tenga como objetivo reducir las colisiones, el cual estará dirigido a todos los conductores que aún no cuentan con esta herramienta.

Para la ejecución del mismo es necesario contar con elementos estratégicos tales como:

1. Análisis del sistema de aparcamiento actual vs sistema de aparcamiento propuesto.
2. Herramientas de recolección de datos (encuesta).
3. Fundamentos teóricos que incluyen información necesaria para la realización del diseño del sistema.
4. Lenguajes de programación:
 - a. C (arduino).
 - b. Python (Raspberry Pi 3B+).
 - c. Java (IDE Android Studio).
5. Servidor de video.
6. Check point.

Cada uno de estos aspectos se puntualiza en el desarrollo de este trabajo dando a conocer detalladamente los anteriormente mencionados.

II. Antecedentes.

El presente proyecto sistema de adquisición de video y medición de distancia a través de un dispositivo móvil no posee estudios previos realizados, no obstante los sistemas de visión trasera en los automóviles surgieron en el año 2001 por la empresa infiniti creada por NISSAN, esta tecnología empezó siendo un elemento de lujo en los autos y ahora es contemplado como una necesidad.

Este sistema de visión trasera consiste en que cuando la transmisión del auto es colocada en la posición de reversa, la pantalla muestra una imagen en vivo del área que está detrás del vehículo. Esta imagen presenta líneas de tres colores (rojo, amarillo y verde), que indican las distancias aproximadas entre un objeto y la facia trasera.

Este método de cámara de reversa puede variar de un auto a otro, en la forma en cómo se proyecta, trabaja y los ajustes que tiene. (Sosa, 2016)

En la actualidad en Nicaragua existen diversos modelos tales como: el Ford escape, Honda Pilot, Chevrolet Trax, Chevrolet Equinox, Chevrolet Traverse, Toyota 4Runner, Toyota Highlander, Hyundai Santa Fe, Toyota Sienna, entre otros, que ya cuentan con este sistema de cámaras de visión trasera y son modelos que surgieron en 2014 (Sanchez, 10), no obstante no todos los vehículos lo poseen, por ello se pretende desarrollar un sistema de adquisición de video y medición de distancia durante el retroceso de un vehículo haciendo uso de un dispositivo móvil, donde a través del móvil se estará visualizando una imagen en tiempo real.

III. Planteamiento del problema

En la actualidad a nivel mundial la modernización es un proceso que viene abriendo camino a nuevas y cada vez mejores herramientas que facilitan al ser humano la realización de determinadas tareas. En el sector industrial automovilístico cada año se fabrica una nueva versión mejorada de vehículos con el fin de cumplir las demandas de los clientes.

En Nicaragua desde año 2017 hasta hoy en día los vehículos cuentan con una herramienta fundamental de gran ayuda para el conductor como es un sistema de adquisición de video y medición de distancia utilizado en el retroceso de los vehículos a través de una pantalla integrada donde se muestra una imagen en tiempo real.

No obstante, no todos los ciudadanos nicaragüenses tienen los recursos para comprar un vehículo último modelo por el factor económico es por ello que se plantea la siguiente interrogante:

¿La falta de implementación de un sistema de adquisición de video y medición de distancia durante el retroceso genera vulnerabilidad de colisión en los puntos ciegos del vehículo?

IV. Justificación del problema.

Debido al alto índice de accidente de tránsito en la ciudad, surge la necesidad de implementar un sistema de adquisición de video y medición de distancia para el retroceso de un vehículo observado en forma remota desde un dispositivo móvil mediante una aplicación para la plataforma Android. Esto será de utilidad para los vehículos de los años 2017 hacia años atrás.

Esta herramienta brindará ayuda a los conductores de vehículos, facilitando el proceso de retroceso de manera que disminuirá colisiones en dicha tarea.

Debido a lo antes mencionado se pretende desarrollar un sistema viable y actualizado que se adaptará a las necesidades de los distintos vehículos los cuales son:

- Modernizar a través de este sistema a vehículos que no cuentan con esta herramienta.
- Limitar el índice de colisiones en la parte trasera del vehículo.

Por lo tanto, esta herramienta ayudará a resolver las necesidades del conductor y facilitarle la salida de un estacionamiento o en cualquier lugar lleno de vehículo y será un sistema de bajo costo para el usuario.

V. Objetivos

Objetivo General

- ✚ Desarrollar un sistema de adquisición de video y medición de distancia durante el retroceso de un vehículo en un dispositivo móvil.

Objetivos Específicos

- ✚ Realizar análisis del sistema de aparcamiento actual vs sistema de aparcamiento propuesto.
- ✚ Describir los componentes que integran el sistema de adquisición de video y medición de distancia durante el retroceso de un vehículo en un dispositivo móvil.
- ✚ Diseñar el sistema de adquisición de video y medición de distancia del vehículo.
- ✚ Demostrar el funcionamiento del sistema de adquisición de video y medición de distancia mediante un prototipo a escala en un ambiente controlado.

VI. Fundamentos Teóricos.

A continuación, se abordará las teorías esenciales para la investigación que estará constituido en 4 partes, las cuales son: Video cámara, redes informáticas, plataforma Android y sensores de proximidad.

1. Video cámara.

Una videocámara o cámara de video es un dispositivo generalmente portátil que permite registrar imágenes y sonidos, convirtiéndolos en señales eléctricas que pueden ser reproducidos por un aparato determinado.

El proceso de funcionamiento de las videocámaras comienza con la descomposición de la luz entre tres componentes (rojo, verde y azul) mediante un prisma de espejos dicróicos. Del otro lado del prisma se encuentran los captadores, que reconstruyen la imagen y la reenvían a los circuitos preamplificadores.

La imagen continúa su recorrido hacia unos procesadores que se encargan de la corrección de distintos parámetros. Una vez que la imagen ya está lista, puede ser grabada o emitida hacia un sistema de producción, o sea, a la red de dispositivos que conforman el entorno de trabajo.

En el caso de las cámaras de TV, pueden reconocerse dos grandes tipos: las portátiles, que están asociadas a un sistema de grabación de señales, y las de estudio, que funcionan integradas a un sistema de producción. (Porto J. P., 2009)

1.1. Sensor CMOS

El Sensor es la parte fundamental de la cámara de fotos, es el elemento que captará la imagen cuya información es completamente analógica y la traducirá a un formato digital que es con el que se trabajará para mejorar la imagen, para imprimirla o para enviarla vía Wifi. Los ordenadores y los equipos digitales utilizan información digital, y la imagen que queremos captar sólo contiene información analógica, pues el sensor es el elemento que realiza la conversión analógico-digital, en la figura 1 se muestra el sensor CMOS.

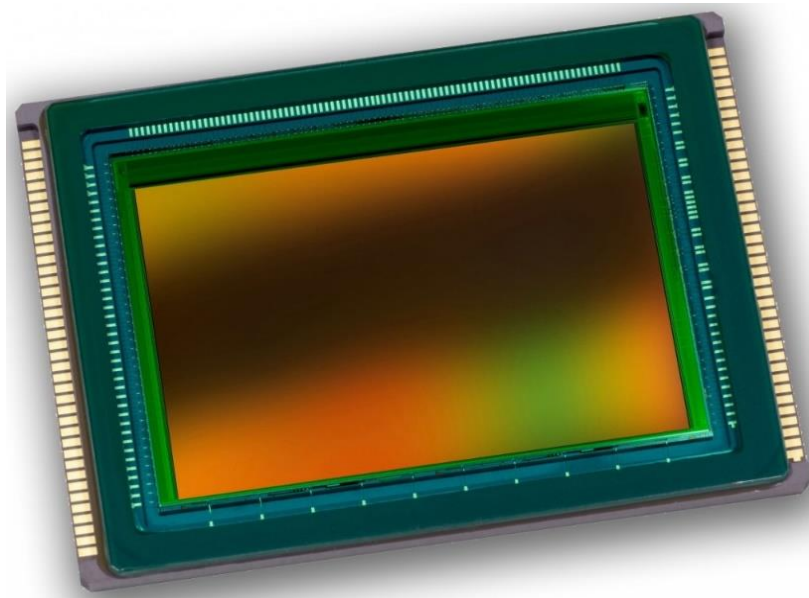


Figura 1.Sensor CMOS.

Fuente: (Lora, 2019)

Físicamente, un sensor fotográfico está compuesto por un conjunto de receptores que recogerán los valores lumínicos recibidos desde el exterior. Cada foto receptor recibirá un nivel de intensidad lumínica y lo almacenará para ser enviado al procesador de la cámara.

El nivel de intensidad que cada receptor registra depende de los factores que está presentes en cada fotografía: características de reflexión de luz de toma fotográfica, apertura del diafragma, estructura de los

elementos intermedios como los objetivos, con su número y calidad de lentes, etc. Cuando se han recogido todos los datos, será enviado, ya en formato digital, al procesador a través del circuito electrónico que recorre todo el chip.

El proceso físico de conversión analógico digital es el siguiente:

- Cada elemento de la imagen emite una intensidad lumínica analógica determinada.
- Cada celda es sensible a la cantidad de fotones recibidos, registrando un valor específico por cada rango de intensidad luminosa recibida
- El valor recibido por cada celda del sensor se envía al procesador junto con toda la información de identificación de la propia celda que registra la información luminosa exterior.
- El procesador ordena todos los datos recibidos de todas las celdas y configura la imagen digital que acabamos de tomar.

¿Cómo funciona el Sensor CMOS?

La hemos visto cómo funciona el sensor físicamente y de forma general, pero tenemos que aclarar ciertos factores, ya que, si sólo se registran los valores de intensidad luminosa, ¿de dónde se obtienen los datos de color?, por ejemplo. La información de color la añade el procesador, y se obtiene a través de diversos procesos de interpolación, en la figura 2 se mostrará cómo funciona la obtención de una imagen.

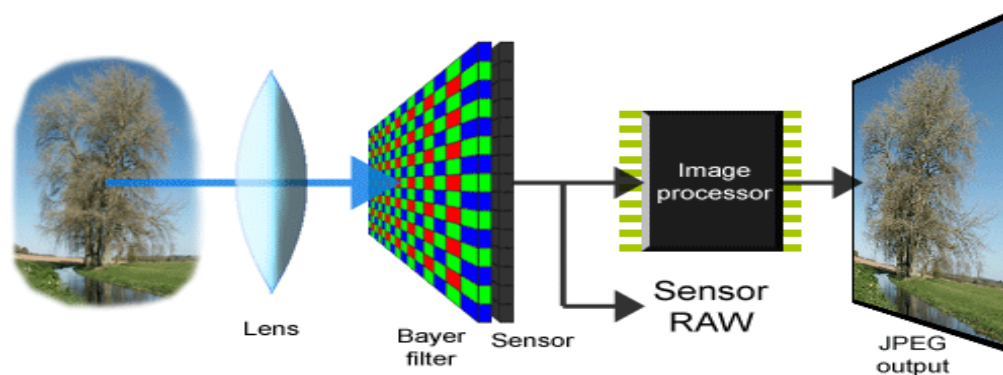


Figura 2. Filtro de Bayer

Fuente: (Lora, 2019)

Uno de los métodos que tradicionalmente han sido utilizados para la obtención de los colores de cada punto de la toma fotográfica es el denominado uso del «Filtro de Bayer». Cada celda foto receptora del sensor fotográfico es recubierta de una malla de filtros con los tres colores básicos: Rojo, Verde, Azul. En este caso, la parte de filtro de color verde tiene el doble de superficie que el de los otros dos colores, ahora veremos por qué. Hay otros tipos de filtros ideados por las diferentes compañías fotográficas, pero el Filtro de Bayer ha sido tomado siempre como referencia para entender cómo se recoge la información de color de la imagen y cómo ésta queda registrada en las células foto receptoras del sensor. (Lora, 2019)

1.2. La imagen digital

Una imagen digital o gráfico digital es una representación bidimensional de una imagen a partir de una matriz numérica, frecuentemente en binario (unos y ceros). Dependiendo de si la resolución de la imagen es estática o dinámica, puede tratarse de una imagen matricial (o mapa de bits) o de un gráfico vectorial. El mapa de bits es el formato más utilizado en informática, en la figura 3 se muestra como los números binarios forman una imagen.



Figura 3. Números binarios formando una imagen digital.

Fuente: (Pineda, 2016)

Más sencillamente dicho o, en otras palabras, una imagen digital proveniente de una cámara de fotos digital está formada por pequeños cuadraditos denominados “píxeles” (del inglés picture-elements, elemento de imagen). Estos miles o millones de píxeles que componen la imagen digital forman una grilla que tiene información acerca cual es la calidad, brillo, contraste, color y definición de la imagen propiamente dicha.

A mayor cantidad de píxeles por pulgada (PPI pixel per inch por sus siglas en inglés) mayor resolución tendrá la imagen, como en el ejemplo de abajo, en donde del lado izquierdo vemos una imagen en full HD (con 2 MP o 2 millones de píxeles) y del lado derecho una con resolución 4K (8 MP u 8 millones de píxeles) en donde claramente se ve la diferencia en resolución. (Imagenmedia.Blogspot, 2016)

Formación de una imagen digital

El proceso de formación de una imagen digital obtenida a través de una cámara es el siguiente:

La luz que detecta el objetivo de la cámara pasa por el objetivo, atraviesa algunos filtros (depende del modelo de cámara) y llega hasta el sensor de imagen, denominado CCD, que formado por multitud de receptores fotosensibles, llamados "fotodiodos".

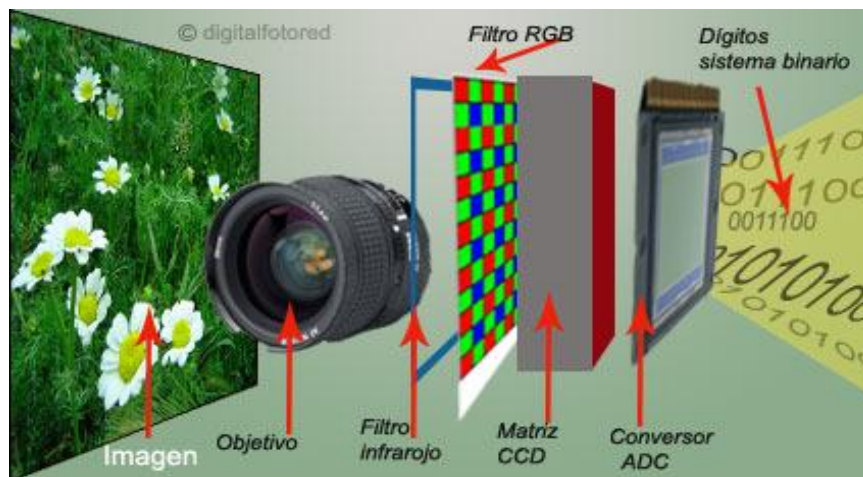


Figura 4. Proceso para la formación de una imagen digital.

Fuente: (Foto Nostra.com, s.f.)

La luz incidente genera una pequeña señal eléctrica en cada receptor. A continuación, esta señal es transformada en datos digitales por el conversor ADC, datos que podemos representar como una serie de cadenas de ceros y unos, denominados dígitos binarios.

Estos números binarios (0, 1), se representan como pequeños cuadraditos, llamados píxeles, que forman un mosaico. Como los píxeles suelen ser muy pequeños, el ojo y el cerebro del observador se encargan de formar una imagen continua, ignorando las diminutas piezas que la forman. (Foto Nostra.com, s.f.)

2. Redes informáticas.

Una red informática es un conjunto de dispositivos interconectados entre sí a través de un medio, que intercambian información y comparten recursos. Básicamente, la comunicación dentro de una red informática es un proceso en el que existen dos roles bien definidos para los dispositivos conectados, emisor y receptor, que se van asumiendo y alternando en distintos instantes de tiempo.

También hay mensajes, que es lo que estos roles intercambian. La estructura y el modo de funcionamiento de las redes informáticas actuales están definidos en varios estándares, siendo el más extendido de todos los modelos TCP/IP, basados en el modelo de referencia o teórico OSI.

2.1. Clasificación.

Considerando el tamaño o la envergadura de una red, podemos clasificarlas de la siguiente manera:

- **WLAN** (Wireless Local Área Network) o red de área local inalámbrica: es una red LAN que emplea medios inalámbricos de comunicación. Es una configuración muy utilizada por su escalabilidad y porque no requiere instalación de cables.

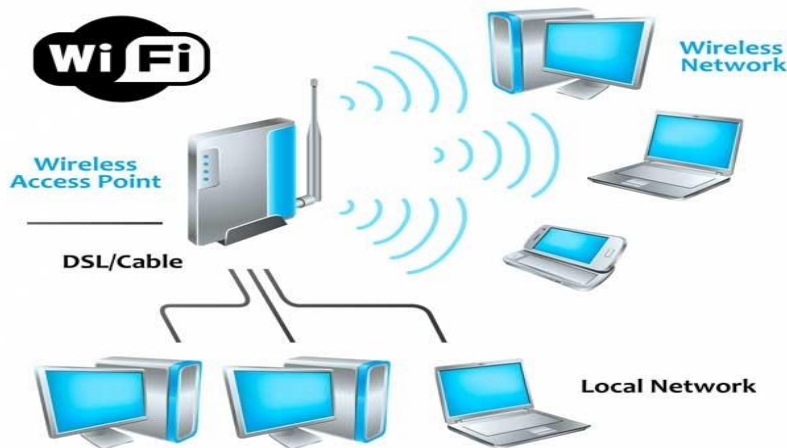


Figura 5. Estructura de una red WLAN.

Fuente: (Quilapan, 2014)

2.2. Velocidades de conexión.

La velocidad a la cual viaja la información en una red está dada por la velocidad máxima que soporta el medio de transporte. Entre los medios más comunes podemos afirmar que la fibra óptica es la más veloz, con aproximadamente 2 Gbps; después le sigue el par trenzado, con 100 Mbps a 1000 Mbps; y por último, las conexiones Wi-Fi, con 54 Mbps en promedio. Las velocidades pueden variar de acuerdo con los protocolos de red utilizados. (¿Qué es una red informática?, 2013)

2.3. Red inalámbrica

Una red inalámbrica, por lo tanto, es aquella que permite conectar diversos nodos sin utilizar una conexión física, sino estableciendo la comunicación mediante ondas electromagnéticas. La transmisión y la recepción de los datos requieren de dispositivos que actúan como puertos.



Figura 6. Esquema de una red inalámbrica.

Fuente: (Tecnología Informática, sf)

Las redes inalámbricas permiten establecer vínculos entre computadoras y otros equipos informáticos sin necesidad de instalar un cableado, lo que supone una mayor comodidad y un ahorro de dinero en infraestructura.

Además de lo expuesto, tendríamos que señalar otra serie importante de ventajas que presenta cualquier red inalámbrica:

- Es muy sencilla de instalar. Y es que, como hemos mencionado anteriormente, no lleva cableado por lo que se evita tener que ir realizando agujeros en las paredes para poder pasar aquel.

- Se convierte en una instalación más elegante precisamente porque no requiere tener cables por todas partes. De esta manera, se garantiza que en absoluto se perjudicará el estilo o la apariencia que tenga la estancia donde se ponga en funcionamiento.
- Permite que puedan estar interconectados un importante número de dispositivos, tanto ordenadores como tablets, teléfonos móviles, periféricos como impresoras o faxes.

Como punto negativo, este tipo de redes suele contar con una seguridad menor ya que, si no se cuenta con una protección eficiente, el ingreso de intrusos es muy probable.

Es importante también tener claro que para poder acometer lo que sería la configuración de una red inalámbrica se hace necesario contar con una serie de elementos que son absolutamente imprescindibles. Nos estamos refiriendo a los siguientes:

- ✓ Un enrutador de tipo inalámbrico.
- ✓ Una conexión a Internet, a ser posible de banda ancha.
- ✓ Adaptadores de red inalámbrica.
- ✓ Un módem.

A partir de tener aquellos dispositivos se podrá comenzar a poner en funcionamiento la mencionada red. Esa es una tarea que requiere acometer pasos imprescindibles tales como colocar el enrutador, reducir lo que son las interferencias, configurar la clave de seguridad que va a tener la red.

De acuerdo al tipo de cobertura, la red inalámbrica puede clasificarse como:

- Wireless Personal Área Network (WPAN)
- Wireless Local Área Network (WLAN)
- Wireless Metropolitan Área Network (WMAN)
- Wireless Wide Área Network (WAN).

La red WPAN es frecuente en la tecnología Bluetooth. No obstante, también dentro de la red WPAN se recurre al uso de tecnologías tales como ZigBee y el infrarrojo para poder poner en funcionamiento lo que es cualquier red inalámbrica que se precie.

Mientras, los sistemas WiFi suelen establecer redes WLAN. Las tecnologías basadas en WIMAX, por su parte, permiten establecer redes WMAN, mientras que las redes WAN se usan con comunicaciones GSM, HSPA o 3G. (Porto J. P., 2014)

Estándares de Wi-Fi

El Wi-Fi, o comúnmente escrito en español como wifi, es un tipo de conexión inalámbrica creado por la Wi-Fi Alliance. Esta alianza es una organización sin ánimo de lucro, formada por empresas como 3Com, Cisco o Motorola. Los principales patrocinadores de esta alianza son las grandes empresas fabricantes de dispositivos que hacen uso del Wi-Fi, como Apple, Samsung, Sony, LG, Cisco, Broadcom, Intel, Qualcomm, Motorola, Microsoft.

El primer estándar fue introducido en el año 1997, y desde entonces hemos visto como una maraña de letras han ido acompañando al subtítulo de cada categoría. Últimamente oímos mucho 'Wi-Fi n', 'Wi-Fi ac'. Pero, ¿qué significa cada letra y qué implica?

Cada letra es uno de los diferentes estándares que ha recibido la certificación Wi-Fi a lo largo de su existencia. Cada una ha supuesto un avance con respecto a la anterior, con mejoras de velocidad o de alcance, en la figura 6 se muestran los diferentes estándares de wifi. (Test de Velocidad, 2016)

Tabla 1. Tabla de los estándares de WiFi.

Estándar	Frecuencia	Velocidad	Punto de Acceso		Modulación
			Alcance Interno	Alcance Externo	
802.11 a	2.4 GHz	11 Mbps	30-90 m	120-460 m	CCK
802.11 b	5.0 GHz	54 Mbps	12-90 m	30-300 m	OFDM
802.11 g	2.4 GHz	54 Mbps	30-90 m	120-460 m	OFDM-CCK
802.11 h	5.0 GHz	54 Mbps	30-90 m	120-460 m	OFDM
802.11 d	Diseñada para uso en países con restricciones de frecuencia.				
802.11 e	Añade a los estándares 802.11b y 802.11a características de QoS y de soporte multimedia, a la vez que mantiene compatibilidad con ellos.				
802.11 i	Está dirigido a batir la vulnerabilidad actual en la seguridad para protocolos de autenticación y de codificación.				
802.11 c	Utilizado para la comunicación de dos redes distintas o de diferentes tipos.				
802.11 f	Es una recomendación para proveedores de puntos de acceso que permite que los productos sean más compatibles.				
802.11 n	Hace uso simultaneo de ambas bandas 2.4 y 5.0 GHz, permite compatibilidad con dispositivos basados en tecnología anteriores.				
802.11 p	Estándar que opera en frecuencia de 5.9 GHz y 6.2 GHz, especialmente indicado para automóviles.				
802.11 v	Permite la comunicación remota de los dispositivos clientes (capacidad de supervisar, configurar y actualizar las estaciones cliente).				

Fuente: Elaboración Propia

3. Plataforma Android

Android. Es un Sistema Operativo además de una plataforma de Software basada en el núcleo de Linux. Diseñada en un principio para dispositivos móviles. Android permite controlar dispositivos por medio de bibliotecas desarrolladas o adaptadas por Google mediante el lenguaje de programación Java.

Google, sin embargo, ha sido quien ha publicado la mayoría del código fuente de Android bajo la licencia de Software Apache, una licencia de software libre y de código abierto a cualquier desarrollador.

Dado que Android está basado en el núcleo de Linux, tiene acceso a sus recursos, pudiendo gestionarlo, gracias a que se encuentra en una capa por encima del Kernel, accediendo así a recursos como los controladores de pantalla, cámara, memoria flash.

3.1. Características

Android es una plataforma de código abierto. Esto quiere decir, que cualquier desarrollador puede crear y desarrollar aplicaciones escritas con lenguaje C u otros lenguajes y compilarlas a código nativo de ARM (API de Android). Una parte importante al entrar al mundo Android es conocer que significa cada uno de los términos o palabras que se usan.



Figura 7. Características de Android.

Fuente: (Basterra, Berteia, Borello, Castillo y Venturi., sf)

- Framework de aplicaciones: permite el reemplazo y la reutilización de los componentes.
- Navegador integrado: basado en los motores open Source Webkit.
- SQLite: base de datos para almacenamiento estructurado que se integra directamente con las aplicaciones.
- Multimedia: Soporte para medios con formatos comunes de audio, video e imágenes planas (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF).
- Máquina virtual Dalvik: Base de llamadas de instancias muy similar a Java.
- Telefonía GSM: dependiente del terminal.
- Bluetooth, EDGE, 3g y Wifi: dependiente del terminal.
- Cámara, GPS, brújula y acelerómetro: Dependiente del terminal.
- Pantalla táctil.

3.2. Arquitectura de Android

La arquitectura interna de la plataforma Android, está básicamente formada por 4 componentes:

Aplicaciones

Todas las aplicaciones creadas con la plataforma Android, incluirán como base un cliente de email (correo electrónico), calendario, programa de SMS, mapas, navegador, contactos, y algunos otros servicios mínimos. Todas ellas escritas en el lenguaje de programación Java. Ejemplo de ello Alerta flash.

Aunque el lenguaje de programación principal es Java, actualmente también permite programar aplicaciones en Kotlin, que es compatible con Java, con lo que se pueden realizar aplicaciones mixtas con los dos lenguajes. En caso de querer reutilizar código escrito en C/C++, se puede introducir mediante Android NDK (Native Development Kit).

Framework de Aplicaciones

Todos los desarrolladores de aplicaciones Android, tienen acceso total al código fuente usado en las aplicaciones base. Esto ha sido diseñado de esta forma, para que no se generen cientos de componentes de aplicaciones distintas, que respondan a la misma acción, dando la posibilidad de que los programas sean modificados o reemplazados por cualquier usuario sin tener que empezar a programar sus aplicaciones desde el principio.

Librerías

Android incluye en su base de datos un set de librerías C/C++, que son expuestas a todos los desarrolladores a través del framework de las aplicaciones Android System C library, librerías de medios, librerías de gráficos, 3D, SQLite.

Runtime de Android

Android incorpora un set de librerías que aportan la mayor parte de las funcionalidades disponibles en las librerías base del lenguaje de programación Java. La Máquina Virtual está basada en registros, y corre clases compiladas por el compilador de Java que anteriormente han sido transformadas al formato. Dex (Dalvik Executable) por la herramienta dx.

El uso de Dalvik fue utilizado hasta la versión 4.4.3 de Android. A partir de la versión 5 de Android (se introdujo en la 4.4), se utiliza ART (Android Runtime) como entorno de ejecución, que compila durante la instalación de una aplicación, el Java bytecode. (Vilchez, 2009)



Figura 8. Arquitectura Android.

Fuente: (Basterra, Berteá, Borello, Castillo y Venturi., sf)

4. Sensores de proximidad.

Son un dispositivo que detecta la presencia o ausencia de un objeto, o ciertas propiedades de ese objeto, y provee retroalimentación sin entrar en contacto con ellos. Los sensores miden la magnitud física y la convierten en una señal que puede ser leída fácilmente por el usuario o un instrumento electrónico simple. Hay varios sensores que trabajan en el campo de la ciencia hoy en día. Algunos sensores están hechos para ser utilizados en la vida cotidiana, mientras que hay muchos sensores que han estado facilitando los seres humanos en la investigación y el campo de la ingeniería.

4.1. Tipos de sensor de proximidad.

Los siguientes son los principales tipos de sensores de proximidad.

❖ Sensores de proximidad capacitivos

Los sensores capacitivos se pueden utilizar para detectar objetos de metal, así como casi todos los otros materiales. Detectan niveles de líquidos, mezclas químicas y el nivel del líquido, etc. de cualquier objeto dentro del rango. Se utilizan generalmente para fines industriales. Estos sensores son capaces de detectar materiales dieléctricos como el plástico, de vidrio o de fluidos. Les permite averiguar el nivel de diferentes materiales en sus composiciones.



Figura 9. Sensores de proximidad capacitivos.

Fuente: (Allbiz, sf)

Gran parte de un sensor capacitivo se compone de dos electrodos metálicos que funcionan como un capacitor. Estos electrodos se colocan en el bucle de realimentación de un oscilador de alta frecuencia. Cuando no hay ningún objeto, el rango de capacidad del sensor disminuye y la amplitud de oscilación se vuelve baja. Un objetivo que está en la cara de un sensor añade capacitancia que aumenta posteriormente la amplitud de la oscilación. Se calcula por un circuito de evaluación que obtiene a su vez en la salida.

❖ **Sensores de proximidad inductivos**

Los sensores inductivos describen la naturaleza de los metales y los sentidos si son o no ferrosos. Tienen amplia gama de uso como el descubrimiento de las partes que faltan, llegar a percibir la ubicación o para contar el número de objetos. Son un mejor reemplazo de interruptores que los hace más factible. Ellos tienen velocidades relativamente más altas y son más fiables.



Figura 10. Sensor de proximidad inductivo.

Fuente: (Nylam Mechatronic - Perú, sf)

Un campo electromagnético de alta frecuencia se extiende alrededor de un sensor inductivo. Cuando hay algo presente dentro de la gama, la corriente es inducida en el sensor. Esto produce alteraciones en el oscilador del sensor y se pierde algo de energía. Los sensores inductivos generan un campo electromagnético de alta frecuencia. Se construyen típicamente utilizando una bobina y un núcleo de ferrita. Cuando un objeto pasa a través del campo magnético del sensor, la corriente inducida en la superficie del objetivo cambia las características del oscilador que genera el campo, haciendo que se pierda energía. El sensor identifica la pérdida de energía; esto convierte la salida de encendido o apagado. (Sensores de proximidad y sus aplicaciones, s.f.)

VII. Análisis del sistema de aparcamiento actual vs sistema de aparcamiento propuesto.

A medida que los dispositivos electrónicos fueron agregados en los vehículos y con ello la tecnología de punta fue invadiendo los mismos surgieron los sistemas de aparcamiento. Estos son sistemas electrónicos que ayudan al conductor a realizar maniobras de estacionamiento asistiéndole en los espacios más ajustados o reducidos para así evitar que el vehículo sea abollado o rayado con el otro de su alrededor. (Motor y Racing, s,f)

1. Sistema de aparcamiento actual.

Los sistema de aparcamiento son elementos de asistencia al conductor para guiarlo al momento de estacionar, de esta forma evitará cualquier tope con otro vehículo o con el andén. Los sensores se instalan en los parachoques de los automóviles, tanto delanteros como traseros y sirven para detectar cualquier obstáculo. (blogdelaautoescuela.com, 2016)

Dichos sensores disponen de tecnología de ultrasonido detectando la distancia restante entre el vehículo involucrado y los de su alrededor y su fin es emitir un alerta ya sea óptica o acústica con el propósito de que el conductor esté consiente y evite colisionar. Es importante señalar que los mismos también pueden ser instalados en coches de antigua generación, por lo que esta tecnología no se limita a los coches de última generación. (Motor y Racing, s,f)

1.1. Ventajas del sistema de aparcamiento

Entre las principales ventajas de su uso encontramos:

Tabla 2. Ventajas del sistema de aparcamiento.

Ventajas	Descripción
Ayuda a maniobrar en los ángulos muertos.	Hay puntos ciegos de visibilidad cuando se maniobra a través de los espejos, por lo que en estos casos, los sensores de aparcamiento son ideales para informar al conductor sobre cualquier obstáculo que se presente en los ángulos muertos.
Sensores de aparcamiento con señal acústica.	Los sensores de aparcamiento cuentan con una señal acústica que se activa al momento de estar cerca un obstáculo, dependiendo del tipo de sensor, la misma va incrementando la intensidad de su sonido a medida que se está más cerca del objeto con el que se podría topar.
Radio de acción amplio.	Los sensores de aparcamiento comunes tienen un radio de acción de 130° y los más modernos cuentan con un radio de acción de 160°, así que permiten una visión central y periférica que los espejos retrovisores no logran brindar al conductor.
Son muy discretos.	Los sensores están diseñados para ser muy discretos y no desequilibrar el diseño del automóvil, tienen el diámetro de una moneda grande y pueden ser removidos cuando se desee sin dañar los parachoques del automóvil.
Fáciles de instalar.	La instalación es muy sencilla, solo basta leer las instrucciones y listo, sin embargo, si no te consideras muy diestro en esto, cualquier electricista lo podrá hacer.

Fuente: (blogdelaautoescuela.com, 2016)

2. Sistema de aparcamiento propuesto

El sistema de aparcamiento propuesto tiene la finalidad de ser una herramienta útil para todos los conductores nicaragüense, facilitando las maniobras de los vehículos en los puntos ciegos al momento de retroceder. Dicho sistema estará constituido de forma novedosa porque se utilizara un dispositivo móvil donde se estará transmitiendo el video en tiempo real de la parte trasera del vehículo.

Las ventajas del sistema de adquisición de video y medición de distancia durante el retroceso de un vehículo en un dispositivo móvil son las siguientes:

1. Sistema de bajo costo.
2. Por ser un sistema manual permite al usuario monitorear la parte trasera del vehículo de manera constante.
3. El rango de monitoreo a través del dispositivo móvil es de 5 metros.
4. El video en tiempo real que se trasmite en el dispositivo móvil posee buena calidad.

3. Sistema de aparcamiento actual vs sistema de aparcamiento propuesto.

Cabe destacar que ambos sistemas son novedosos al permitir reducir los índices de colisiones, por poseer ventajas que facilitan a los usuarios la utilización de los mimos.

Los sistemas de aparcamiento actual son automáticos, pues al momento de retroceder este se activa, no obstante el sistema de aparcamiento propuesto se acciona de manera manual.

Otra ventaja que posee el sistema de aparcamiento actual es su sistema integrado el cual se acciona solo al momento de retroceder, mientras que el sistema propuesto no es un sistema integrado y para su utilización se debe acceder a la app sin embargo el usuario puede utilizarlo en cualquier momento.

El costo del sistema de aparcamiento actual ya viene incluido en la adquisición del vehículo, al contrario el sistema de aparcamiento propuesto tiene un costo accesible.

VIII. Diseño Metodológico.

A. Tipo de estudio:

Basándose en el tema, contenido y el alcance que el presente proyecto tiene, es de tipo cuantitativo, ya que se utilizaron encuestas donde se recabaron datos para conocer la perspectiva de la población en relación al tema en estudio.

B. Universo:

Está conformado por estudiantes, personal docente y administrativo que pertenecen a la Universidad Nacional Autónoma de Nicaragua (UNAN-Managua).

C. Muestra:

Seleccionada con el requisito de poseer un vehículo, el número de personas encuestadas es equivalente a 21 personas.

El tipo de muestreo utilizado es por conveniencia, la muestra fue seleccionada por el acceso y la disposición que tenían para aportar información útil para el desarrollo de este proyecto.

D. Definición y operacionalización de variables (MOVI)

Objetivo Específico	Variable conceptual	Indicadores
Realizar análisis del sistema de aparcamiento actual vs sistema de aparcamiento propuesto.	Comparación del sistema de aparcamiento actual y el sistema propuesto.	❖ Ventajas del sistema de aparcamiento actual y sistema de aparcamiento propuesto.
Describir los componentes que integran el sistema de adquisición de video y medición de distancia durante el retroceso de un vehículo en un dispositivo móvil.	Componentes que integran el sistema de adquisición de video y medición de distancia.	<ul style="list-style-type: none"> ❖ Selección del componente que se acople al sistema. ❖ Costo de los componentes que integran el sistema.
Diseñar el sistema de adquisición de video y medición de distancia del vehículo.	Hacer un sistema de adquisición de video y medición de distancia.	<ul style="list-style-type: none"> ❖ Diagramas Esquemáticos. ❖ Software a utilizar. ❖ Programacion del sistema.
Demostrar el funcionamiento del sistema de adquisición de video y medición de distancia mediante un prototipo a escala en un ambiente controlado.	Pruebas de funcionamiento del sistema diseñado.	<ul style="list-style-type: none"> ❖ Pruebas de programación de software Arduino. ❖ Instalación del servidor de Video UV4L en Raspberry. ❖ Configuración Access Point en Raspberry. ❖ Pruebas de programación.

E. Métodos, técnicas e instrumentos de recolección de datos.

Para la recolección de datos se utilizó la encuesta como instrumento, misma que apporto datos significativos para el desarrollo de este proyecto.

Encuesta: Consiste en obtener información de los sujetos de estudio, proporcionados por ellos mismos, sobre opiniones, conocimientos, actitudes o sugerencias. (Canales, Alvarado y Pineda, 1994)

F. Procedimientos para la recolección de datos e información

La encuesta previamente elaborada fue aplicada a estudiantes, personal docente y personal administrativo que cuenta con la posesión de un vehículo pues la misma proporciona información acerca de si realmente existe la necesidad de diseñar un nuevo sistema de adquisición de video y medición de distancia durante el retroceso de un vehículo en un dispositivo móvil y la aceptación que este sistema pueda tener dentro de la población.

La encuesta cuenta con 8 preguntas claves que van desde si el encuestado posee un vehículo con sistema ya implementado hasta la aceptación que puede tener la creación de un nuevo sistema para vehículos que no poseen esta herramienta.

G. Plan de análisis y procesamiento

Puesto que el enfoque de este proyecto es de tipo cuantitativo los datos obtenidos de la encuesta serán reflejados mediante gráficos de pastel que señalan el porcentaje de aceptación o no aceptación de acuerdo a la pregunta en cuestión.

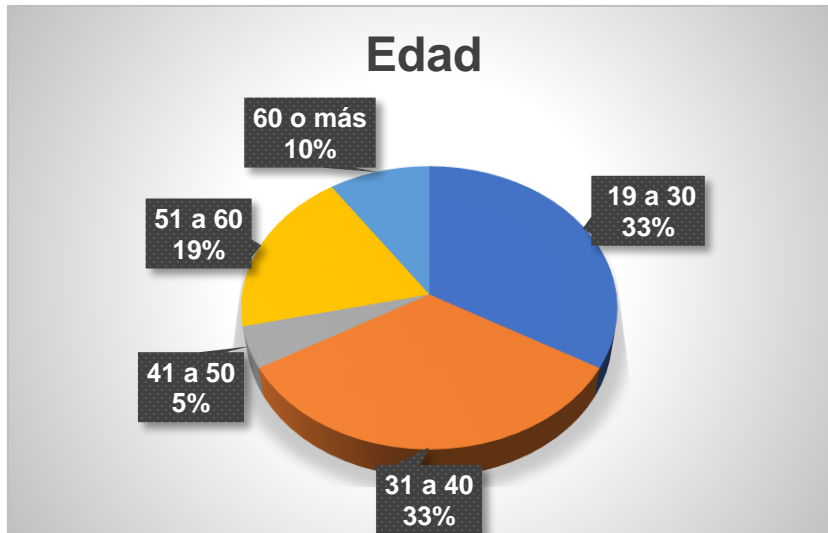


Figura 11. Grafica N° 1 Edad.

Fuente: Elaboración propia

En la figura 11 se observa que la edad varia de 19 a 68 años reflejando 33% correspondiente a una edad de 19 a 30 años, un 33% edad de 31 a 40 y situando un 19% a una edad promedio de los 51 a 60, quedando un 10% en edad de 60 a más y un 5% a 41 a 50.

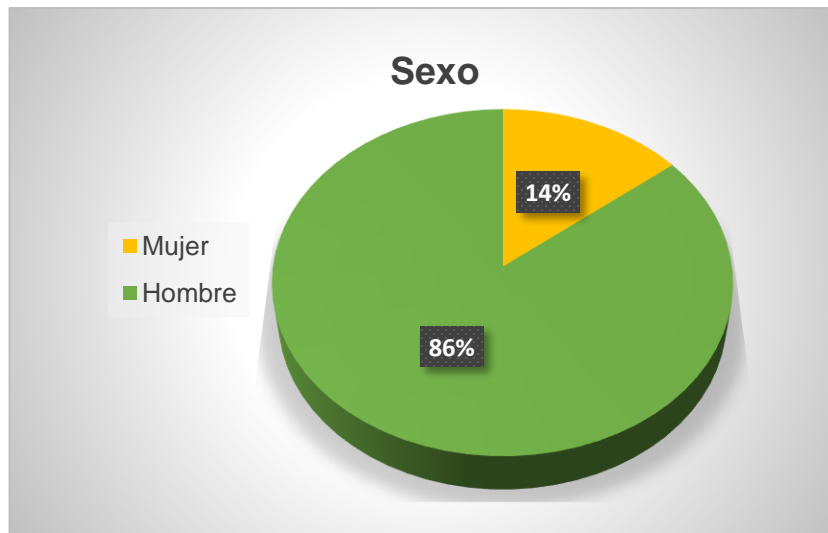


Figura 12. Gráfico N° 2 Sexo.

Fuente: Elaboración propia

El 86% de las personas encuestadas corresponden al sexo masculino y el 14% son de sexo femenino.

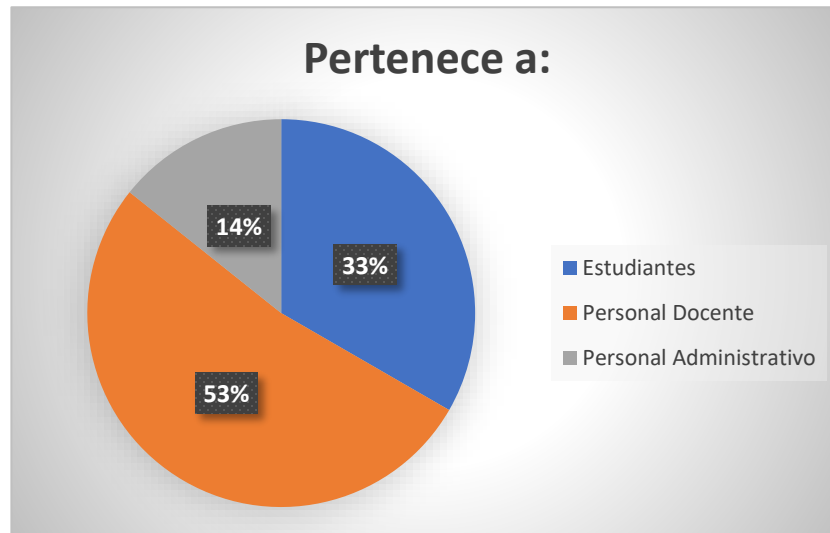


Figura 13 Gráfico N°3 Pertenece.

Fuente: Elaboración propia.

La muestra de la población encuestada pertenecen al 33% estudiantes, 53% personal docente y 14% personal administrativo.

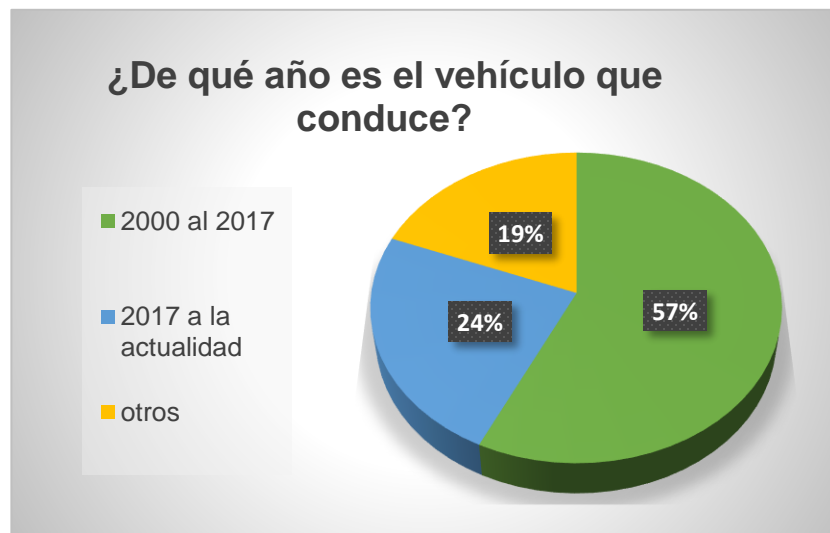


Figura 14. Gráfico N° 4 Año del vehículo que conduce.

Fuente: Elaboración propia.

Al preguntar ¿De qué año es el vehículo que conduce? Se obtuvieron los siguientes datos el 57% conduce vehículos del año 2000 al 2017, el 24% del 2017 a la actualidad y un 19% respondió que otros. En los cuales destacan modelos anteriores al 2000.



Figura 15. Gráfico N° 5 Sistema de ayuda para el retroceso del vehículo.

Fuente: Elaboración propia.

Según datos obtenidos de la encuesta realizada el 67% de la muestra expresa que su vehículo no posee un sistema de ayuda para el retroceso, mientras que el 33% posee un vehículo con sistema de ayuda que correspondería a autos modernos que ya cuentan con esta herramienta.

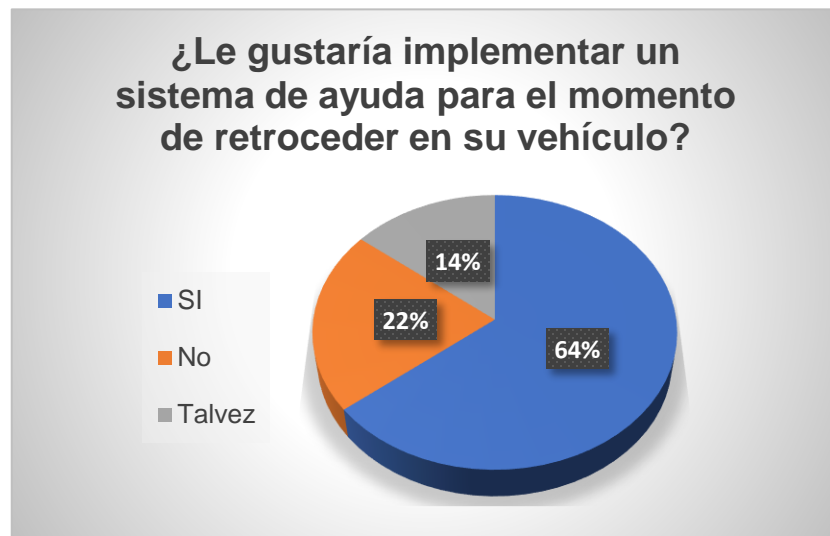


Figura 16. Gráfico N° 6 Implementación de un sistema de ayuda para el momento de retroceder en su vehículo.

Fuente: Elaboración propia.

El 64% de la muestra opina que le gustaría implementar un sistema de ayuda en su vehículo, el 22% afirma que no y un 14% respondió que talvez.

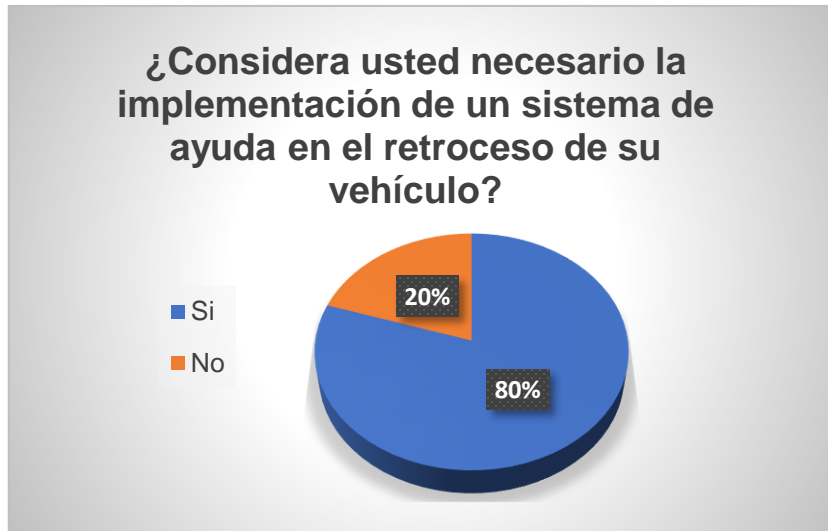


Figura 17. Gráfico N° 7 ¿Considera necesario la implementación de un sistema de ayuda en el retroceso de un vehículo?

Fuente: Elaboración propia.

El 80% considera necesario la implementación de un sistema de ayuda en el retroceso de su vehículo, y un 20% considera lo contrario.

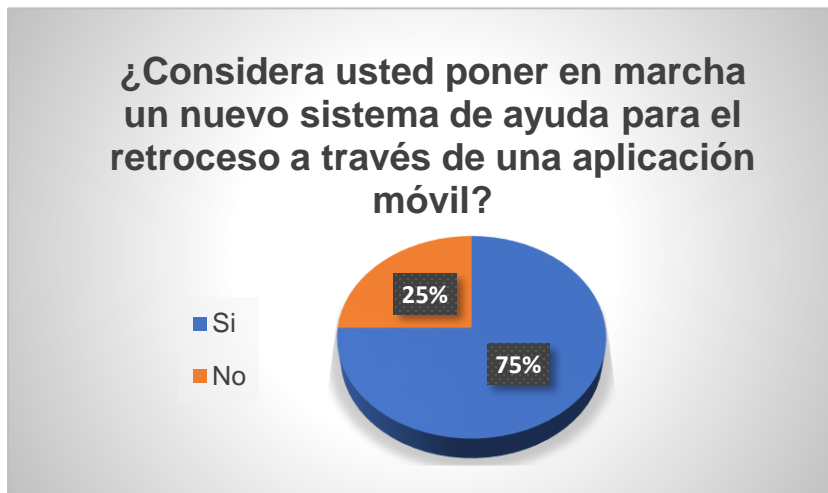


Figura 18. Gráfico N° 8 Un nuevo sistema de ayuda para el retroceso a través de una aplicación móvil.

Fuente: Elaboración propia.

El 75% de la muestra considera que pondría en marcha un nuevo sistema de ayuda para el retroceso del vehículo a través de una aplicación móvil, el 25% considera que no.

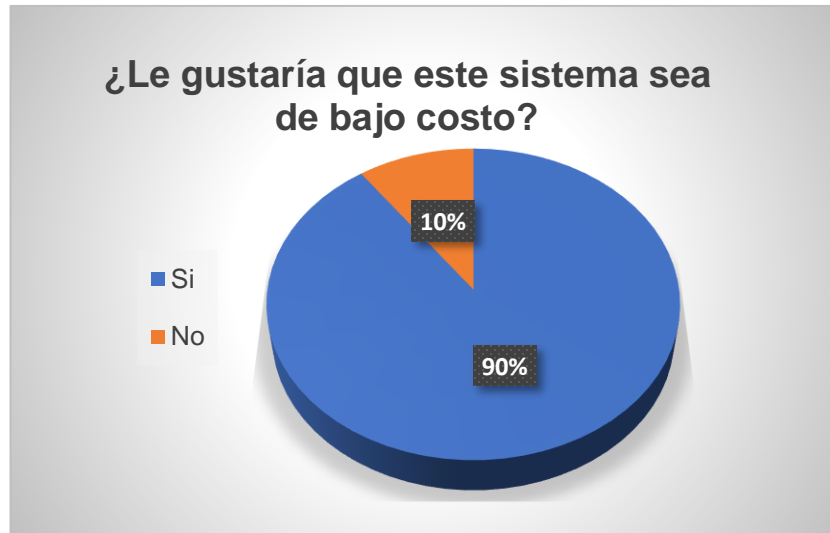


Figura 19. Gráfico N° 9 ¿Le gustaría que este sistema sea de bajo costo?

Fuente: elaboración propia.

Al 90% de la muestra le gustaría que este sistema sea de bajo costo y el 10% no considera importante el precio.

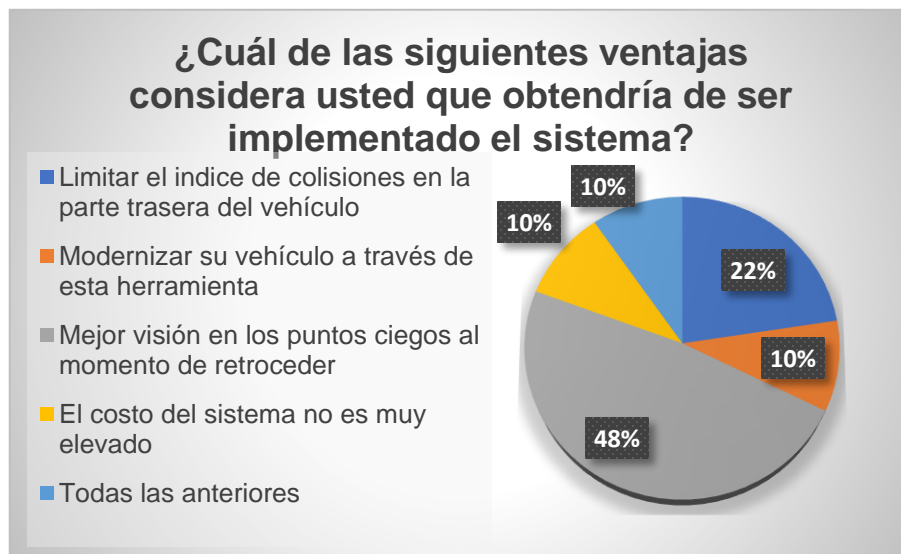


Figura 20. Gráfico N° 10 Ventajas de ser implementado el sistema.

Fuente: Elaboración propia.

La encuesta aplicada brindaba una serie de ventajas para que el encuestado opinara según el criterio que más estimara conveniente. Al ser implementado el sistema de adquisición de video y medición de distancia de retroceso del vehículo a través de un dispositivo móvil se lograría;

Limitar el índice de colisiones en la parte trasera del vehículo. Un 22% de los encuestados la seleccionó, un 10% considera que este sistema modernizaría su vehículo, el 48% destaca que se obtendría una mejor visión en los puntos ciegos al momento de retroceder, un 10% ve factible el costo debido a que no es muy elevado, y el restante 10% considera que el sistema cumple con todas la ventajas anteriormente mencionada.

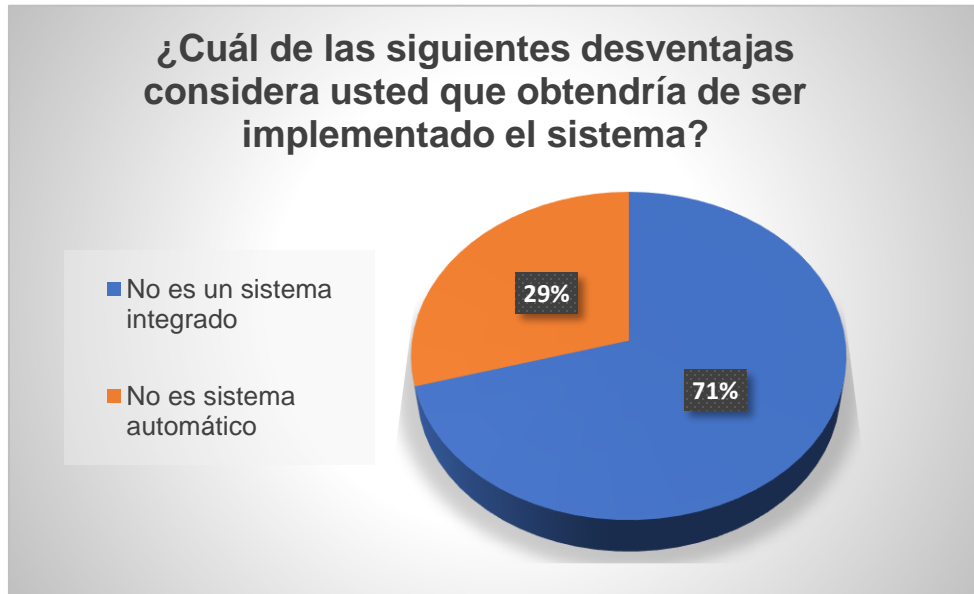


Figura 21. Gráfico N° 11 Desventajas de ser implementado el sistema.

Fuente: Elaboración propia.

La encuesta concluyo con la interrogante ¿Cuál de las siguientes desventajas considera usted que obtendría de ser implementado el sistema? Evidenciando a un 71% que opina que no es un sistema integrado y a un 29% que no es un sistema automático.

Los datos obtenidos mediante la encuesta demuestran la necesidad de realizar un sistema de adquisición de video y medición de distancia durante el retroceso del vehículo a través de un dispositivo móvil.

IX. Hardware

2. Diseño del sistema de adquisición de video y medición de distancia durante el retroceso de un vehículo en un dispositivo móvil.

El siguiente acápite abordara los procesos que conforman el diseño del sistema de adquisición de video y medición de distancia durante el retroceso de un vehículo en un dispositivo móvil.

1.1. Diseño Electrónico.

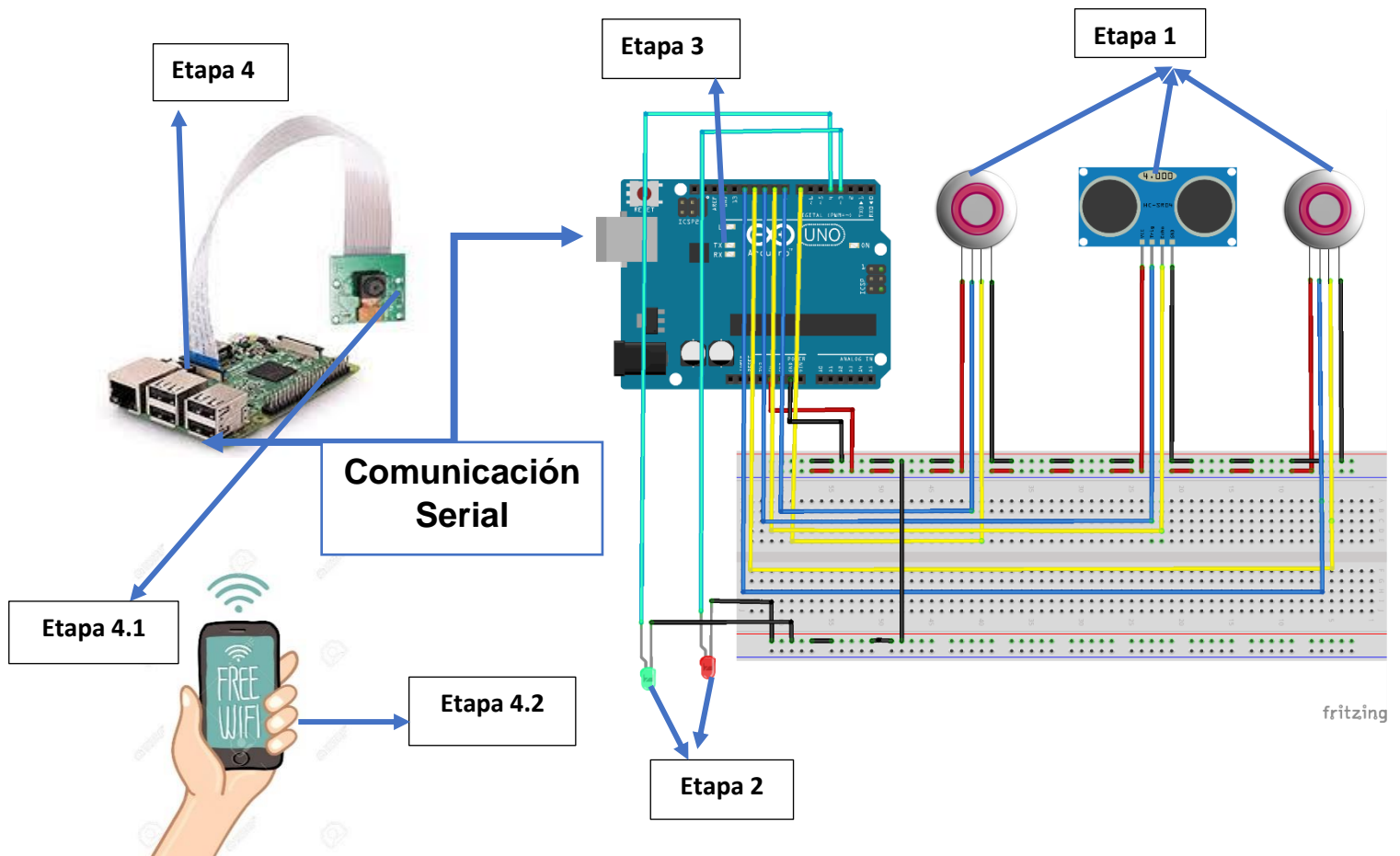


Figura 22. Principales Componentes del Sistema.

Fuente: Elaboración propia

Etapa 1. Medicion de distancia.

En esta etapa, éste circuito se encarga de detectar obstáculos cercanos, para lo cual se utilizan tres sensores de proximidad, donde con la combinación de estos se logra no solo detectar la presencia de obstáculos sino que además permiten conocer la proximidad (distancia) hacia los mismos. Por lo tanto se utilizan los siguientes sensores:

✚ El sensor JSN-SR04T.

Es un sensor de distancia que utiliza ultrasonido (sonar) para determinar la distancia de un objeto en un rango de 25 a 450 cm. Destaca por su pequeño tamaño, bajo consumo energético, buena precisión y especialmente por su resistencia al Agua.



Figura 23. Sensor JSN-SR04T.

Fuente: (coorporativo, sf)

El sensor trabaja con ultrasonido y contiene toda la electrónica encargada de hacer la medición. El funcionamiento del sensor es el siguiente: se emite un pulso de sonido (TRIG), se mide la anchura del pulso de retorno (ECHO), se calcula la distancia a partir de las diferencias de tiempos entre el Trig y Echo. El funcionamiento no se ve afectado por la luz solar o material negro (aunque los materiales blandos acústicamente como tela o lana pueden ser difícil de detectar).

Perfecto para aplicaciones donde el sensor estará expuesto a la intemperie, utilizado en automóviles para medir distancia de colisión/parqueo.

Especificaciones técnicas.

- ✓ Modelo: JSN-SR04(FZ1655)
- ✓ Voltaje de Operación: 5V DC
- ✓ Corriente de trabajo: 30mA
- ✓ Rango de detección: 25cm-450cm
- ✓ Precisión: puede variar entre los 3mm o 0.3cm
- ✓ Frecuencia de emisión acústica: 40KHz
- ✓ Duración mínima del pulso de disparo (nivel TTL): 10 μ s.
- ✓ Tiempo mínimo de espera entre una medida y el inicio de otra 20 ms.
- ✓ Ángulo de detección: menor a 50°
- ✓ A prueba de agua (parte delantera)
- ✓ Diámetro: 22mm
- ✓ Longitud: 17mm
- ✓ Temperatura de trabajo: -10°C hasta 70°C

Pines de conexión.

- ✓ VCC (+5V DC)
- ✓ TRIG (Disparo del ultrasonido)
- ✓ ECHO (Recepción del ultrasonido)
- ✓ GND (0V)

✚ Sensor de ultrasonidos HC-SR04.

Dentro del mundo Arduino, el emisor/receptor de ultrasonidos HC-SR04 es quizás uno de los complementos más reconocibles. Este sensor se emplea en todo tipo de proyectos, siendo su principal uso el de reconocer obstáculos, aunque debido su buena resolución, también se emplea para medir la distancias a objetos.

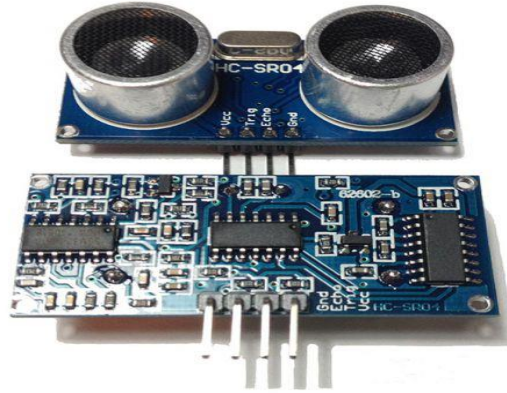


Figura 24. Sensor de ultrasonidos HC-SR04.

Fuente: (Raúl Diosdado, 2014-2018)

Lo más singular del sensor de ultrasonidos HC-SR04 es quizás su "par de ojos", estos no son más que un emisor y un receptor de ultrasonidos que trabajan a una frecuencia de 40KHz (una frecuencia inaudible para las personas).

Este sensor consta de 4 pines, alimentación (Vcc), un disparador (Trig), el receptor (Echo) y masa (GND). Para el correcto funcionamiento del sensor es necesario el conectar estos 4 pines.

Las características básicas de este sensor y que se deben de tener en cuenta al trabajar con él son:

Características	
Alimentación	+5v DC
Frecuencia de trabajo	40 KHz
Consumo (suspendido)	< 2mA
Consumo (trabajando)	15mA
Ángulo efectivo	< 15°
Distancia	2cm a 400cm *
Resolución	0.3 cm

Figura 25. Tabla de especificaciones del sensor HC-SR04.

Fuente: (Raúl Diosdado, 2014-2018)

Como funciona.

El principio en el que se basa su funcionamiento es muy sencillo, tan solo hay que generar una onda sónica en el emisor mediante un pulso en la patilla que pone "Trig" (Trigger o disparador), esta onda al encontrarse con algún obstáculo rebotará, volviendo al sensor y siendo registrada por el receptor, traduciéndose esta en un pulso en la patilla "Echo".

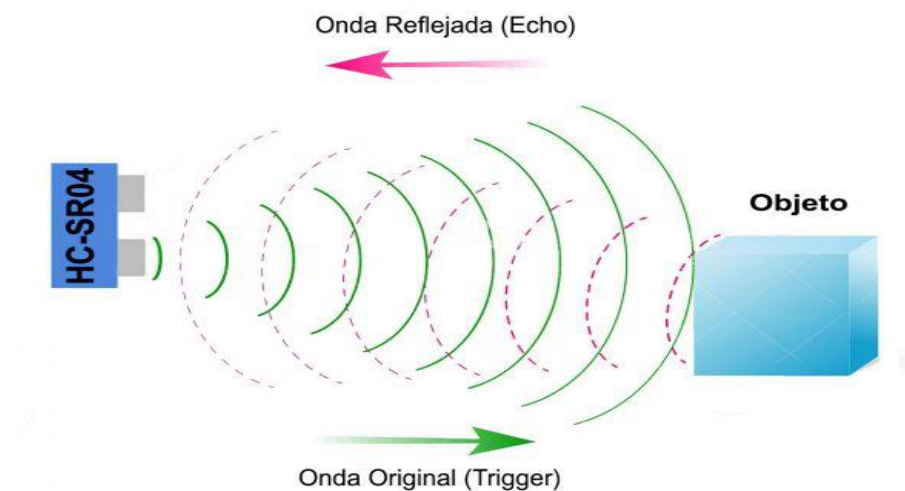


Figura 26. Envío y recepción del pulso ultrasónico.

Fuente: (Raúl Diosdado, 2014-2018)

Con esto podemos hacer dos cosas, detectar un obstáculo esperando simplemente que Arduino reciba un "Echo" o contar el tiempo que transcurre desde que se manda el pulso por el Trigger hasta que se recibe, de esta forma, y conociendo cual es la velocidad del sonido, podemos determinar de forma muy sencilla la distancia exacta a la que se encuentra el objeto en el que está rebotando la señal.

Para aclarar un poco el factor de multiplicación que vamos a introducir en Arduino, basta con decir que la velocidad es igual al espacio dividido por el tiempo que se tarda en recorrer dicho espacio. La velocidad del sonido es conocida (343m/s) y el tiempo lo vamos a determinar, como el tiempo que transcurre desde que efectuamos el disparo hasta que recibimos el eco.

Incertidumbre angular.

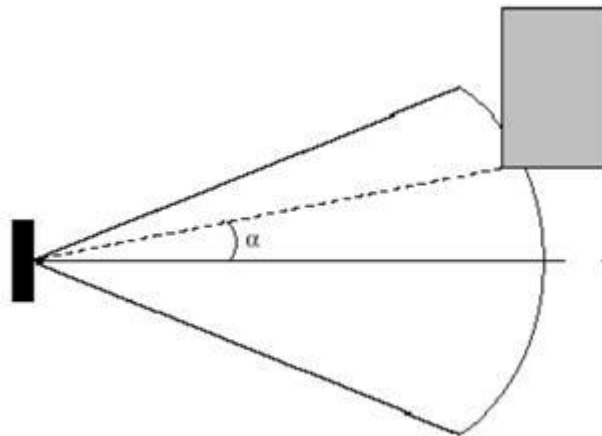


Figura 27. Incertidumbre angular del sensor HSR04.

Fuente: (Javier, 2013)

Otra fuente de error muy común son los llamados falsos ecos. Estos son generados por diversas razones, partiendo desde el caso en que la onda generada por el emisor puede reflejarse varias veces en diversas superficies antes de ser captada por el receptor. Este fenómeno implica una falsa lectura del sensor ya que el viaje de la onda es mucho más largo, dándonos como resultado una distancia mucho mayor hacia un obstáculo cercano.

Un error muy común al utilizar varios sensores al mismo tiempo es el llamado crosstalk, el cual se produce cuando un sensor emite un pulso de ultrasonido y este es detectado por otro sensor, en donde se distorsionan totalmente las mediciones realizadas. Esto se observa con claridad en ante el funcionamiento de dos sensores al mismo tiempo.

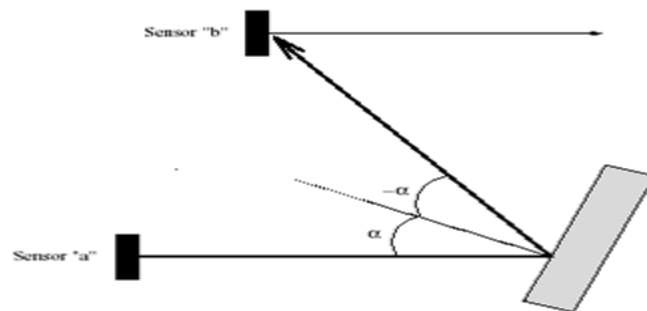


Figura 28. Error producido por varios sensores ultrasónico (crosstalk).

Fuente: (Javier, 2013)

Además de los problemas ya señalados anteriormente se muestran a continuación los aspectos a tener en cuenta al momento de diseñar el sistema de detección de distancias basadas en ultrasonidos:

- La posición real del objeto es desconocida, únicamente se puede conocer la distancia hacia el mismo, mas no la medida angular.
- La dirección de la onda reflejada depende del ángulo de incidencia.
- Al tener ángulos de incidencia altos, se incrementa la posibilidad de que los pulsos se pierdan, es decir, no retornen hacia el receptor.

Si se utiliza varios sensores es necesario considerar una separación adecuada de los mismos para evitar falsos ecos.

La conexión de los sensores al microcontrolador Arduino uno se abordaran en la **etapa 3**.

Etapa 2. Indicadores.

Led rojo

Este indicador se encargara de mostrarle al usuario que esta encendido el sistema.

Led Verde

Este indicador tipo led es el que se encarga de decirle al usuario que estan funcionando los tres sensores ultrasónicos donde el led trabaja de la siguiente forma:

1. **Led verde encendido** indica que el sensor 1 ya envió los datos de medición al microcontrolador (Arduino UNO PRO) y cuando ya se realizó esta función se apaga el led. También informa al usuario que está ocurriendo una comunicación entre el Arduino Uno Pro con Raspberry 3 B+.

De la misma forma funciona en el caso del sensor 2 y 3. Por tanto, se creó un bucle infinito que va de la mano con la medición de los sensores ultrasónicos.

La conexión de los indicadores led (rojo, verde) se estarán describiendo en el siguiente apartado (**etapa 3**).

Etapa 3. Microcontrolador Arduino Uno Pro

Arduino Uno

El Arduino Uno es una placa electrónica que en este sistema se encarga de controlar los sensores ultrasónicos, para esto la placa utilizando sus entradas y salidas digitales genera los pulsos necesarios para el funcionamiento de los sensores.

Esta es una plataforma que permite conectar el mundo físico con el mundo virtual, su principal ventaja es ser una placa electrónica de hardware libre basada en un microcontrolador de la familia de Atm el AVR (Atmega 328), además de presentar un entorno de desarrollo amigable con el usuario apoyado en el lenguaje C que facilita el desarrollo de programas.

Tabla 3. Características del Arduino Uno.

Características del Arduino Uno	
Microcontrolador	ATmega328.
Voltaje Operativo	5V
Voltaje de Entrada (Recomendado)	7-12V
Pines de Entradas/Salidas Digital	14 (De las cuales 6 son salidas PWM).
Pines de Entradas Análogas	6
Memoria Flash	32 KB (ATmega328) de los cuales 0,5 KB es usado por Bootloader.
SRAM	2 KB (ATmega328)
Velocidad del Reloj	16 MHZ.

Fuente: Elaboración propia



Figura 29. Placa Electrónica Arduino Uno.

Fuente: (Guerrero, 2014)

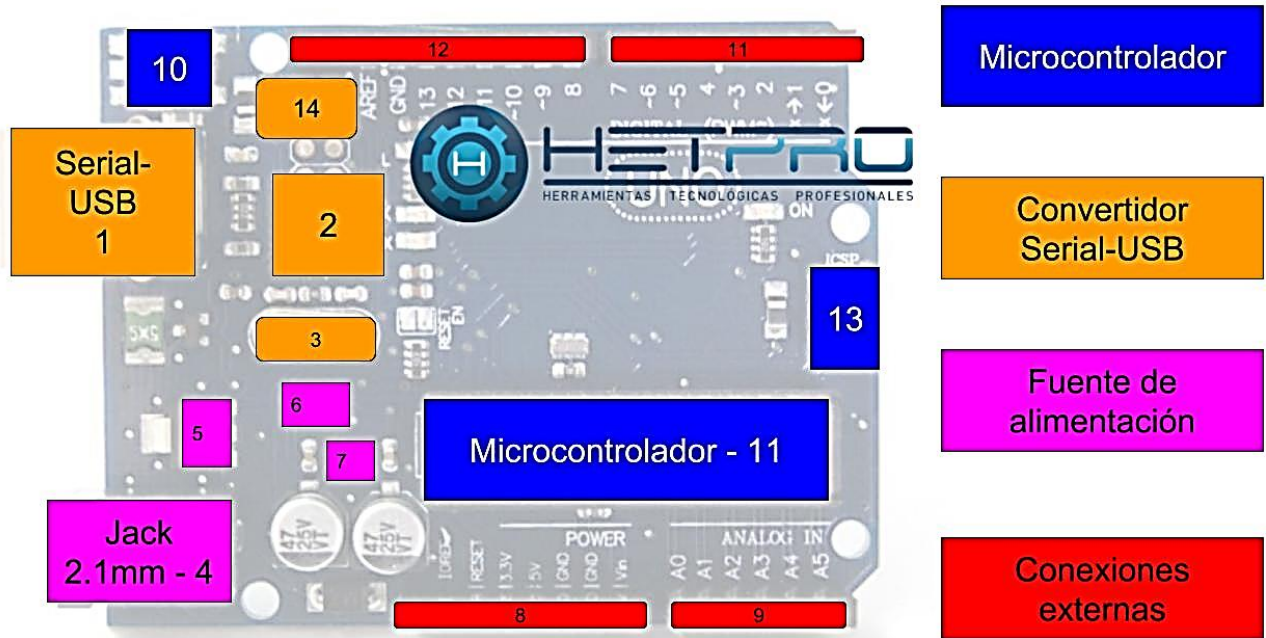


Figura 30. Diagrama de bloques de las partes de Arduino.

Fuente: (Marmolejo, 2017.)

En la figura 32 se detallan las partes del arduino las cuales corresponden a:

- 1. Conector USB Serial:** Permite la conexión/programación y alimentación vía un puerto serial-USB tipo-B. Este conector puede suministrar un voltaje de corriente directa de 5Vdc a una corriente de máximo 500mA. La salida del voltaje de alimentación de este conector pasa por un fusible de 500mA. La corriente también está limitada por la capacidad de corriente que pueda suministrar el puerto USB.
- 2. Chip electrónico ATmega16U2:** Es un microcontrolador que se encarga de programar y comunicar a la computadora con el Arduino a través de un convertidor de protocolo serial-TTL a USB de baja velocidad. En Algunos modelos anteriores, este circuito es reemplazado por un ATMEGA8U2.

3. **Oscilador de cuarzo de 16Mhz:** Proporciona una fuente de oscilación estable al microcontrolador ATmega16U2.
4. **Jack-2.1mm:** Si se requiere de usar más corriente que la que puede proporcionar el conector USB, una opción es alimentar al Arduino vía el Jack-2.1mm y un regulador que proporciona una salida entre 7Vdc y 12Vdc. La recomendación es usar un regulador de voltaje de 9V-1A. Si se usa un voltaje mayor es muy posible que el regulador interno se sobrecaliente.
5. **Regulador 5Vdc-1A:** Cuando se desea utilizar una fuente externa, este circuito regula el voltaje para proporcionarle al Arduino 5Vdc. Se puede usar al mismo tiempo el Jack-2.1mm y el conector USB. Existe un circuito en la tarjeta Arduino que permite hacer el cambio de alimentación para activar la del Jack-2.1mm.
6. **Circuito que permite activar la fuente proveniente del conector USB o del Jack-2.1mm:** Cuando no se encuentra una fuente externa, se activa el conector USB.
7. **Regulador de 3.3Vdc 150mA:** Es un regulador interno que usa la salida de 5Vdc del regulador principal para proporcionar un voltaje que muchos sensores como el NRF24L01+ requiere para funcionar, sin la necesidad de un regulador adicional.
8. **Conector de alimentación:** Este conector Header-hembra incluye conexiones a GND, Vin, 3V3 de salida, 5Vdc de salida y una entrada externa de Reset.
9. **Conector Header-hembra de conexiones analógicas de entrada.** Esta identificadas por las letras desde A0-A5. En el lenguaje Arduino se pueden

usar leyendo el valor digital de la conversión de una magnitud de voltaje para guardarla en una variable de tipo entero. Ejemplo: `int valorADC = analogRead (A0).`

10. Botón de Reset: Permite llevar a un Hard-reset a la tarjeta Arduino. Este botón reinicia el programa cargado en el Arduino. Un Hard-reset es como un apagado y prendido eléctrico de Arduino. Microcontrolador del Arduino.

11. Microcontrolador: Arduino Mega R3 es el ATmega2560.

12. Conector de salida: Es un puerto para entradas/salidas digitales que se encuentran enumeradas desde el 8 hasta el 13. El Pin 13 está conectado también a un LED interno de la tarjeta Arduino. Este Led tiene la leyenda L. Este conector también tiene una entrada para una GND digital y la referencia externa para el ADC, esta referencia no es necesaria en una configuración básica del uso de Arduino. Adicionalmente se encuentra dos pines de comunicación serial I2C llamados SDA y SCL, estos únicamente en las versiones más recientes de Arduino.

13. Header-macho 6x2: Si se requiere re-programar el Bootloader de Arduino - ATmega328p, se usa este puerto de programación. Puede ser conectado a otro Arduino para poder programar el Bootloader. Header-macho 6x2. Si se requiere re-programar el Bootloader de Arduino.

14. ATmega16U2: Se usa este puerto de programación. Puede ser conectado a otro Arduino para poder programar el Bootloader.

✚ **A continuación se detallan todos los pines utilizados y la función desempeñada por cada uno:**

- **Pin 9:** Conexión con el pin Trigger del sensor 1, se configura como salida para el envío del pulso de inicio. ·
- **Pin 8:** Conexión con el pin Echo del sensor 1, se configura como entrada, para la detección de los pulsos de retorno hacia el sensor.
- **Pin 11:** Conexión con el pin Trigger del sensor 2, se configura como salida para el envío del pulso de inicio. ·
- **Pin 10:** Conexión con el pin Echo del sensor 2, se configura como entrada, para la detección de los pulsos de retorno hacia el sensor.
- **Pin 13:** Conexión con el pin Trigger del sensor 3, se configura como salida para el envío del pulso de inicio. ·
- **Pin 12:** Conexión con el pin Echo del sensor 3, se configura como entrada, para la detección de los pulsos de retorno hacia el sensor.
- **Pin 3:** Led indicador de encendido color rojo. ·
- **Pin 4:** Led indicador de lecturas de los sensores ultrasónicos color verde y comunicación serial.

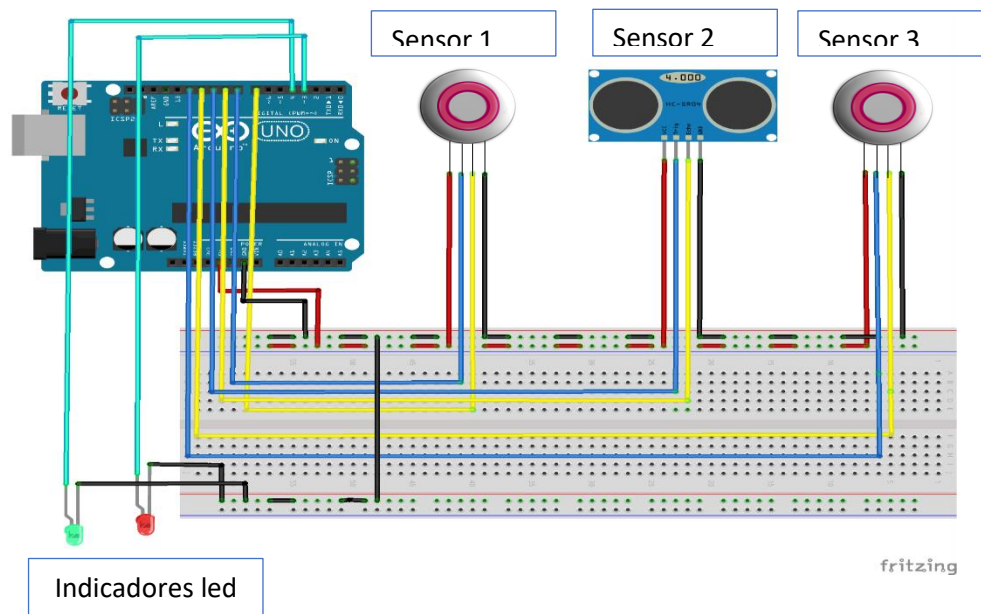


Figura 31. Conexiones de los Sensores Ultrasónicos.

Fuente Propia

Funcionamiento circuito de medición de proximidad

El sensor ultrasónico basa su funcionamiento en la transmisión de una ráfaga de ultrasonido en una frecuencia muy por encima del rango escuchado por el oído humano y la recepción de las ondas de eco (ondas que regresan al rebotar en objetos cercanos), para esto el sensor cuenta con un pin dedicado para el disparo (Trigger) por el cual se envía el pulso inicial y otro pin conocido como echo que envía al microcontrolador un pulso cuyo ancho es proporcional al tiempo que tarda el sonido en viajar desde el emisor al obstáculo y luego de vuelta al módulo receptor.

Para el manejo de los sensores ultrasónicos se tiene en cuenta las consideraciones previstas en el Datasheet del mismo

- 5 V, voltaje de operación.
- Pulso de disparo mínimo 10 μ s. (Pin Trigger), nivel TTL.
- El sensor luego del pulso inicial envía 8 pulsos de 40 khz.
- Duración del pulso de eco de salida: 100 – 25000 μ s.
- Tiempo mínimo de espera entre una medida y el inicio de otra 20 ms.

Considerando que tanto la tarjeta Arduino Uno como los sensores ultrasónicos trabajan bajo el mismo nivel de voltaje, es posible conectar directamente los pines de cada sensor hacia las entradas de la tarjeta sin la necesidad de un divisor de voltaje.

Etapas 4. Microcontrolador Raspberry Pi.

🌈 Raspberry pi 3 B+.

Raspberry Pi 3 Model B+, una iteración interesante del modelo previo que llega con una CPU "acelerada", soporte WiFi de doble banda y también soporte Gigabit Ethernet "capado". Esas mejoras, capadas o no, proporcionan mejores prestaciones con una gran noticia: las Raspberry Pi siguen siendo tan baratas como siempre.

La pequeña placa en la que encontramos los distintos circuitos, chips y conectores tiene una disposición prácticamente idéntica a la de su predecesora, pero hay algunos cambios significativos.

El más llamativo es probablemente la presencia de un disipador situado del SoC encima, algo que permite ayudar a controlar las temperaturas generadas por la CPU ahora que su frecuencia de trabajo es algo mayor.

También encontramos una placa metálica que también protege el chip de conectividad inalámbrica, tanto para la red WiFi como para Bluetooth. Esa protección tiene además un elemento diferencial: el célebre logotipo en forma de frambuesa de las Raspberry Pi está grabado en esa placa. (xataka.com, S,f)

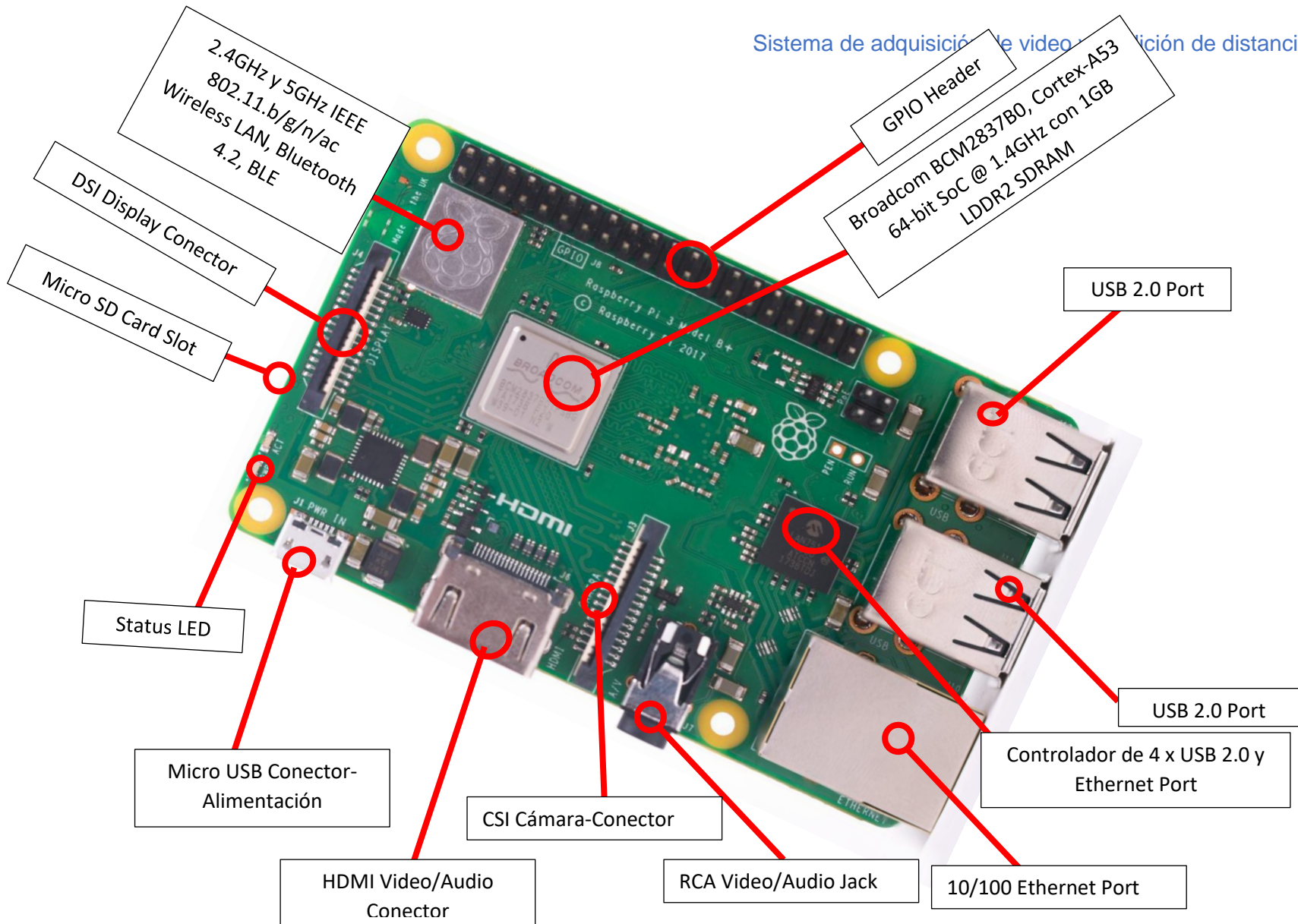


Figura 32. Raspberry Pi 3 Model B+ 2017 Partes que lo conforman

Fuente: Elaboración propia

Todas las especificaciones técnicas de este nuevo modelo de Raspberry Pi 3 son:

- CPU + GPU: **Broadcom BCM2837B0**, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz
- RAM: **1GB** LPDDR2 SDRAM
- Wi-Fi + Bluetooth: 2.4GHz y 5GHz IEEE 802.11.b/g/n/ac, Bluetooth 4.2, BLE
- Ethernet: Gigabit Ethernet sobre USB 2.0 (300 Mbps)
- GPIO de 40 pines
- HDMI
- 4 puertos USB 2.0
- Puerto CSI para conectar una cámara.
- Puerto DSI para conectar una pantalla táctil
- Salida de audio estéreo y vídeo compuesto
- Micro-SD
- Power-over-Ethernet (PoE)

1. Procesador BCM2837B0

Este es el chip Broadcom utilizado en Raspberry Pi 3B + y 3A +. La arquitectura subyacente del BCM2837B0 es idéntica al chip BCM2837A0 utilizado en otras versiones del Pi. El hardware del núcleo ARM es el mismo, solo que la frecuencia tiene una clasificación más alta.

Los núcleos ARM son capaces de funcionar a hasta 1.4GHz, haciendo que el 3B + / 3A + sea aproximadamente un 17% más rápido que el Raspberry Pi 3. El Video Core IV funciona a 400MHz. El núcleo ARM es de 64 bits, mientras que Video Core IV es de 32 bits.

El chip BCM2837B0 está empaquetado de manera ligeramente diferente al BCM2837A0, y más notablemente incluye un difusor de calor para mejores térmicas. Permiten frecuencias de reloj más altas (o funcionan a voltajes más bajos para reducir el consumo de energía) y un monitoreo y control más preciso de la temperatura del chip. (Raspberrypi.org, S,f)

2. GPIO de la Raspberry Pi

Los puertos GPIO que se encuentran integrados en la placa de la Raspberry Pi, son pines digitales creados para trabajar con señales eléctricas con propósitos generales, como se lo haría con los puertos de cualquier microcontrolador, de allí proviene su nombre GPIO (General Purpose Input Output), puertos de entrada / salida de propósito general.

En la imagen se identifica la zona de los GPIO en la placa Raspberry Pi B+, como se observa son un total de 40 pines, los cuales se encuentran distribuidos en dos filas, cabe mencionar que no todos los pines son iguales, la mayoría de estos presentan funciones especiales, como son puertos de comunicación, voltajes de alimentación de 5V y 3.3 V , entre otros.

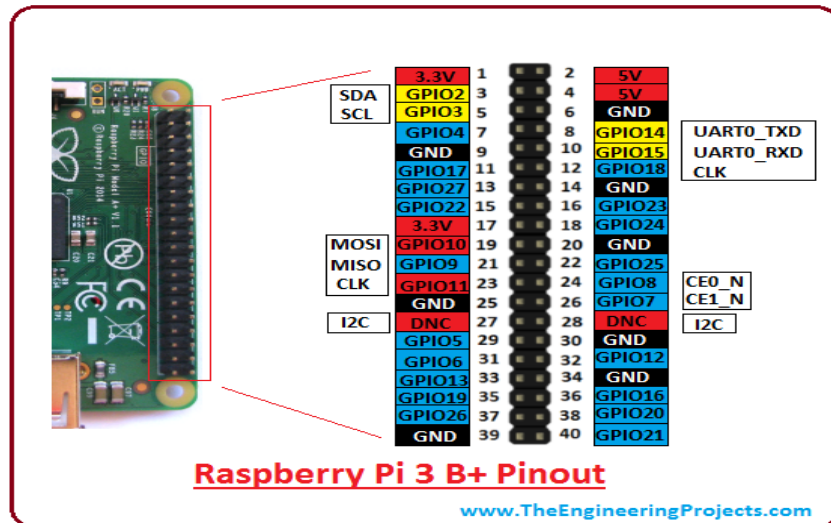


Figura 33. Raspberry Pi 3 B+ Pin Out.

Fuente: (RASPBERRY, s.f.)

3. Conexión inalámbrica.

Para la comunicación inalámbrica se ha seleccionado entre las tecnologías más comunes en los dispositivos móviles tales como: Bluetooth o WI-FI, para esto se analizó las prestaciones otorgadas por cada una. Estas son dos tecnologías inalámbricas utilizan radio frecuencia para el manejo e intercambio de información, cabe mencionar que ambas tecnologías son adaptables al proyecto ya que la mayoría o prácticamente todos los dispositivos que trabajan con sistema operativo Android cuentan con el hardware y el software requerido para su funcionamiento.

Una red WI-FI presenta algunas características superiores a la tecnología Bluetooth, como es la posibilidad de crear una red de dispositivos, lo cual en Bluetooth únicamente es posible tener emparejados dos dispositivos, otra notable ventaja es el rango de alcance, con WI-FI se puede alcanzar distancias superiores a los 10 metros, distancia considera como límite en Bluetooth, estas características se muestra en la Tabla 4.

Tabla 4. Tabla de Tecnología WI-FI y Bluetooth.

Característica	Wi-fi	Bluetooth
Frecuencia	2.4 / 3.6 / 5.0 GHz	2.4 GHz
Ancho de Banda	1 Gbps	24 Mbps
Rango	Hasta 300 metros	Hasta 10 metros
Consumo	Elevado	Reducido
Seguridad	Moderada	Baja

Fuente: Elaboración Propia

Como se observa en la Tabla 3, aunque las dos tecnologías partan del mismo principio, el WI-FI presenta prestaciones superiores. Otro aspecto importante son las librerías existentes para Raspberry Pi 3B+ que utilizan WI-FI, esta y

otras consideraciones referentes al software se detallan con mayor claridad más adelante.

El modelo utilizado para la red WI-FI entre el ordenador Raspberry Pi 3B+ y el dispositivo móvil con sistema operativo Android es el de cliente servidor, para esto fue necesario utilizar el módulo WI-FI de la Raspberry Pi 3B+ para lograr la comunicación inalámbrica con el dispositivo móvil.

Etapa 4.1.Cámara.

Módulo cámara Raspberry

El módulo de la cámara de fotos para Raspberry Pi, con un precio económico, presenta un sensor de imagen OmniVision CMOS de 5 megapíxeles con tecnología de retroiluminación (BSI) de 1.4 micras. Realiza fotografías de 5 megapíxeles, captura de vídeo de alta definición (HD) de 1080p/ (30fps) y velocidad de fotogramas de 720p/(60fps).

El módulo de cámara comunica con Raspberry Pi a través del conector de cámara existente, utilizando CSI para los datos e I2C para el control. Esto permite a los usuarios grabar 30 fotogramas por segundo a 720p y 1080p en formato de video H264. (Diarioelectronicohoy.com, S,f)



Figura 34.Módulo de Cámara Raspberry.

Fuente: (sn, s.f.)

Características

- Compatible con Raspberry PI Zero
- Módulo de la cámara de 5MP OmniVision 5647
- Resolución de imagen fija: 2592x1944
- Vídeo: 1080p @ 30fps, grabación de 720P @ 60FPS y de 640x480p 60/90
- Interfaz serial de la cámara de 15-pin
- Tamaño: 20mm, 25mm, 9mm
- Peso 20 g
- Dimensiones 2.5 x 2 x 1 cm
- Largo cable
- 15 cm

Etapa 4.2. Comunicación.

La plataforma de control Arduino Uno se encarga de controlar los sensores de proximidad, manipular las señales de los mismos y por medio del puerto de comunicación serial enviar el valor de distancia hacia la Raspberry Pi.

El dispositivo principal, el ordenador de placa reducida Raspberry B+, es el que se encarga de comandar las tareas más importantes en el sistema, es el encargado de recibir y procesar los valores provenientes del dispositivo esclavo (Arduino) y de empaquetar los datos por medio de datagramas, utilizando un socket tipo UDP, así como obtener la señal de video del módulo de cámara Raspberry, para con esto generar un servidor de video.

Por último se encuentra un dispositivo Android, el cual se encarga de recibir el valor de distancia y el video proveniente del ordenador Raspberry, para esto se utiliza una aplicación desarrollada en Android Studio, la cual combina diversos

periféricos del dispositivo para mostrar todos los parámetros requeridos por el usuario.

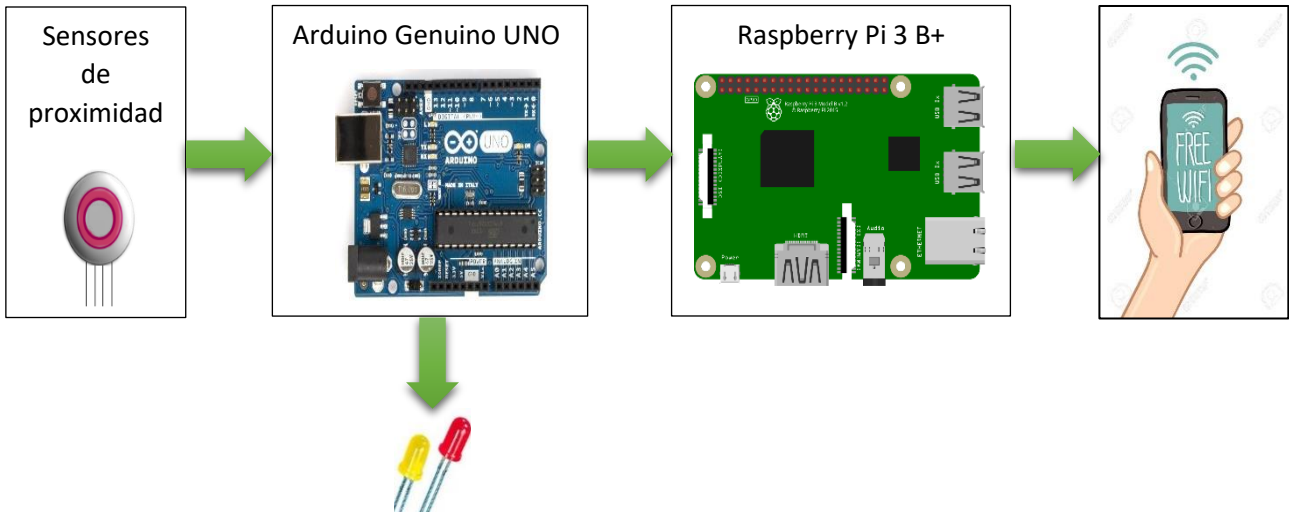


Figura 35. Comunicación entre los dispositivos.

Fuente: Elaboración propia

Esta comunicación se establece por dos tipos; una de ellas es por medio físico donde los componentes que están regidos a este medio son los sensores ultrasónicos, los indicadores led, el microcontrolador arduino uno pro, raspberry pi 3 B+, módulo de cámara raspberry pi.

Por tanto el medio Inalámbrico rige la comunicación de la placa maestra raspberry pi 3 B+ con el dispositivo android.

1.2. Diagrama de bloque de periféricos de entrada y salida del sistema Electrónico.

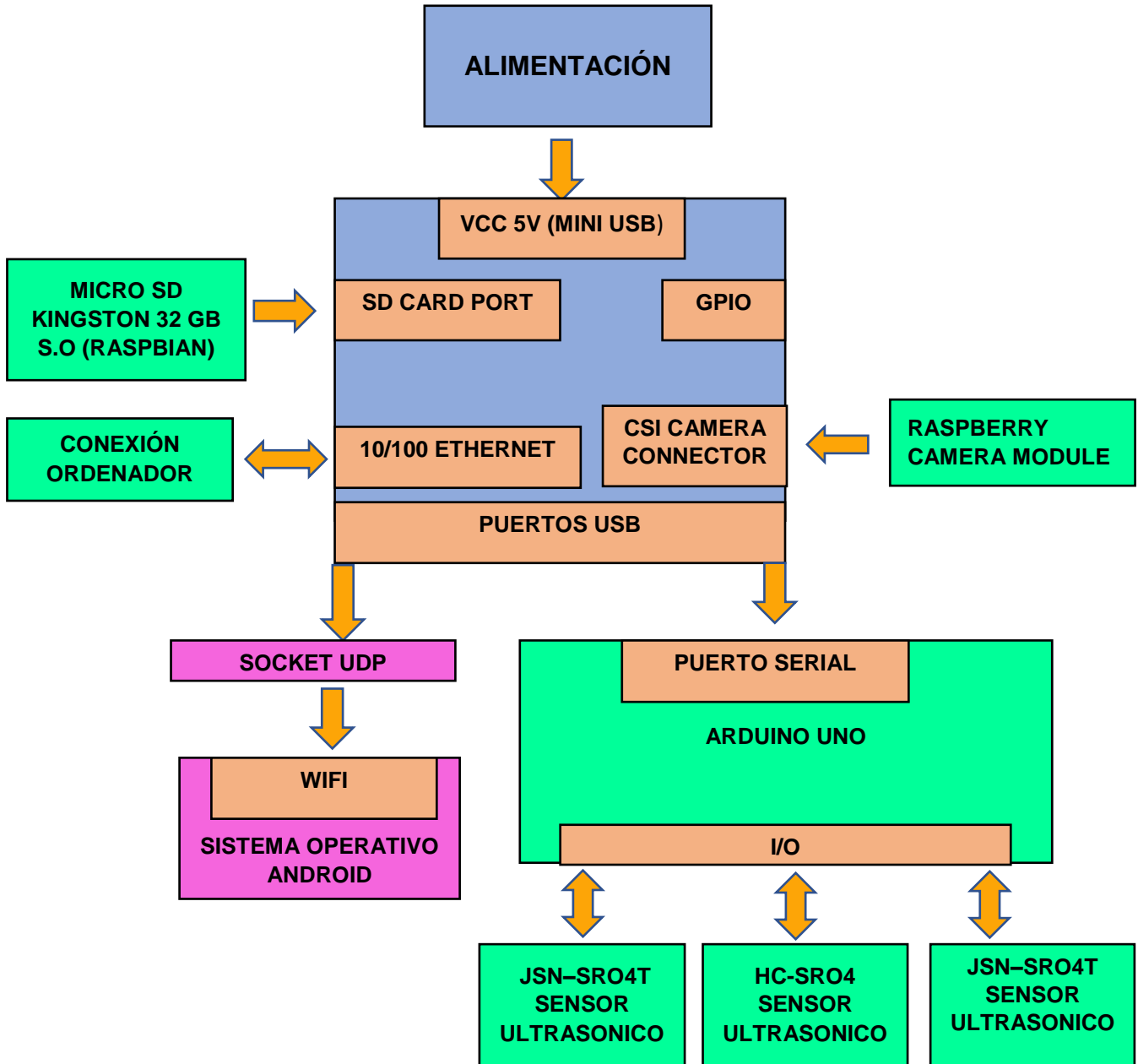


Figura 36. Diagrama de bloque de periféricos de entrada y salida del sistema Electrónico.

Fuente: Elaboración propia

En la Figura 23, el sistema está compuesto por varios dispositivos electrónicos de entrada y salida por lo cual están son:

- Tres sensores de proximidad tienen la función de medir la distancia hacia los obstáculos, por lo tanto son periféricos de entrada/salida.
- El módulo de cámara raspberry pi es un periférico de entrada debido a que solo está obteniendo video en tiempo real.
- El socket es de salida debido a que está transmitiendo a través de un paquete encapsulado los datos de medición de los sensores ultrasónicos
- El arduino solo es de salida ya que está procesando los datos de los sensores ultrasónicos a través de una programación en lenguaje java.
- Tarjeta de almacenamiento de 32 GB es de entrada.
- Conexión con el ordenador o computadora es tanto de entrada a como salida debido que esta placa (raspberry) puede ser manipulada.
- La alimentación es el último componente de tipo entrada este se encarga de comunicar al raspberry pi y el raspberry pi alimenta al resto de los componentes.

X. Software

En este inciso se describe todo lo referente a la programación de los dispositivos requeridos para un sistema de adquisición de video y medición de distancia para el retroceso de un vehículo, para esto se plantean todos los algoritmos requeridos para poder cumplir con todos los requerimientos de software. Es por eso que es necesario comenzar hablando un poco de lenguaje de programación.

1. Lenguajes de programación.

Los lenguajes necesarios para la programación de este sistema son los siguientes:

1.2. Python

Python es un lenguaje de programación interpretado de tipado dinámico cuya filosofía hace hincapié en una sintaxis que favorezca un código legible. Se trata de un lenguaje de programación multiparadigma y disponible en varias plataformas.

Al hacer uso de una sintaxis legible, la curva de aprendizaje es muy rápida, siendo de este modo, uno de los mejores lenguajes para iniciarse en la programación en modo texto. Por ejemplo, si comparamos un código escrito en lenguaje de programación por bloques como Blockly y el mismo código lo escribimos utilizando Python, vemos las similitudes en las instrucciones. (Programo ERGO SUM, s,f)



Figura 37. Logo de Python

Fuente: (PNGOCEAN, s,f)

1.3. Lenguaje C ++

C++ es un lenguaje imperativo orientado a objetos derivado del C. En realidad un superconjunto de C, que nació para añadirle cualidades y características de las que carecía. El resultado es que como su ancestro, sigue muy ligado al hardware subyacente, manteniendo una considerable potencia para programación a bajo nivel, pero se le han añadido elementos que le permiten también un estilo de programación con alto nivel de abstracción. Estrictamente hablando, C no es un subconjunto de C++; de hecho es posible escribir código C que es ilegal en C++. Pero a efectos prácticos, dado el esfuerzo de compatibilidad desplegado en su diseño, puede considerarse que C++ es una extensión del C clásico. La definición "oficial" del lenguaje nos dice que C++ es un lenguaje de propósito general basado en el C, al que se han añadido nuevos tipos de datos, clases, plantillas, mecanismo de excepciones, sistema de espacios de nombres, funciones in-line, sobrecarga de operadores, referencias, operadores para manejo de memoria persistente, y algunas utilidades adicionales de librería (en realidad la librería Estándar C es un subconjunto de la librería C++).

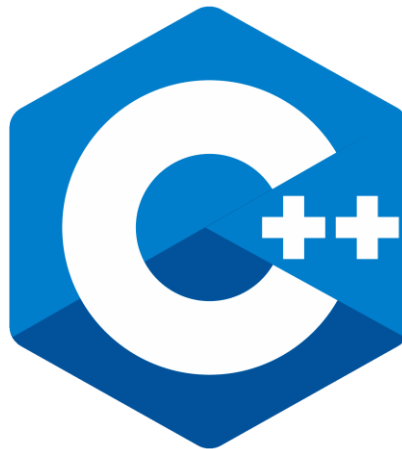


Figura 38.Lenguaje C++ logo.

Fuente: (Adictoalcodigo.blogspot.com, 2017)

1.4. Java

Java es un lenguaje de programación de propósito general, concurrente, orientado a objetos que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible. Su intención es permitir que los desarrolladores de aplicaciones escriban el programa una vez y lo ejecuten en cualquier dispositivo (conocido en inglés como WORA, o "write once, run anywhere"), lo que quiere decir que el código que es ejecutado en una plataforma no tiene que ser recompilado para correr en otra. Java es, a partir de 2012, uno de los lenguajes de programación más populares en uso, particularmente para aplicaciones de cliente-servidor de web, con unos 10 millones de usuarios reportados.

El lenguaje Java se creó con cinco objetivos principales:

1. Debería usar el paradigma de la programación orientada a objetos.
2. Debería permitir la ejecución de un mismo programa en múltiples sistemas operativos.
3. Debería incluir por defecto soporte para trabajo en red.
4. Debería diseñarse para ejecutar código en sistemas remotos de forma segura.
5. Debería ser fácil de usar y tomar lo mejor de otros lenguajes orientados a objetos, como C++.



Figura 39.Lenguaje Java logotipo.

Fuente: (lenguajesdeprogramacion.net, s,f)

Para conseguir la ejecución de código remoto y el soporte de red, los programadores de Java a veces recurren a extensiones como CORBA (Common Object Request Broker Architecture), Internet Communications Engine u OSGi respectivamente.

Estos tres tipos de lenguaje de programación serán usados para crear la lógica dentro del sistema de adquisición de video y medición de distancia en un vehículo a través de un dispositivo móvil donde:

- Python corresponde a la programación de raspberry pi 3 B+.
- C++ corresponde a la programación usada en arduino uno pro.
- Java corresponde a la programación realizada a entorno de IDE de android studio.

2. Tarjeta arduino uno.

Esta tarjeta se encarga de realizar el control de cada uno de los sensores ultrasónicos, además de esta tarea el Arduino maneja la comunicación serial, utilizando su salida USB con el componente central del proyecto la Raspberry Pi, para el envío del valor de proximidad al obstáculo más cercano.

2.1. Diagrama de flujo del programa de arduino.

En la figura 47 se muestra el diagrama de flujo de la programación de la tarjeta Arduino, presentando una idea general de las instrucciones necesarias para el funcionamiento de los sensores ultrasónicos, así como de las subrutinas utilizadas para el envío y recepción de información por medio del puerto serial.

También se detalla el programa principal, partiendo de la inicialización de cada una de las variables a utilizarse en algoritmo propuesto, posteriormente se definen cada uno de los pines físicos que se van a utilizar tanto para los sensores como para los leds indicadores.

Como se muestra en la figura 40, inicialmente se declaran las variables, las cuales son de tipo int, lo que permite obtener mayor precisión valores enteros, también se usó la librería NewPing, siendo esencial a la hora de poner en marcha los 3 sensores ultrasónico obteniendo de manera eficientes las medidas con un margen de error de 1%.

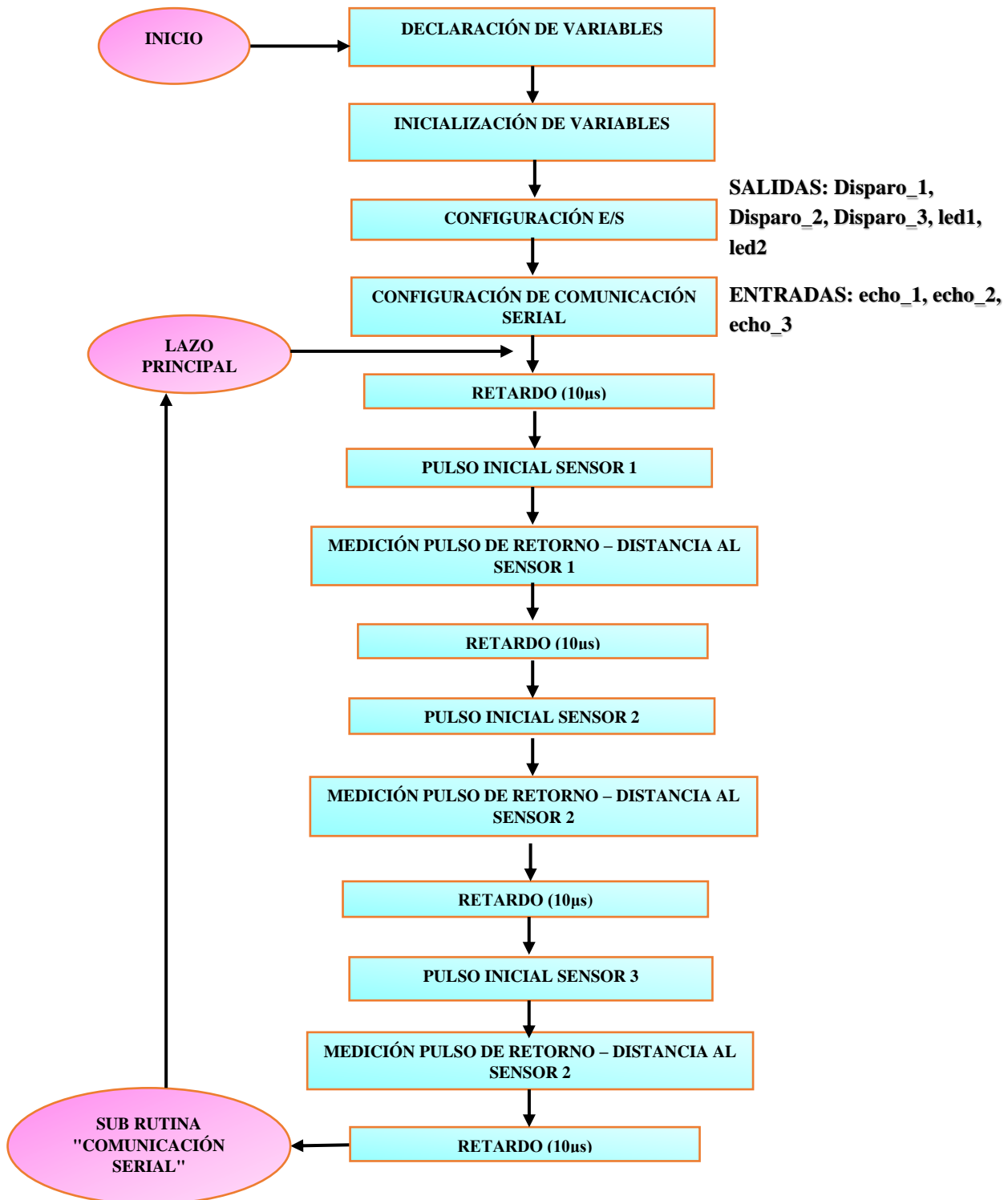


Figura 40. Diagrama de Flujo de la programación de arduino.

Fuente: Elaboración propia

2.2. Comunicación serial con la Raspberry pi 3 B+

Luego de realizadas las mediciones de proximidad por medio de los sensores ultrasónicos, es necesario enviar estos datos hacia la Raspberry Pi. Existen varias formas de comunicar estos dispositivos tales como el uso de los GPIO y los pines de comunicación serial o por medio del protocolo I2C y quizás el método más sencillo es la utilización de un cable USB a micro USB, para esto se utiliza la comunicación serial prevista por la tarjeta Arduino, mientras que en la Raspberry Pi se conecta a cualquiera de los puertos USB dotados en hardware de la placa.

Para realizar esta comunicación es necesario el uso de librerías especiales para el manejo de una tarjeta Arduino, estas son usadas en el sketch para la configuración y manejo del puerto serial. Definiendo a la comunicación serial como una “Interfaz de comunicación de datos digitales que permite establecer transferencia de información entre varios dispositivos”, esta función de la tarjeta le permite interactuar con otros elementos para la generación de sistemas mucho más complejos.

La tarjeta Arduino utiliza el microcontrolador Atmega328p, el cual cuenta con una unidad UART que opera en el nivel TTL 0V – 5V, siendo esta una característica de compatibilidad con la tecnología USB. Los puertos serie están físicamente unidos a los distintos pines de microcontrolador, por lo que si se utiliza la comunicación serial no se puede usar estos pines como entradas y salidas digitales, en el sistema propuesto se emplea los siguientes pines:

- Pin 0 corresponde a Rx.
- Pin 1 corresponde a Tx.

Es por esto que se recomienda su uso siempre y cuando sea indispensable el mismo ya que no solo inhabilita estos pines digitales sino que además las librerías empleadas para el manejo de dicha comunicación ocupa un tamaño considerable al limitado espacio de memoria previsto por el microcontrolador.

Para implementar la comunicación en un primer paso se debe de inicializar el puerto serie por medio del siguiente código:

```
Serial.begin (9600);
```

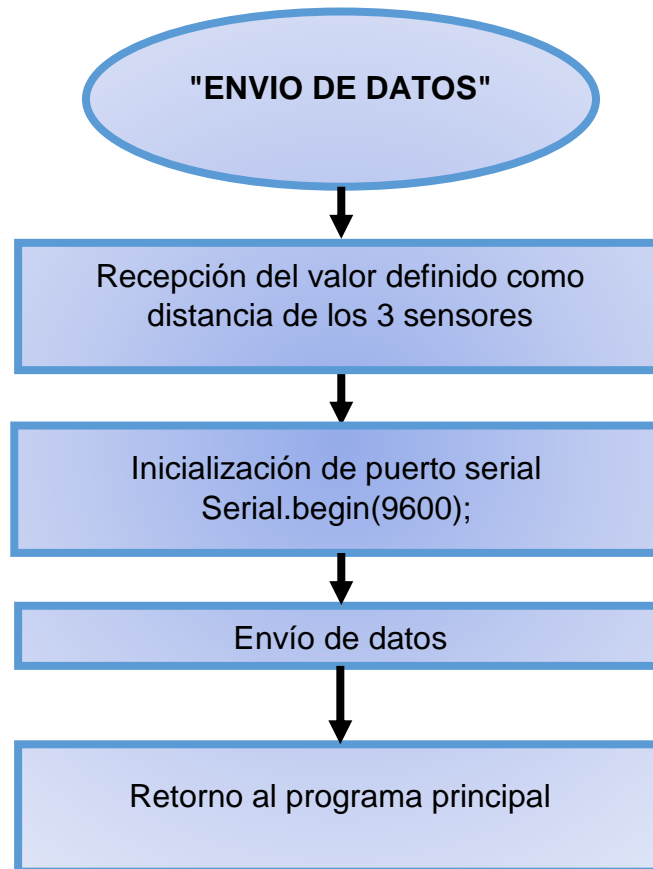


Figura 41. Diagrama de bloque de envío de datos a Raspberry pi 3 B+.

Fuente: Elaboración Propia

De esta forma se configura el puerto serial con una tasa de baudios (baud rate) a 9600, es decir, que se configura el número de unidades a transmitirse por segundo, dicho valor no siempre corresponde a los Bps (bits por segundo) debido a que un baudio puede contener uno o más bits de acuerdo al tipo de codificación.

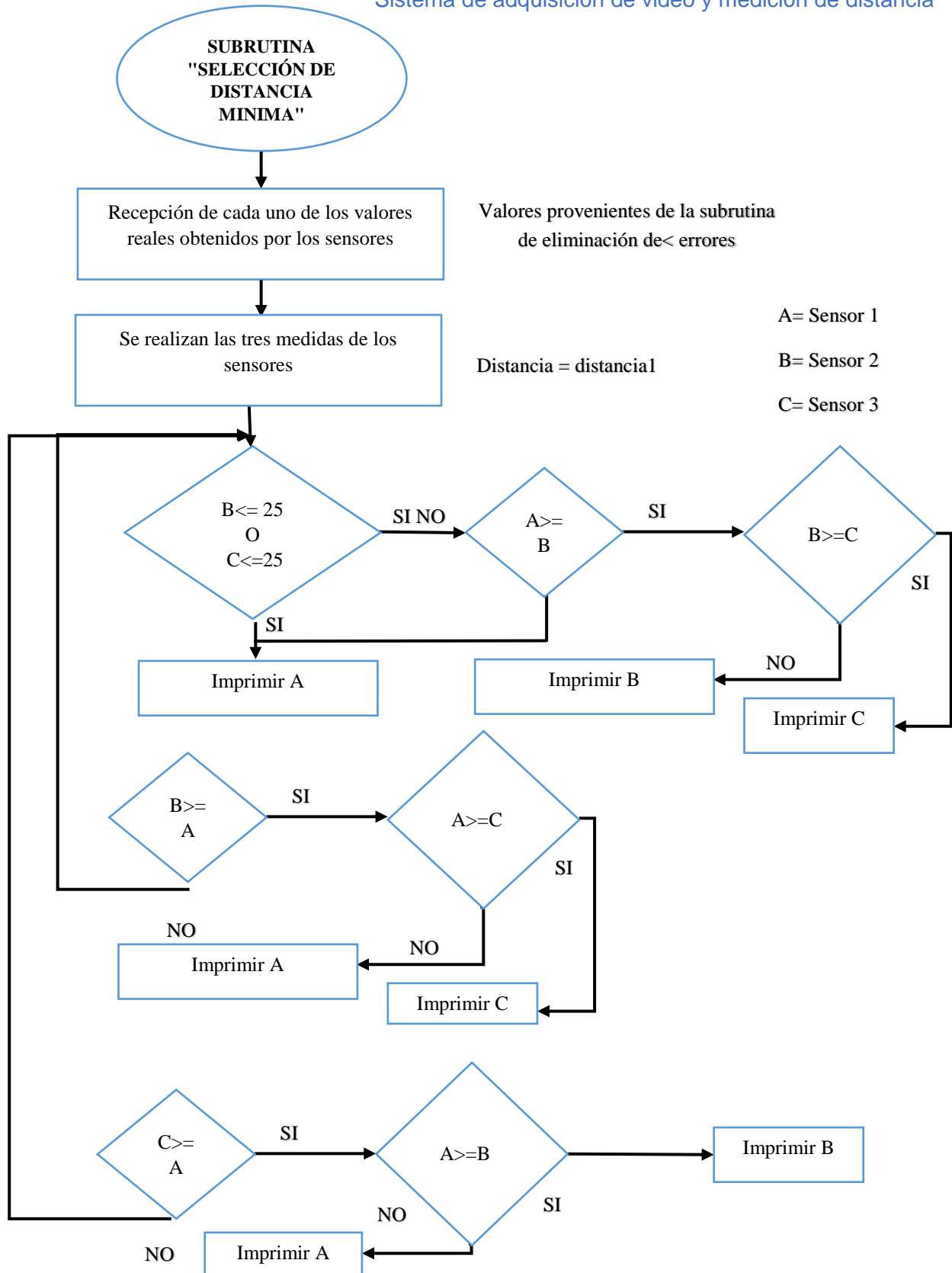


Figura 42. Diagrama de flujo de selección del valor mínimo de los sensores.

Fuente: Elaboración Propia

3. Software implementado en Raspberry pi 3 B+

3.1. Configuración inicial de la Raspberry pi 3 B+

La Raspberry Pi utiliza comúnmente sistemas operativos en el núcleo Linux, siendo Raspbian una versión derivada de Debían el principal sistema operativo optimizado para la utilización del hardware de la Raspberry Pi, pero no es este el único sistema operativo adaptable ya que existen otras distribuciones dedicadas y ligeras como el IPfire, OpenELEC y OSMS, estas tiene su uso especialmente en aplicaciones multimedia.

Para permitir esta flexibilidad la Raspberry Pi no viene con ningún sistema operativo precargado, para esto se requiere de una tarjeta micro SD, la cual sirve de espacio de almacenamiento tanto del sistema operativo como de los diferentes archivos que se creen en el dispositivo, es decir, esta tarjeta actúa como un disco duro en un ordenador convencional.

Para instalar un sistema operativo es necesario preparar inicialmente la tarjeta micro SD, (según modelo) que usamos como memoria. El proceso de instalación se ha simplificado mucho desde el lanzamiento de la Raspberry Pi original, gracias a la aparición de Noobs que en la actualidad es un proceso sencillo y rápido, apto para todos los públicos.

3.2. Pasos para la instalación de NOOBS.

A. Descargar Noobs.

Disponible en la página web: <https://www.raspberrypi.org/downloads/noobs>

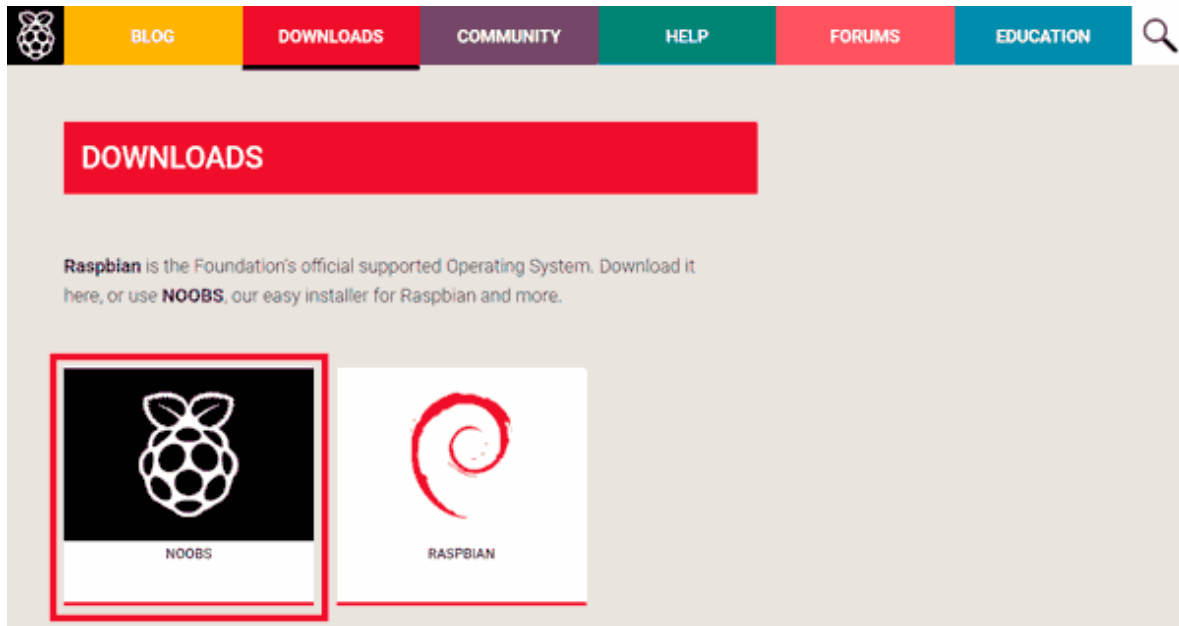


Figura 43. Página oficial para descargar NOOBS.

Fuente: (LUIS LLAMAS, 2019)

Existen dos versiones de Noobs. La "normal" viene precargada con el sistema operativo Raspbian, por lo que, en caso de elegir instalar este sistema, no es necesario que lo descargue Noobs pues se descarga de internet al iniciar Raspberry Pi por primera vez. En caso de elegir otro sistema, este se descargaría antes de instalarse.

B. Formatear en Windows / Mac

Existen varias herramientas específicas para formatear tarjetas SD/micro SD. Una buena opción es emplear SD Card Formatter de la propia asociación SD (SDA), que está disponible en este enlace https://www.sdcard.org/downloads/formatter_4/

En nuestro caso utilizamos Windows y lo descargamos de ahí.

SD Interface Devices

The following interface devices can be used to access SD/SDHC/SDXC memory cards:

- SD Card slot on PC
- USB SD Card reader for USB2.0, USB3.0, USB3.1 & USB-C

Always confirm that the device is compatible with the SD, SDHC or SDXC memory card before formatting.

SD Memory Card Formatter Download for Windows and Mac



Figura 44. *Página de descarga de SD Card Formatter.*

Fuente: (SD Memory Card Formatter, s.f)

Se descargó la aplicación, se eligió la unidad correspondiente a la tarjeta SD/micro SD y se formateó.

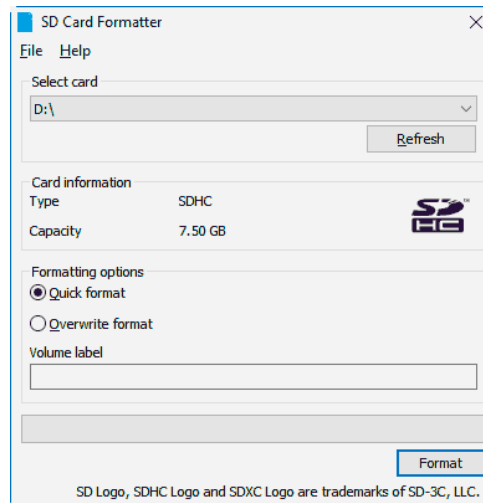


Figura 45. *Página principal del software SD Card Formatter.*

Fuente: (LUIS LLAMAS, 2019)

C. Instalar Noobs.

Para instalar Noobs se necesita un ordenador con un lector de tarjetas SD, para preparar la tarjeta antes de usarla en la Raspberry Pi. Algunos ordenadores (sobre todo portátiles) ya cuentan con un lector integrado y en caso de no poseer éste se debe utilizar un lector de tarjetas USB, se debe proceder antes a formatear la tarjeta SD, y descomprimir el contenido del fichero Zip que hemos descargado en la tarjeta.

La única "complicación" es que, dado las diferencias de formatos de archivo se debe formatear de forma adecuada.

Una vez descomprimido los archivos en la tarjeta SD se procede a colocar la tarjeta microSD a la placa electrónica Raspberry Pi.

Al iniciar aparece un menú con los distintos sistemas operativos disponibles. Elegimos el sistema que queremos y pulsamos la tecla 'I' para instalarlo.

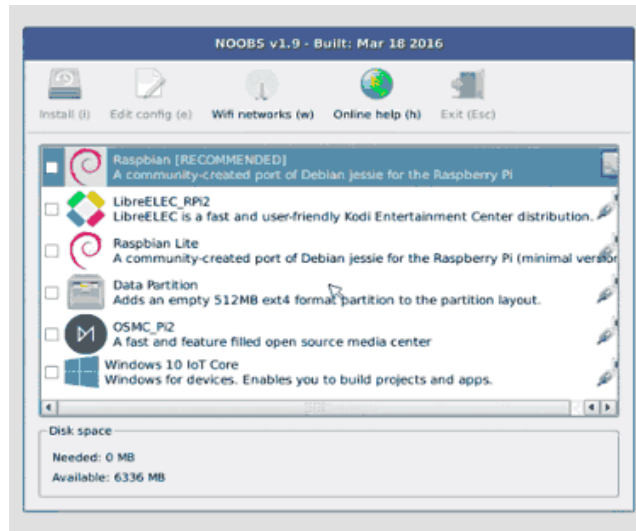


Figura 46. Menú de distintos sistemas operativos disponibles en Raspberry Pi.

Fuente: (LUIS LLAMAS, 2019)

Esperamos pacientemente mientras Noobs instala el sistema operativo seleccionado en la tarjeta de memoria.



Figura 47. Ventana de espera de instalación del sistema operativo seleccionado.

Fuente: (LUIS LLAMAS, 2019)

Una vez finalizado el proceso, hacemos click en el botón para reiniciar Raspberry Pi



Figura 48. Ventana emergente se ha instalado correctamente.

Fuente: (LUIS LLAMAS, 2019)

Raspberry Pi se reinicia y ejecutará el sistema operativo instalado y ¡ya podemos empezar a realizar las configuraciones iniciales!

Poniendo Raspbian en español

Para que muchas de las cosas que aparecen en Raspbian estén en español tenemos que cambiar la localización del sistema operativo. En el menú de inicio vamos a *Preferences* y ahí a *Raspberry Pi Configuration*. En la ventana que se nos abre vamos a la pestaña *Localisation* y en todos los botones que vemos ponemos todo en "spain" o "spanish". De esta forma ponemos la distribución del teclado en español y decimos a Raspbian que cada vez que se actualice se baje los paquetes españoles de los programas.

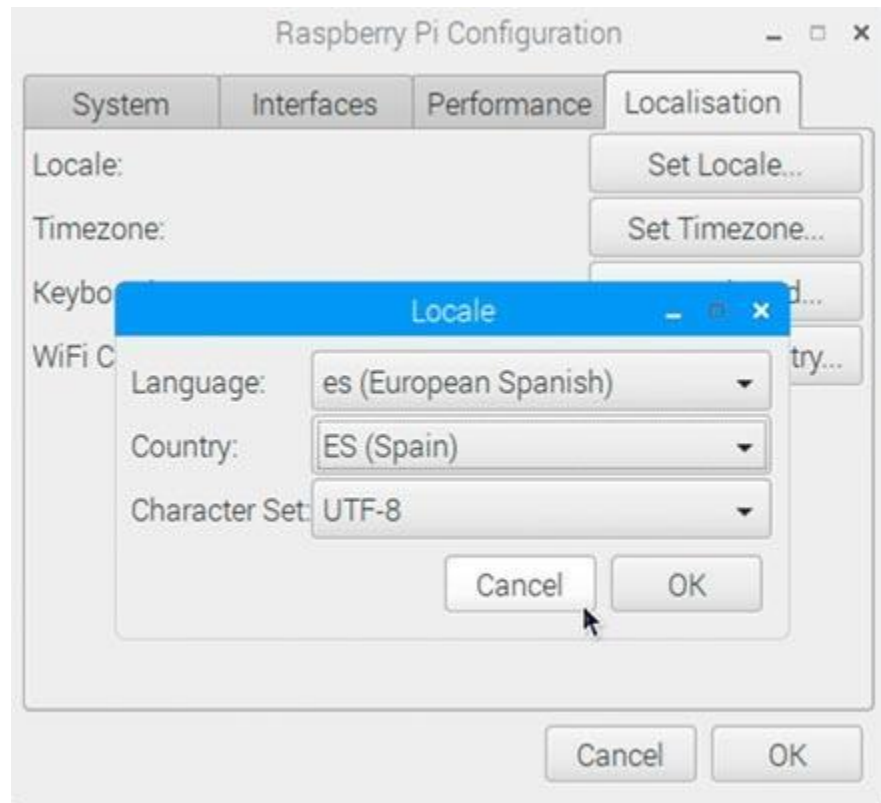


Figura 49. Ventana de configuración de lenguaje.

Fuente: (raspberryparanovatos, 2018)

Configurando el acceso

Una de las cosas más importantes es cambiar la contraseña de acceso de Raspbian. Si dejamos la contraseña por defecto y alguien es capaz de acceder a nuestra Raspberry Pi puede hacer cientos de trastadas con ella. Para esto abrimos el menú de inicio vamos a *Preferences* y ahí a *Raspberry Pi Configuration*, como en el paso anterior. En la pestaña de *Sistema* pulsamos en *Cambiar Clave* y ponemos la clave que queramos. Por defecto esta clave es *raspberrypi*, por lo que es muy recomendable que la cambies si no quieres que accedan a tu ordenador como Pedro por su casa.

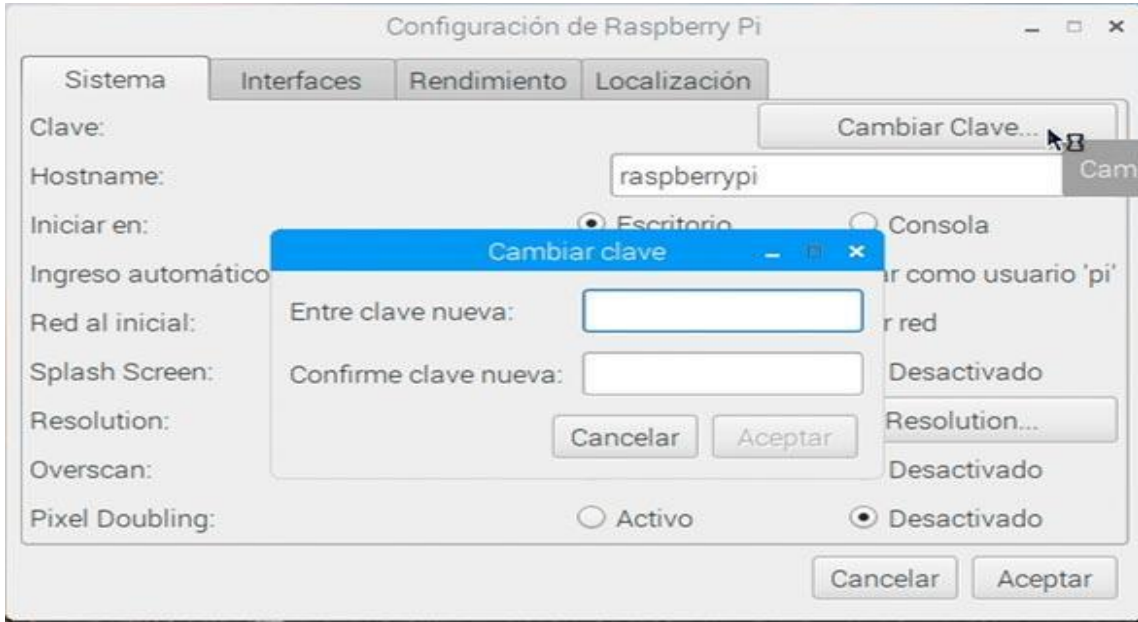


Figura 50. Cambiar clave de usuario de administrador.

Fuente: (raspberryparanovatos, 2018)

También podemos activar la interfaz SSH si queremos acceder a nuestra Raspberry Pi desde cualquier otro ordenador de la red. Para ello desde la ventana de Raspberry Pi Configuration vamos a *Interfacing* y ahí activamos SSH. Para luego acceder desde otro ordenador necesitaremos saber la IP de la Raspberry Pi, la cual pondremos en cualquier programa que permita conectarse a un ordenador por SSH, como puede ser Putty, y acceder con el usuario pi y la contraseña que hemos puesto antes.

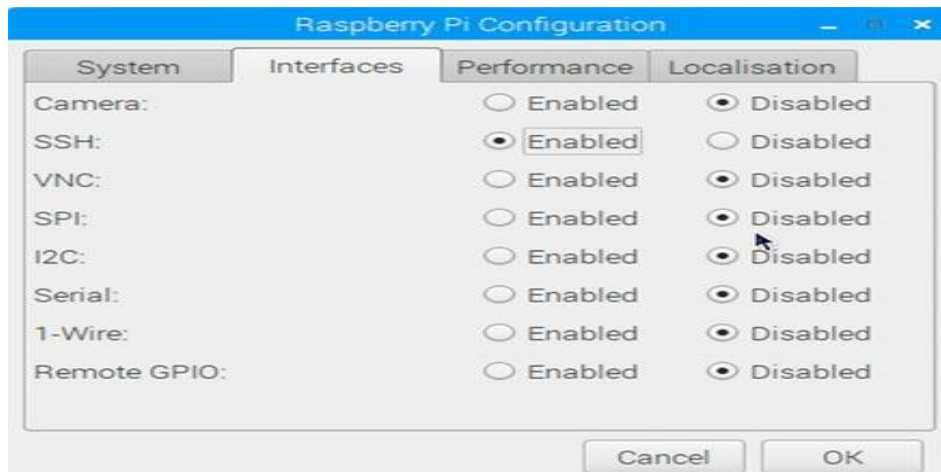


Figura 51. Activar la interfaz SSH.

Fuente: (raspberryparanovatos, 2018)

Conexión de la Raspberry Pi a internet

Para descargar o actualizar cualquier programa o librería es indispensable tener acceso a la red para esto la Raspberry Pi puede conectarse por medio de:

- Conexión directa al Router por medio de un cable Ethernet. .
- Conexión Wireless, a un Router inalámbrico. .
- Por medio de una conexión directa a una Pc.

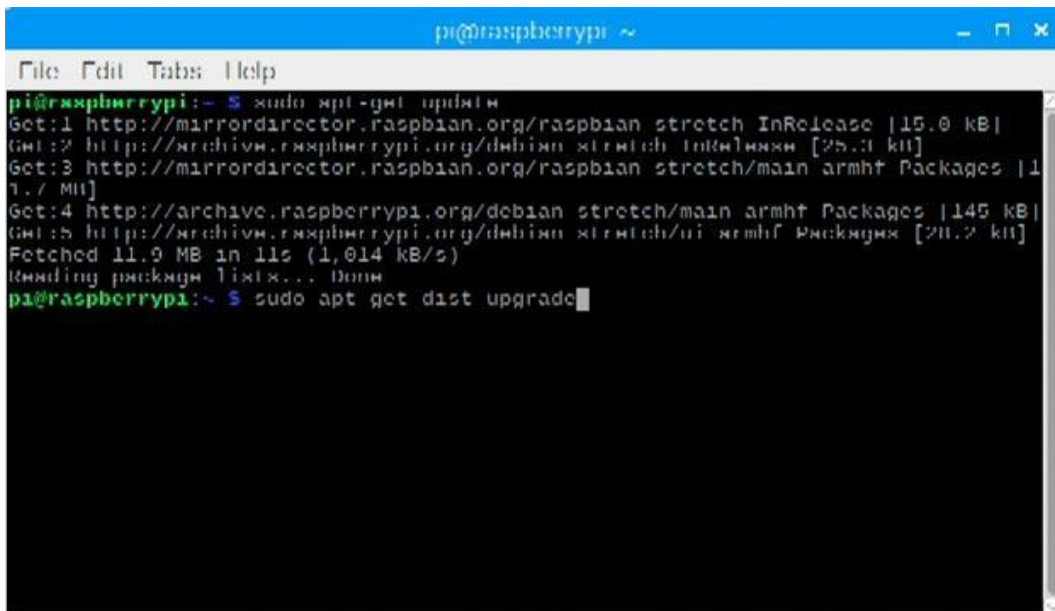
Se optó por la primera opción por medio del cable Ethernet visualizado a una pantalla, teclado y un mouse

Actualizando Raspbian

Para terminar, vamos a actualizar Raspbian. Con esto nos aseguraremos de que tenemos todo listo para funcionar. Para ello tenemos que abrir una terminal y escribir los siguientes comandos:

```
sudo apt get update
```

```
sudo apt get dist-upgrade -y
```



```
pi@raspberrypi ~  
File Edit Tabs Help  
pi@raspberrypi:~$ sudo apt-get update  
Get:1 http://mirrordirector.raspbian.org/raspbian stretch InRelease [15.0 kB]  
Get:2 http://archive.raspberrypi.org/debian stretch InRelease [25.1 kB]  
Get:3 http://mirrordirector.raspbian.org/raspbian stretch/main armhf Packages [1  
7.7 MB]  
Get:4 http://archive.raspberrypi.org/debian stretch/main armhf Packages [145 kB]  
Get:5 http://archive.raspberrypi.org/debian stretch/ui armhf Packages [20.2 kB]  
Fetched 11.9 MB in 11s (1,014 kB/s)  
Reading package lists... Done  
pi@raspberrypi:~$ sudo apt-get dist-upgrade
```

Figura 52. Terminal de Raspbian comandos de actualización.

Fuente: (raspberryparanovatos, 2018)

3.3. Instalación del servidor de video.

La Raspberry Pi no solo se encarga de encapsular y enviar los datos referentes a la medición de proximidad, sino que además actúa como servidor de video, es decir, que capta la señal de video proveniente de la cámara Raspberry y por medio de un direccionamiento IP otorga accesos al dispositivo que actúa en calidad de cliente para poder observar dichas señales de video.

Para esto la Raspberry Pi durante el proceso de la captura de la señal en directo codifica al mismo momento dicha señal, para que después del envío de la misma en el dispositivo móvil se decodifique y se muestre al usuario.

3.3.1. Instalación de la cámara Raspberry.

Luego de conectar la cámara al puerto previsto para misma, se debe de habilitar la cámara para esto se accede al menú rasp-config y se activa la opción de Enable Camera, después se actualiza la lista de repositorios disponibles para la Raspberry Pi (sudo apt-get update).

De acuerdo a la transmisión de video deseada se sigue una metodología diferente, partiendo del uso programas distintos para la transmisión del flujo de video.



Figura 53. Módulo de cámara raspberry.

Fuente: (taringa, 2018)

3.3.2. UV4L

Utilizando este servidor de video se puede transmitir en formato MJPEG y con una latencia mínima. Este formato es soportado por los navegadores web, permitiendo una fácil integración a una aplicación Android por medio de la herramienta Webview, con esto se logra acceder a la señal de video con simplemente estar enlazados a una misma red y conociendo la dirección IP acceder al servidor de video.

El traje de software UV4L consta de una serie de controladores altamente configurables, un módulo opcional de Streaming Server que proporciona una API RESTFULL para desarrollo personalizado y varias extensiones para el servidor que cooperan juntas. Streaming Server también proporciona la interfaz de usuario web básica para que los usuarios finales prueben o usen todas las funcionalidades clave directamente. Para una máxima eficiencia, cada instancia de UV4L se ejecuta como un proceso de sistema único e independiente que explota el hardware subyacente de forma nativa (siempre que sea posible). Aquí hay una lista más detallada de características.

Para la instalación de UV4L en la Raspberry pi se siguen los siguientes pasos.

Nota: Si está ejecutando *Raspbian Wheezy* o *Raspbian Jessie*, abra una terminal y escriba los siguientes comandos:

```
$ curl http://www.linux-projects.org/listing/uv4l_repo/lrkey.asc | sudo apt-key add -
```

❖ En *Raspbian Wheezy*, agregue la siguiente línea al archivo */etc/apt/sources.list* :

```
deb http://www.linux-projects.org/listing/uv4l_repo/raspbian/ wheezy main
```

Mientras en *Raspbian Jessie* agrega esta línea:

```
deb http://www.linux-projects.org/listing/uv4l_repo/raspbian/ Jessie main
```

Si estás ejecutando **Raspbian Stretch** , escribe:

```
$ curl http://www.linux-projects.org/listing/uv4l_repo/lpkey.asc | sudo apt-key  
add -
```

y agregue la siguiente línea al archivo **/etc/apt/sources.list** :

```
deb http://www.linux-projects.org/listing/uv4l_repo/raspbian/stretch Stretch  
main
```

- ❖ Finalmente, estamos listos para actualizar el sistema y buscar e instalar los paquetes:

```
$ sudo apt-get update
```

```
$ sudo apt-get install uv4l uv4l-raspicam
```

Los dos comandos anteriores actualizarán UV4L a la versión más reciente, si ya está instalada.

- ❖ Si desea que el controlador se cargue en el arranque, también instale este paquete opcional:

```
$ sudo apt-get install uv4l-raspicam-extras
```

Para su comodidad, el paquete anterior instalará un script de servicio para iniciar, detener o reiniciar el controlador en cualquier momento, por ejemplo:

```
$ sudo service uv4l_raspicam restart
```

Ahora están instalados el módulo central UV4L y el controlador Video4Linux2 para la placa de cámara CSI. Si ocasionalmente obtiene errores inesperados del controlador, asegúrese de que la cámara esté habilitada y que haya suficiente memoria reservada para la *GPU* (se sugieren 256 MB o más) desde este menú:

```
$ sudo raspi-config
```

También considere actualizar el firmware con el siguiente comando:

```
$ sudo rpi-update
```

Si no ha instalado el *paquete* opcional *uv4l-raspicam-extras* (que proporciona una secuencia de comandos conveniente para iniciar uv4l con la configuración tomada de un archivo de configuración) y desea probar rápidamente uv4l con el módulo de la cámara, primero cargue el controlador manualmente:

```
$ uv4l --driver raspicam --auto-video_nr --width 640 --height 480 --encoding jpeg
```

Es posible que también deba configurar la *opción* *-tc358743-i2c-dev* correctamente en caso de que tenga el TC358743 en su lugar). Ahora puede tomar una instantánea JPEG del dispositivo:

```
$ dd if = / dev / video0 of = snapshot.jpeg bs = 11M count = 1
```

Además del controlador para la placa de cámara Raspberry Pi, se pueden instalar opcionalmente los siguientes front-end y controladores de Streaming Server:

```
$ sudo apt-get install uv4l-server uv4l-uvc uv4l-xscreen uv4l-mjpegstream uv4l-dummy uv4l-raspidisp
```

De lo contrario, si tiene algún otro modelo (por ejemplo, Raspberry Pi 2 o 3), escriba:

```
$ sudo apt-get install uv4l-webrtc
```

Como Streaming Server puede servir y ejecutar cualquier aplicación web personalizada, está disponible un paquete opcional que contiene el código fuente de algunas de las demostraciones mencionadas en los ejemplos (por ejemplo, este). Los archivos se instalarán en **/usr/share/uv4l/demos/**:

```
$ sudo apt-get install uv4l-demos
```

3.4. Configuración como Access Point

La Raspberry Pi se puede usar como un punto de acceso inalámbrico, ejecutando una red independiente. Esto se puede hacer usando las funciones inalámbricas incorporadas de Raspberry Pi 3 o Raspberry Pi Zero W, o usando un dongle inalámbrico USB adecuado que admita puntos de acceso.

Tenga en cuenta que esta documentación se probó en un Raspberry Pi 3, y es posible que algunos dongles USB necesiten ligeros cambios en su configuración.

Pero en nuestro caso trabajamos con la tarjeta de wifi de la raspberry ya que no se pudo conseguir dongle usb

Para funcionar como un punto de acceso, la Raspberry Pi necesitará tener instalado un software de punto de acceso, junto con el software del servidor DHCP para proporcionar dispositivos de conexión con una dirección de red.

Para crear un punto de acceso, necesitaremos DNSMasq y HostAPD.

Ejecutamos el comando:

ifconfig

El comando anterior mostrara todas las conexiones. Una de ellas será nuestra WiFi, normalmente será *wlan0*. Asegúrate de que sea esa o ajusta el código siguiente y cambia *wlan0* por tu tarjeta.

```

TX packets 0 bytes 0 (0.0 B)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
inet 127.0.0.1 netmask 255.0.0.0
inet6 ::1 prefixlen 128 scopeid 0x10<host>
loop txqueuelen 1000 (Local Loopback)
RX packets 204 bytes 17792 (17.3 KiB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 204 bytes 17792 (17.3 KiB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 192.168.42.1 netmask 255.255.255.0 broadcast 192.168.42.255
inet6 fe80::2467:a83f:f348:c597 prefixlen 64 scopeid 0x20<link>
ether b8:27:eb:c7:2c:19 txqueuelen 1000 (Ethernet)
RX packets 60 bytes 4460 (4.3 KiB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 179 bytes 15396 (15.0 KiB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@raspberrypi:/home/pi#

```

Figura 54. Comando ifconfig.

Fuente: Elaboración propia.

Ejecutamos las siguientes líneas de comando:

1. *sudo apt-get update*
2. *sudo apt-get upgrade*

Se procede a instalar el software que actuara como punto de acceso(Hostap) y sus respectivos iptable manager, se ejecutan las siguientes líneas de comando:

1. *sudo apt-get install hostapd isc-dhcp-server*
2. *sudo apt-get install iptables-persistent*
3. *sudo apt install dnsmasq hostapd* (este comando instalara todo el software requerido).

```
Package configuration
|-----| Configuring iptables-persistent |-----|
Current iptables rules can be saved to the configuration file
/etc/iptables/rules.v4. These rules will then be loaded automatically
during system startup.

Rules are only saved automatically during package installation. See the
manual page of iptables-save(8) for instructions on keeping the rules
file up-to-date.

Save current IPv4 rules?
  <Yes>                                <No>
```

Figura 55. Instalación de las reglas de IPV4.

Fuente: Elaboración propia.

Como los archivos de configuración aun no estan listo, apague el software con los siguientes comandos:

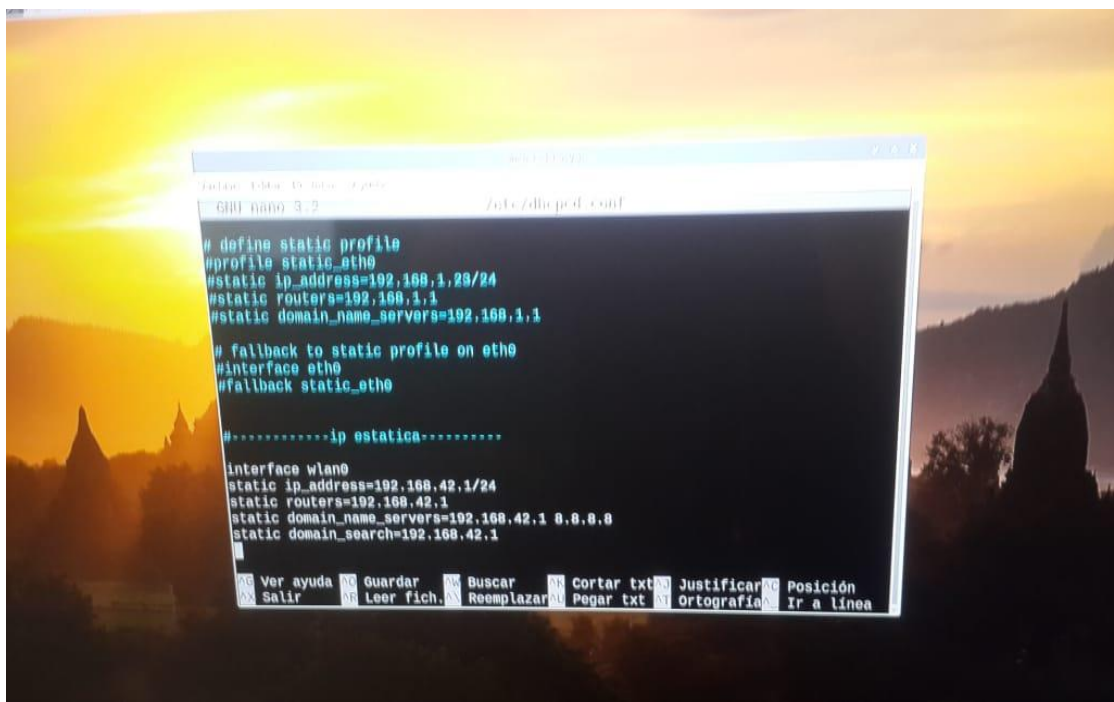
1. *sudo systemctl stop dnsmasq*
2. *sudo systemctl stop hostapd*

3.4.1. Configurar una IP Estática

Para que una red independiente actúe como un acceso point, se le debe asignar una dirección IP estática a la Raspberry Pi en el puerto inalámbrico wlan0, por lo cual se le asigna la dirección 192.168.42.1 con un rango de conexiones de 192.168.42.10 a la 192.168.42.50 obteniendo 40 ip disponibles a la hora de conectarse a la zona wifi.

Para configurar la dirección IP estática, edite el archivo de configuración dhcpd con:

```
sudo nano /etc/dhcpd.conf
```



```
GNU nano 3.2 /etc/dhcpd.conf
# define static profile
#profile static_eth0
#static ip_address=192.168.1.29/24
#static routers=192.168.1.1
#static domain_name_servers=192.168.1.1

# fallback to static profile on eth0
#interface eth0
#fallback static_eth0

#-----ip estatica-----
interface wlan0
static ip_address=192.168.42.1/24
static routers=192.168.42.1
static domain_name_servers=192.168.42.1 8.8.8.8
static domain_search=192.168.42.1

Ver ayuda Guardar Buscar Cortar txt Justificar Posición
Salir Leer fich. Reemplazar Pegar txt Ortografía Ir a línea
```

Figura 56. Archivo de modificación dhcpd.conf

Fuente: Elaboración propia.

Ahora se reinicia el demonio dhcpd con el siguiente comando:

1. `sudo service dhcpd restart`

3.4.2. Configuración del servidor DHCP (dnsmasq)

El servicio DHCP es proporcionado por dnsmasq. De manera predeterminada, el archivo de configuración contiene mucha información que no es necesaria y es más fácil comenzar desde cero. Cambie el nombre de este archivo de configuración y edite uno nuevo con los siguientes comandos:

1. `sudo mv /etc/dnsmasq.conf /etc/dnsmasq.conf.orig`
2. `sudo nano /etc/dnsmasq.conf`

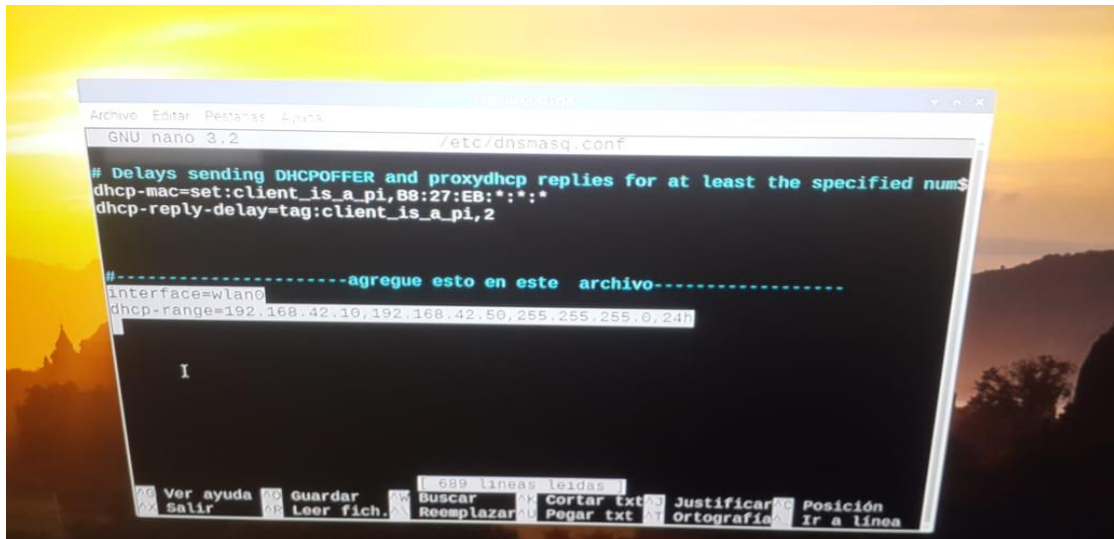


Figura 57. Archivo de modificación dnsmasq.conf

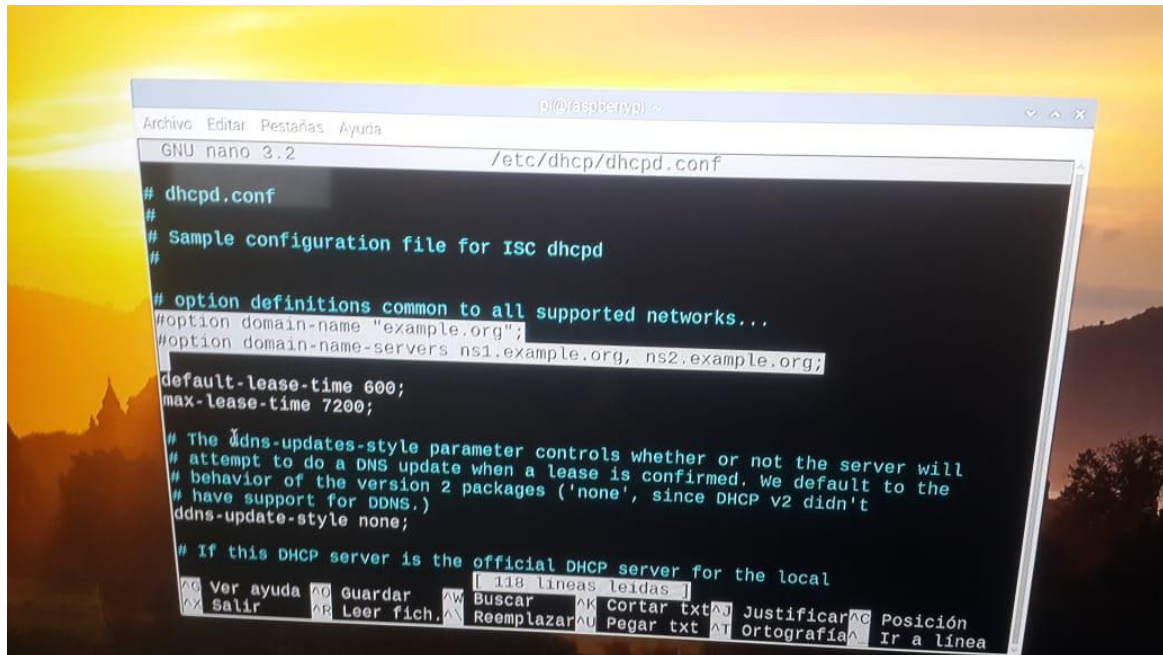
Fuente: Elaboración propia.

Luego se direcciona a la ruta `/etc/dhcp/dhcpd.conf` y se ejecuta el comando:

1. `Sudo nano dhcpd.conf`

Se busca las siguientes líneas y se comentan con un `#` quedando de la siguiente forma:

1. `#option domain-name "example.org";`
2. `#option domain-name-servers ns1.example.org, ns2.example.org;`



```
Archivo Editar Pestañas Ayuda
p@raspberrypi ~
GNU nano 3.2 /etc/dhcp/dhcpd.conf
# dhcpd.conf
#
# Sample configuration file for ISC dhcpd
#
# option definitions common to all supported networks...
#option domain-name "example.org";
#option domain-name-servers ns1.example.org, ns2.example.org;
default-lease-time 600;
max-lease-time 7200;
# The ddns-update-style parameter controls whether or not the server will
# attempt to do a DNS update when a lease is confirmed. We default to the
# behavior of the version 2 packages ('none', since DHCP v2 didn't
# have support for DDNS.)
ddns-update-style none;
# If this DHCP server is the official DHCP server for the local
# network, the authoritative directive should be uncommented.
authoritative;
#authoritative;
```

Figura 58. Archivo de configuración dhcpd.conf

Fuente: Elaboración propia.

Ahora busca ahí mismo en el mismo documento.

1. *# If this DHCP server is the official DHCP server for the local*
2. *# network, the authoritative directive should be uncommented.*
3. *#authoritative;*

Se quita el símbolo # de la línea *authoritative;* para que SI se ejecute, quedando:

1. *If this DHCP server is the official DHCP server for the local*
2. *network, the authoritative directive should be uncommented. authoritative;*

Entonces wlan0, vamos a proporcionar direcciones IP entre

Reload dnsmasq para utilizar la configuración actualizada:

1. *sudo systemctl reload dnsmasq*

3.4.3. Configuración del software host del punto de acceso (hostapd)

Se debe editar el archivo de configuración de hostapd, ubicado en /etc/hostapd/hostapd.conf, para agregar los diversos parámetros para su red inalámbrica. Después de la instalación inicial, este será un archivo nuevo / vacío por lo cual se ejecuta el siguiente comando:

1. `sudo nano /etc/hostapd/hostapd.conf`

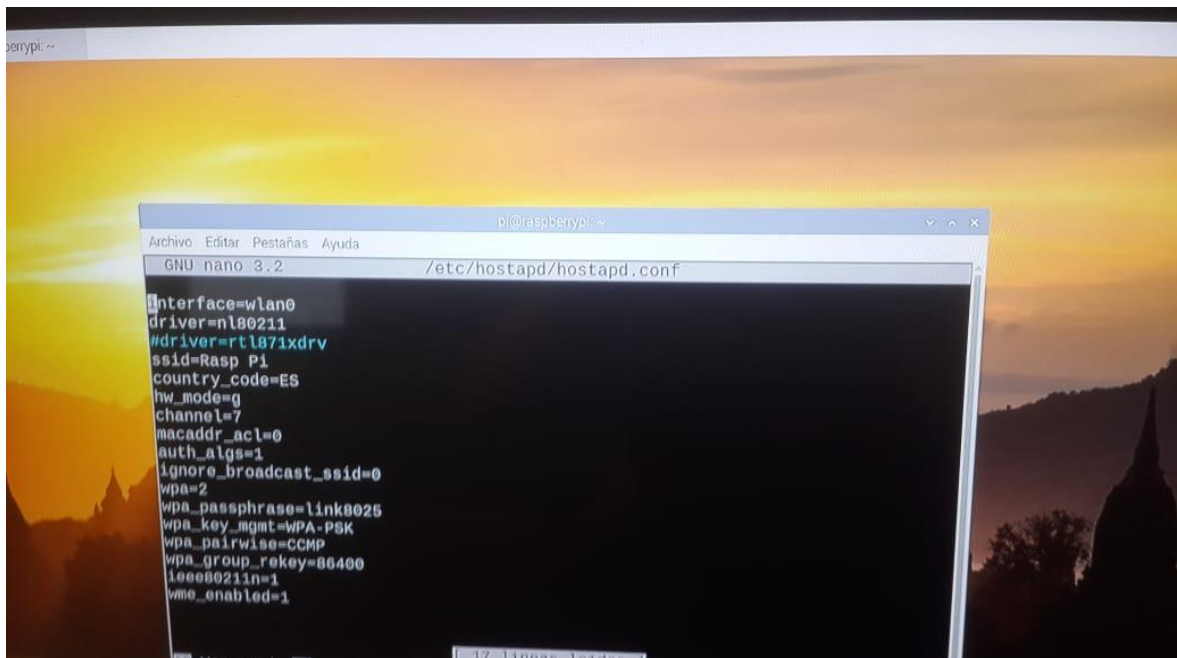
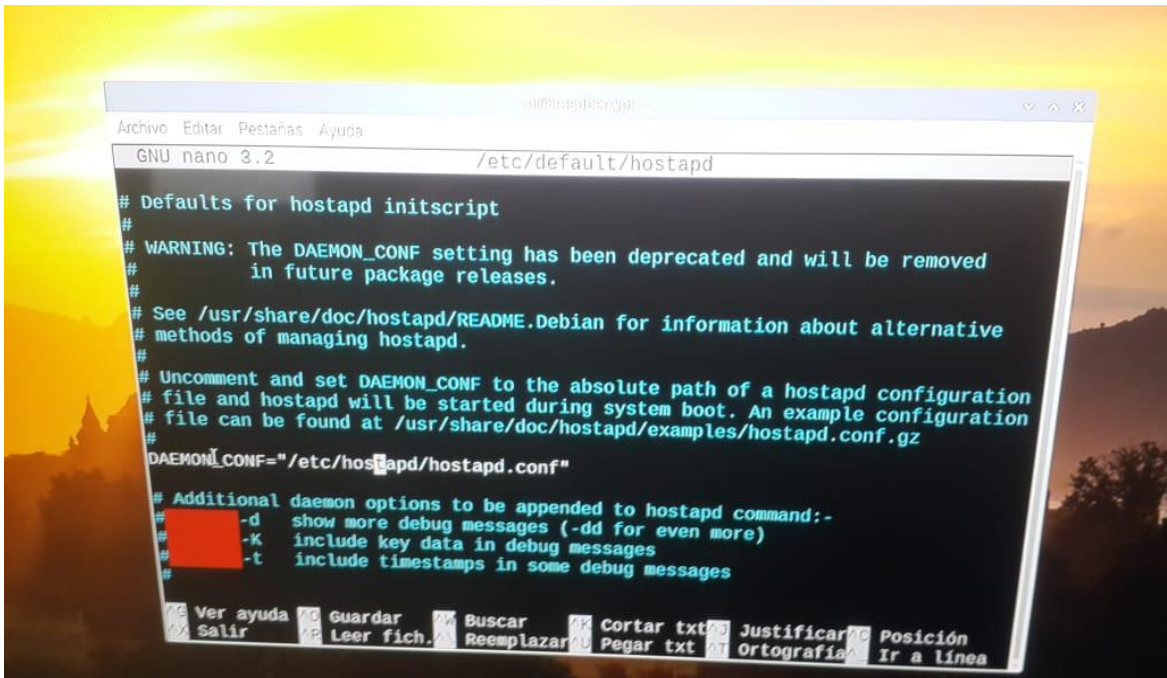


Figura 59.Parámetros de configuración para el hostapd.

Fuente: Elaboración propia.

Ahora se necesita decir al sistema donde se encuentra el archivo de configuración por lo cual nos dirigimos a la ruta y haciendo uso del siguiente comando:

1. `sudo nano /etc/default/hostapd`



```
Archivo Editar Pestañas Ayuda
GNU nano 3.2 /etc/default/hostapd
# Defaults for hostapd initscript
#
# WARNING: The DAEMON_CONF setting has been deprecated and will be removed
# in future package releases.
#
# See /usr/share/doc/hostapd/README.Debian for information about alternative
# methods of managing hostapd.
#
# Uncomment and set DAEMON_CONF to the absolute path of a hostapd configuration
# file and hostapd will be started during system boot. An example configuration
# file can be found at /usr/share/doc/hostapd/examples/hostapd.conf.gz
#
DAEMON_CONF="/etc/hostapd/hostapd.conf"
#
# Additional daemon options to be appended to hostapd command:-
# -d show more debug messages (-dd for even more)
# -K include key data in debug messages
# -t include timestamps in some debug messages
#
Ver ayuda Guardar Buscar Cortar txt Justificar Posición
Salir Leer fich. Reemplazar Pegar txt Ortografía Ir a línea
```

Figura 60. Archivo de configuración hostapd el default.

Fuente: Elaboración propia.

Se busca la línea con #DAEMON_CONF y se reemplaza con esto:

```
DAEMON_CONF="/etc/hostapd/hostapd.conf"
```

Para ponerlo en marcha se ejecutan los siguientes comandos:

1. `sudo systemctl unmask hostapd`
2. `sudo systemctl enable hostapd`
3. `sudo systemctl start hostapd`

Los comandos anteriores habilitaron el hostapd.

Ahora se realiza una comprobación rápida de su estado para asegurarse de que estén activos y en ejecución por lo cual se teclea:

1. `sudo systemctl status hostapd`
2. `sudo systemctl status dnsmasq`

3.4.4. Agregar enrutamiento y mascarada

Se edita en la ruta /etc/sysctl.conf y se descomenta la siguiente línea:

```
net.ipv4.ip_forward=1
```

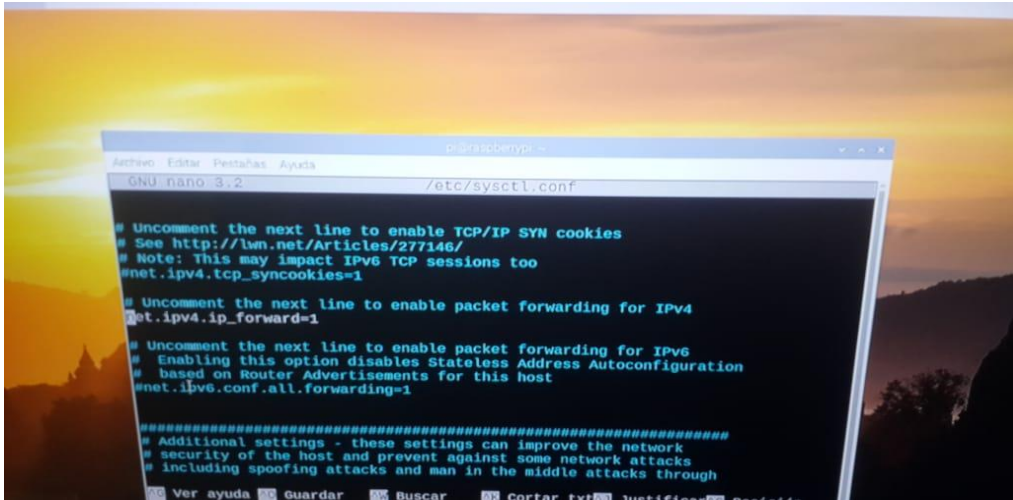


Figura 61. Archivo de enrutamiento y mascarada.

Fuente: Elaboración propia.

Se agrega una mascarada para el tráfico saliente en eth0:

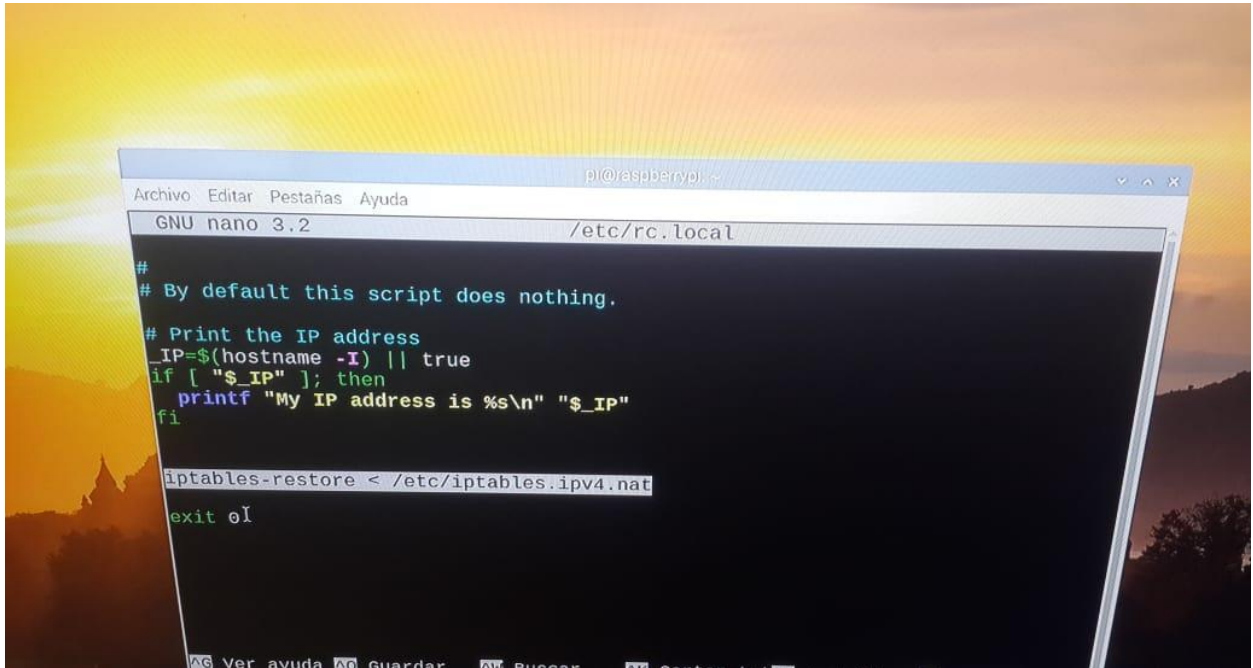
1. `sudo iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE`

Se guarda la regla de iptables.

1. `sudo sh -c "iptables-save > /etc/iptables.ipv4.nat"`

Se edita en la ruta /etc/rc.local y se le agrega esto justo arriba de "exit 0" para instalar estas reglas en el arranque.

1. `iptables-restore < /etc/iptables.ipv4.nat`



```
pi@raspberrypi ~  
Archivo Editar Pestañas Ayuda  
GNU nano 3.2 /etc/rc.local  
#  
# By default this script does nothing.  
# Print the IP address  
_IP=$(hostname -I) || true  
if [ "$_IP" ]; then  
  printf "My IP address is %s\n" "$_IP"  
fi  
  
iptables-restore < /etc/iptables.ipv4.nat  
  
exit 0
```

Figura 62. Archivo de configuración para que se ejecute el `hostapd` como `Daemon`.

Fuente: Elaboración propia.

Luego se reinicia el ordenador, en caso de que no levante usar el siguiente comando:

```
sudo /usr/sbin/hostapd /etc/hostapd/hostapd.conf
```

3.5. Programa principal.

Se crea un fichero en un directorio específico, en dicho fichero utilizando el lenguaje de programación de Python se desarrolla el programa encargado de recibir el valor de distancia proveniente de los sensores de proximidad por medio de la tarjeta Arduino Uno, así como realizar la configuración para que la Raspberry Pi actúe como servidor en una comunicación por medio de sockets.

Se utilizan sockets para intercambiar flujos de datos entre los dispositivos cliente – servidor, en donde la primera función realiza el dispositivo móvil y la Raspberry Pi realiza las tareas de servidor. Al usar el socket se intercambia la información de una manera fiable y ordenada constituyendo un mecanismo para entrega de paquetes de datos provenientes de tarjetas de red distintas utilizando un protocolo de red.

3.5.1. Socket UDP.

UDP son las siglas de Protocolo de Datagrama de Usuario (en inglés User Datagram Protocol) un protocolo sin conexión que, como TCP, funciona en redes IP.

UDP/IP proporciona muy pocos servicios de recuperación de errores, ofreciendo en su lugar una manera directa de enviar y recibir datagramas a través una red IP. Se utiliza sobre todo cuando la velocidad es un factor importante en la transmisión de la información. (masadelante.com, s,f)

Por lo tanto, en este proyecto se usara un protocolo UDP, donde se a través de socket se estará estableciendo la comunicación entre el cliente, servidor porque a diferencia de los sockets TCP los UDP no está orientado a establecer una conexión 100% segura pero con una latencia mucha menor, es decir, que la velocidad de transmisión es superior ya que este el UDP es un protocolo a nivel de la capa de transporte basado en el intercambio de datagramas, el cual permite el envío de datagramas a través de la red sin que se haya establecido previamente una conexión.

El datagrama contiene toda la información necesaria de direccionamiento en su cabecera, esto permite el envío y recepción de los mismos sin una conexión permanente.

Se utiliza los sockets UDP porque estos no restringen a un modelo basado en la conexión, lo que permite velocidades altas de transmisión reduciendo la sobrecarga al sistema operativo, además de que la información no se fragmenta lo cual facilita las tareas del receptor. Considerando que la latencia del sistema propuesto no puede ser superior a los 100 ms, este protocolo se adapta al sistema adecuadamente.

Los datagramas generados en Python trasladan la información referente a la medición de proximidad, para esto en la cabecera del datagrama se incorpora la dirección IP destino y un número de puerto de comunicación de la siguiente manera:

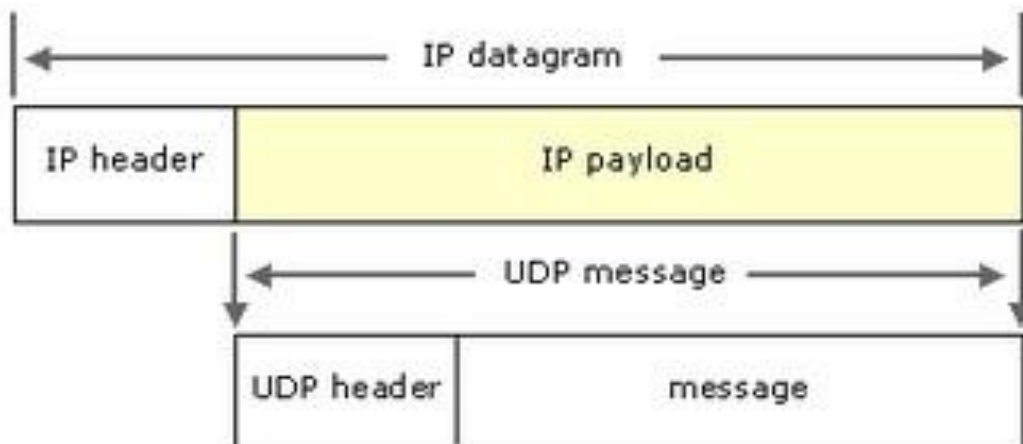


Figura 63. Encapsulación de datagramas UDP.

Fuente: (Intercomunicación y Seguridad en Redes, 2012)

Dónde:

Cabecera: Dirección IP destino > 192.168.42.1

Puerto: 6000

Mensaje: valor de medición de proximidad

3.5.2. Comunicación con el dispositivo móvil.

El socket tiene un periodo de vida diferente en el lado de la Raspberry Pi como al lado del dispositivo móvil, mientras en el primero se están enviando constantemente datagramas, el segundo cada vez que recibe un datagrama envía nuevamente un datagrama hacia la Raspberry Pi de la siguiente manera:

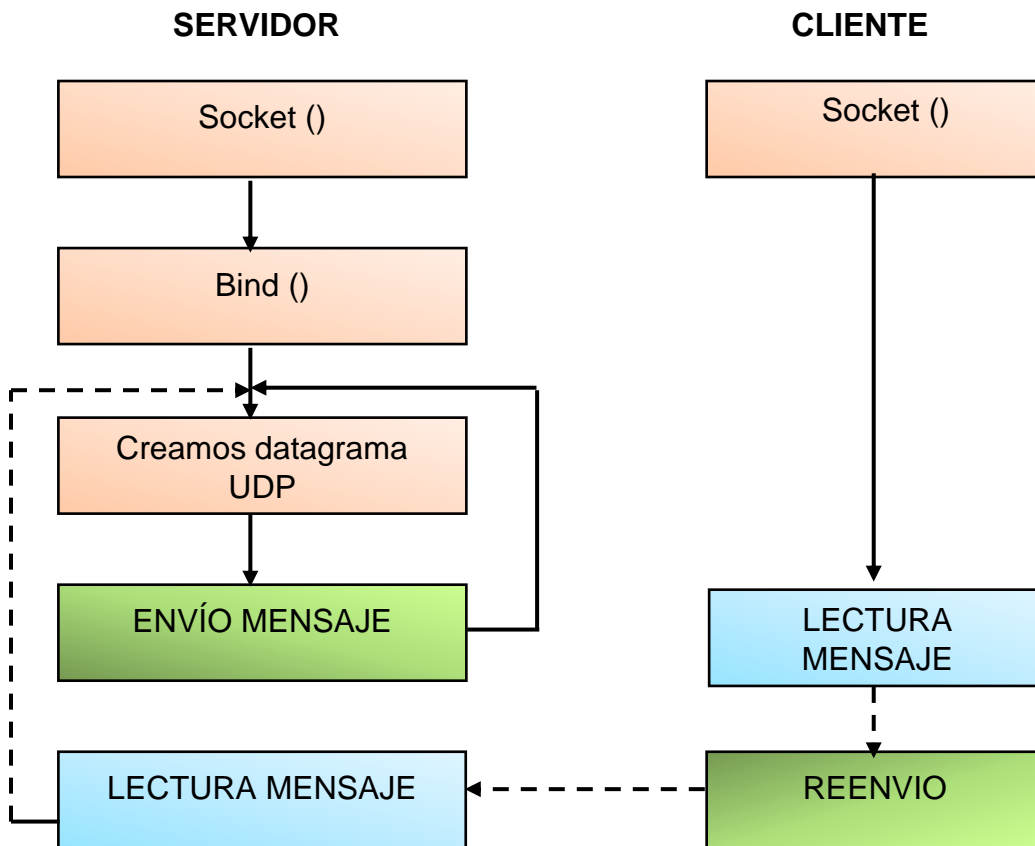


Figura 64. Flujo de datos cliente-servidor.

Fuente: Elaboración propia

4. Desarrollo de la aplicación para Android

El dispositivo móvil cuenta con un sistema operativo Android, para lo cual es necesario desarrollar una aplicación que permita al usuario interactuar con el sistema de video y medición de proximidad por medio de una interfaz amigable con el mismo.

Se diseña la aplicación para Android porque la mayoría de dispositivos móviles cuentan con este sistema además Android es de naturaleza “open source”, lo que facilita el acceso y modificación del sistema operativo, otorgando permisos sobre casi todos los elementos.

Para la creación de la aplicación se utiliza el entorno de desarrollo Android Studio, este software está diseñado para el desarrollo de aplicaciones de alto nivel para la plataforma Android, proporcionando una serie de librerías y complementos que facilitan la programación orientada a objetos.

4.1. Creación de la aplicación

El nuevo proyecto se inicia en el entorno de desarrollo Android Studio IDE, con todas las herramientas previamente instaladas, facilitando al desarrollador el uso de las mismas en la creación de su propia aplicación. Pasos para la creación de un nuevo proyecto en Android Studio:

Paso 1: Se selecciona en la barra de herramientas Start a new Android Studio Project, tal como se observa en la Figura 65.

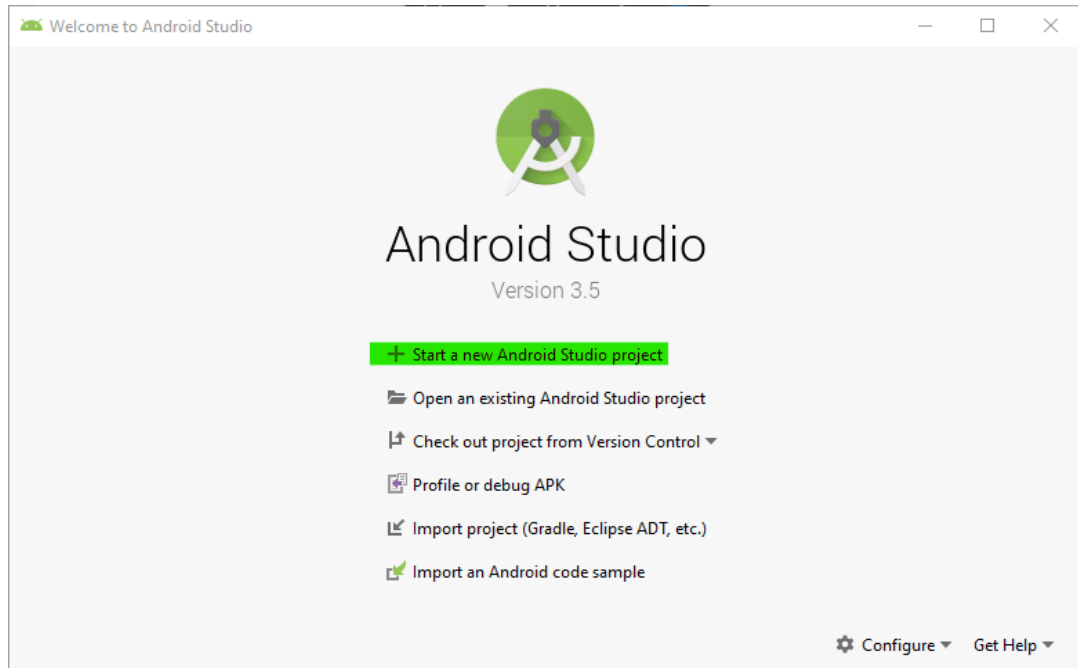


Figura 65. Welcome to Android Studio

Fuente: Elaboración propia.

Paso 2: se selecciona la plantilla base a usar, para tener todo el control del diseño de la aplicación se escoge la plantilla (Empty Activity), como se muestra en la figura 66.

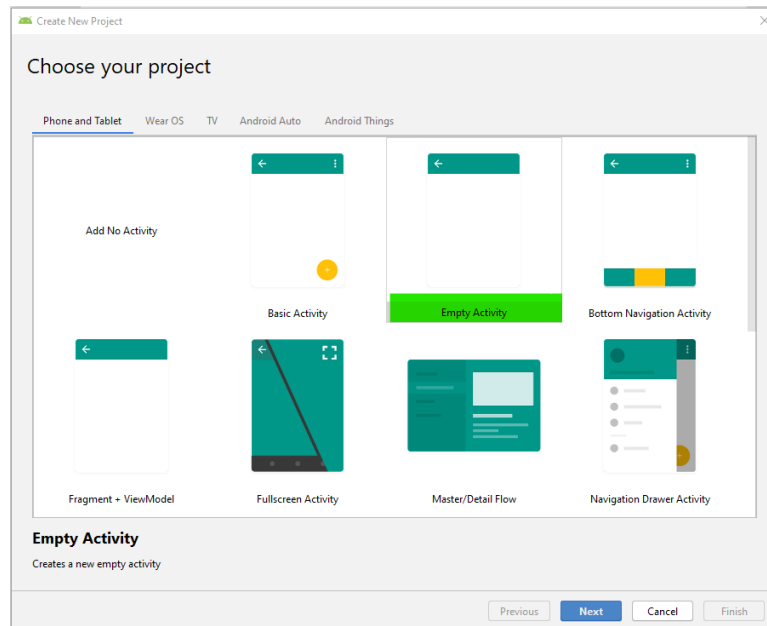


Figura 66. Ventana de Choose Your Project.

Fuente: Elaboración propia.

Paso 3: Se le otorga un nombre al proyecto, cabe mencionar que por defecto este nombre pasara hacer el mismo de la aplicación a mostrarse en el dispositivo móvil. El proyecto se llama “BackwardView”, y para terminar las configuraciones previas se da click en Finish.

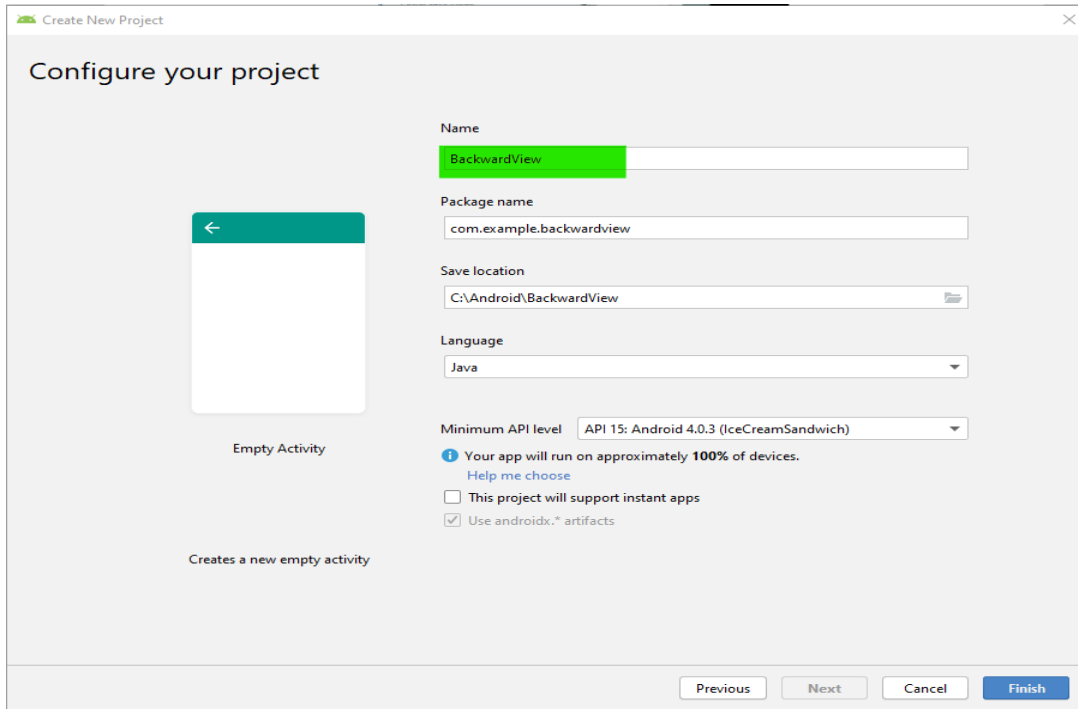


Figura 67. Ventana de Configure Your Project

Fuente: Elaboración Propia

4.2. Activity de BackwardView.

Cada una de las pantallas a mostrarse en una aplicación Android es denominada “*Activity*”, es decir, es la clase básica para generar una pantalla. Para el proyecto se planea utilizar 4 pantallas:

- ✚ Pantalla Inicial > Usuario y contraseña.
- ✚ Pantalla Principal > Es básicamente la aplicación diseñada para el dispositivo móvil en el sistema de video y medición de proximidad durante el retroceso de un vehículo.
- ✚ Pantalla de Configuración > Configuraciones del sistema.
- ✚ Pantalla de Ayuda > Recomendaciones para el usuario.

A cada actividad está enlazado un archivo .xml en el cual se realiza la interfaz de usuario, introduciendo botones, títulos, barras de progreso, imágenes y otros componentes más complejos como Webview o Videoview y uno o más archivos de tipo Java en los cuales se va realizar la programación, haciendo referencias a los componentes creados en el archivo .xml

- ✚ Archivo .xml > Diseño de la Interfaz de usuario.
- ✚ Archivos .java > programación de los elementos.

4.2.1. Pantalla de Inicio – Login Activity.

En esta primera actividad se ingresaran los datos de usuario y contraseña, esto se implementa por motivos de seguridad para que usuarios no autorizados no accedan al sistema.

Para esto el archivo login.xml cuenta con los siguientes elementos:

- ✚ ImageView “Logo”, es la imagen que hace referencia a la universidad UNAN-Managua.
- ✚ ImageView(icono usuario), EditText “Usuario”, es un bloque de entrada de texto, para que el usuario ingrese su cuenta.
- ✚ ImageView(icono de contraseña), EditText “Contraseña”, es un bloque de entrada de texto, para el ingreso de la contraseña.
- ✚ ImageView(icono fresa), Button “Conectar”, es el botón de ingreso al programa principal, es decir, lanza la actividad principal llamada “Main Activity”, después inmediatamente cierra Login Activity.
- ✚ ImageView(icono engranaje), Button “Configuración”, es el botón de ingreso a la pantalla de configuraciones, si los datos de usuario y contraseña son correctos el flujo de programa nos lanza “Configuración Activity”.
- ✚ ImageView(icono ?), Button “Ayuda”, es el botón de ingreso a la pantalla de ayuda donde muestra tips al usuario con información referente al contenido de la aplicación.

La figura 68 muestra los elementos descritos anteriormente que pertenecen a la LoginActivity.xml



Figura 68.LoginActivity.xml(Pantalla de inicio)

Fuente: Elaboración Propia

A todos los elementos que tengan una acción dinámica en la aplicación se les debe de otorgar un identificador para que se puedan manejar desde archivo de actividad en lenguaje de programación Java. Para la pantalla de inicio los componentes dinámicos se les asignan los siguientes nombres de identificación tal como se muestra en la Tabla:

Tabla 5. Componentes dinámicos de Login Activity.

Tipo	Nombre	Identificador
ImageView	-	im_logo
ImageView	-	im_user
ImageView	-	im_password
ImageView	-	imar_modi
ImageView	-	im_help
ImageView	-	im_config
EditText	Usuario	txt_usuario
EditText	Contraseña	txt_password
Button	Conectar	btn_conectar
Button	Ayuda	btn_help
Button	Configuración	btn_configuracion

Fuente: Elaboración Propia

Con estos identificadores de cada componente, se logra acceder desde archivo Login Activity.java, aquí se realiza la programación de la interfaz de inicio tomando en cuentas los siguientes requerimientos:

- ✚ El campo usuario debe de recibir el valor tipo String con el nombre del usuario, la longitud máxima de la cadena debe de ser de 8 caracteres.
- ✚ El campo contraseña debe de recibir el valor tipo entero con el password del usuario, este número entero debe de estar compuesto por 4 cifras.

- ✚ Al accionar el botón “CONECTAR” independiente de la información ingresada en los campos usuario y contraseña, se lanza la pantalla principal de la aplicación, donde el usuario tiene acceso a las mediciones y video provenientes de la Raspberry Pi.
- ✚ Al accionar el botón “CONFIGURACION” si la información ingresada en los campos anteriores es correcta debe de lanzar la pantalla de configuraciones. En caso de ser incorrecta la información debe de mostrar un mensaje de alerta al usuario.
- ✚ Al accionar el botón “AYUDA” debe lanzar la actividad ayuda, donde brindara al usuario la información referente a la aplicación, y unos tips.

Conociendo todos los requerimientos se realiza la programación en Java y se establecen los elementos para desarrollar la interfaz gráfica.

4.2.2. Pantalla de configuración.

En esta actividad, se presenta al usuario la posibilidad de modificar los parámetros referentes al sistema de video, es decir, que en esta pantalla se muestra los controles de la cámara Raspberry entre los cuales se tiene:

- ✚ Tamaño de la imagen
- ✚ Fotos por segundo
- ✚ Formato de codificación del video
- ✚ Latencia en la imagen
- ✚ Brillo
- ✚ Contraste
- ✚ Color

Para poder acceder a esta pantalla es necesario llenar correctamente los campos de usuario y contraseña en la pantalla LOGIN ACTIVITY.

Para esto en el diseño de esta pantalla se utilizó los siguientes elementos:

- ✚ TextView “Parámetros BackwardView”, es el título que hace referencia a la función realizada por dicha actividad.
- ✚ ListView “lista_conf”, esta herramienta despliega una serie de resoluciones de imagen adaptables al cámara Raspberry, esto responde a que la misma requiere de relaciones de ancho y altura adecuados al lente de la misma.
- ✚ Button “LISTO”, es el botón que lanza nuevamente la pantalla principal, es decir, que cierra la Configuración Activity y lanza la Main Activity. Luego de realizados los cambios devuelve al usuario a la aplicación.
- ✚ Webview: para mostrar al usuario los controles de la cámara Raspberry. Al igual que en el Webview usado en la pantalla principal, con parámetros preestablecidos como son la dirección IP y puerto se accede a las configuraciones del servidor de video.

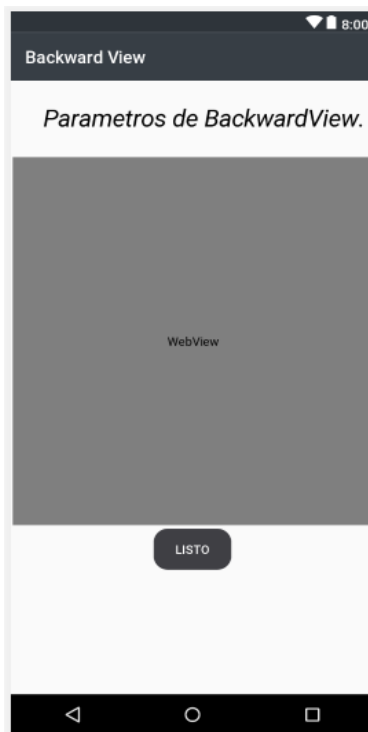


Figura 69. Actividad configuración.

Fuente: Elaboración propia.

A todos los elementos que tengan una acción dinámica en la aplicación se les debe de otorgar un identificador para que se puedan manejar desde archivo de actividad en lenguaje de programación Java. Para la pantalla de Configuración los componentes dinámicos se les asignan los siguientes nombres de identificación tal como se muestra en la Tabla 5.

Tabla 6. Componentes dinámicos de la ConfActivity

Tipo	Nombre	Identificador
TextView	Parámetros BackwardView	tv_parametros
ListView	Lista_conf	lista_conf
Button	Listo	btn_listo
Webview	Web	web_1

Fuente: Elaboración propia

4.2.3. Pantalla de ayuda.

En esta actividad se le muestra al usuario la versión de la Apps y los campos que tiene que llenar para acceder de forma satisfactoria a la pantalla principal y de configuración, por lo cual esta pantalla está conformada por los siguientes elementos:

- ✚ TextView: Se encarga de mostrar información importante para el usuario.
- ✚ ImageView: Su objetivo es trasponerse como icono de la aplicación.
- ✚ Button: Al ser presionado lo redirigirá a la pantalla del Login.



Figura 70. Pantalla de Ayuda.

Fuente: Elaboración propia

La tabla 6 muestra los identificadores necesarios para realizar la programación de Java.

Tabla 7. Componentes dinámicos de HelpActivity.

Tipo	Nombre	Identificador
TextView	-	txt_version
TextView	-	txt_requerimientos
TextView	-	txt_user_re
TextView	-	txt_passw_re
TextView	-	txt_listo_re
TextView	-	txt_backwardview_re
Imageview	-	ima_view
Button	-	button

Fuente: Elaboración propia.

4.3. Escala de proximidad

Para esto se utiliza una ProgressBar, esta escala tiene como función principal transmitir al usuario ciertos rangos de proximidad, esto permite que personas que no manejen fácilmente relaciones de distancia, es decir, que no reconozcan lo que representa medida de proximidad a un obstáculo en relación al riesgo que este representa, denoten esta característica al instante.

Esta escala le permite al usuario determinar en qué rango de seguridad se encuentra de esta forma se definen los colores convencionales como por ejemplo el verde cuando se encuentre el vehículo a una distancia adecuada hasta llegar al color rojo que identifica el peligro ante la cercanía a un obstáculo, además de otros colores que identifican rangos intermedios, en la Tabla 8 se muestra los diferentes colores utilizados en esta escala:

Tabla 8. Relación de color - Proximidad.

Distancia	Riesgo	Color	
≥ 100 cm	Muy bajo	Verde	
< 100 cm y ≥ 75 cm	Bajo	Verde-amarillo	
< 75 cm y ≥ 50 cm	Medio	Amarillo	
< 50 cm y ≥ 25 cm	Alto	Naranja	
< 25 cm	Muy Alto	Rojo	

Fuente: Elaboración propia.

Para poder actualizar constantemente la ProgressBar, fue necesario ejecutar esta tarea en un segundo hilo, en donde además se realiza la recepción de los datagramas y la actualización de la salida de texto por pantalla llamada "DISTANCIA".

4.4. Alarmas Sonoras

Adicionalmente a los componentes visuales, la aplicación emite alarmas sonoras, estos sonidos son similares a los que emiten los equipos comerciales utilizados para el retroceso del vehículo.

Para esto se utilizó un sonido de base de corta duración llamado:

Beep.mp3

Este sonido se reproduce un sin número de veces en la ejecución de la aplicación, para esto se utiliza el reproductor de sonido propio de Android llamado Sound Pool, el cual recibe los parámetros:

- Archivo de sonido
- Canal
- Número de repeticiones
- Frecuencia

En la aplicación se emite dicho sonido a frecuencias diferentes, las cuales son proporcionales a los rangos de proximidad hacia un obstáculo, es decir, que a grandes distancias de proximidad la frecuencia es muy baja hasta llegar a una distancia riesgosa en donde el sonido se reproduce a una alta frecuencia, existen pasos intermedios tal como se muestra en la Tabla 9:

Tabla 9. Sonidos de escala de proximidad.

Distancia	Riesgo	Frecuencia	Duración	Pulsos(repetitivos)
≥ 100 cm	Muy bajo	0.1 Hz	6.41 s	10
< 100 cm y ≥ 75 cm	Bajo	0.7 Hz	4.86 s	8
< 75 cm y ≥ 50 cm	Medio	1.1 Hz	3.90 s	6
< 50 cm y ≥ 25 cm	Alto	1.4 Hz	2.48 s	4
< 25 cm	Muy Alto	1.8 Hz	1.51 s	2

Fuente: Elaboración propia.

Cabe mencionar que esta tarea se actualiza constantemente, es decir, que en la ejecución de la aplicación la frecuencia del sonido variará acorde a los cambios de medición de proximidad hacia un obstáculo, para que se pueda realizar es necesario implementar esta tarea en un segundo hilo.

XI. Pruebas de funcionamiento del sistema de adquisición de video y medición de distancia durante el retroceso de un vehículo en un dispositivo móvil mediante un prototipo a escala en un ambiente controlado.

En este apartado se evidencia el funcionamiento del servidor de video, la comunicación socket-UDP (transporte de los datos de arduino valor mínimo), donde lo antes mencionado estará siendo recepcionado por la app (Backward View) creada en la plataforma Android Studio.

Paso 1.Montaje de los componentes a utilizar.

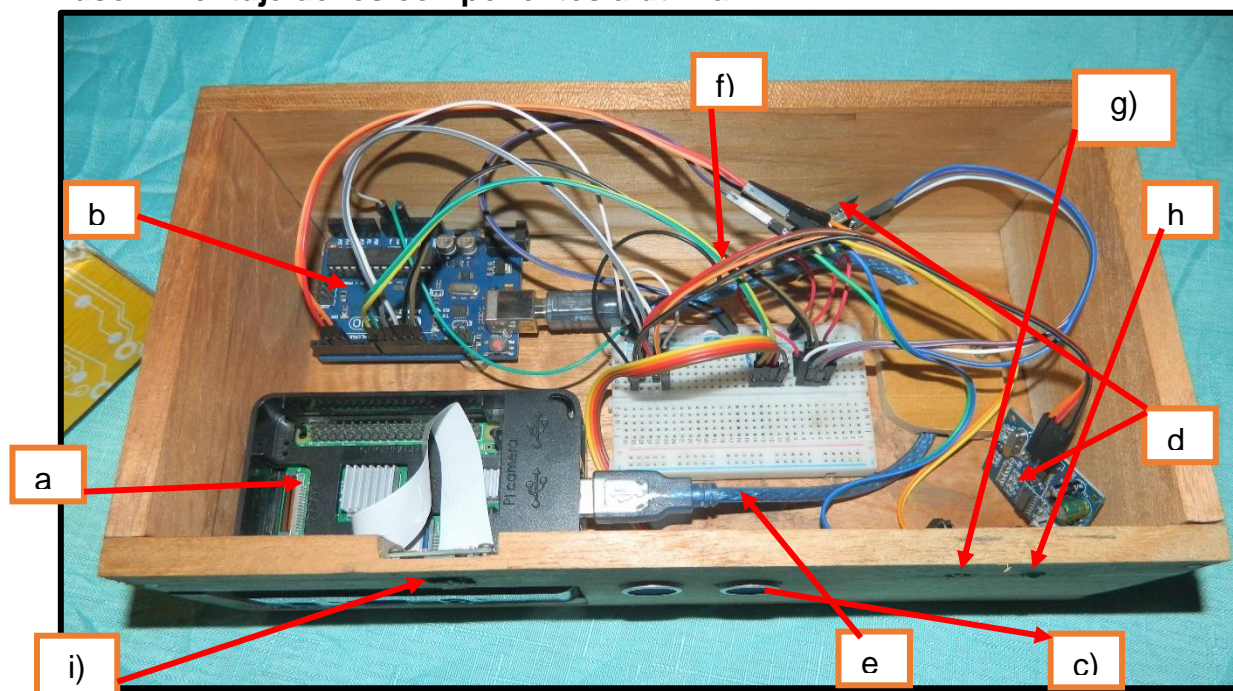


Figura 71.Ensamble.

Fuente: Elaboración Propia

En la figura 71 se observa como esta constituido internamente el sistema de video y medicion de distancia. El sistema esta formado por los siguientes componentes:

- | | |
|-------------------------------------|---------------------------------|
| a) Raspberry Pi 3 B+ | e) Cable de comunicación Serial |
| b) Arduino Uno Pro | f) Cables de conexiones |
| c) Sensor de proximidad HC-SR04 | g) Led rojo |
| d) Sensores de proximidad JSN-SR04T | h) Led verde |
| | i) Modulo de camara nativa |

Paso 2. Arrancar el emulador donde se estaran haciendo pruebas de la app.

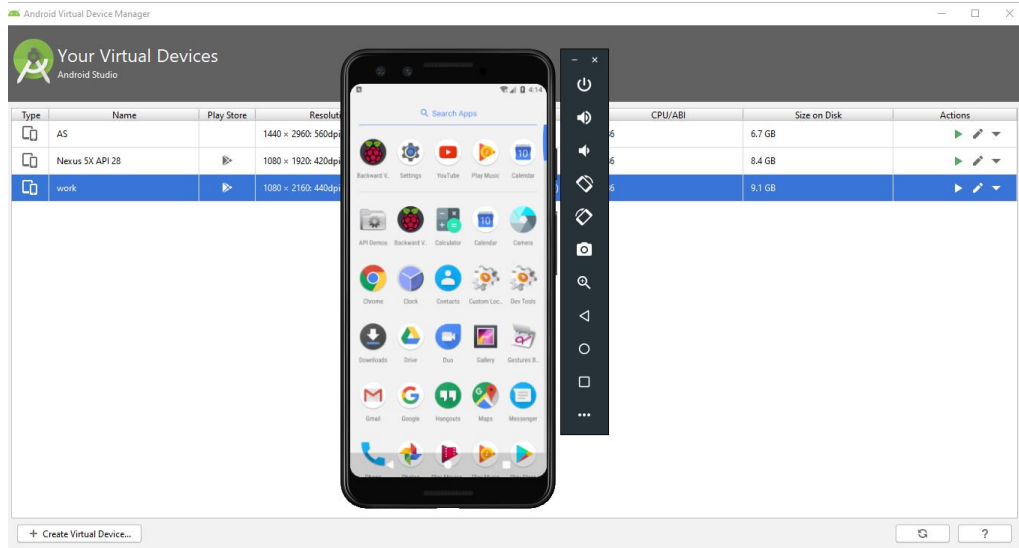


Figura 72. Emulador Android 7.1

Fuente: Elaboración propia

En la figura 72 se muestra el emulador con un sistema android 7.1 (Nougat) originario de la plataforma Android Studio.

Paso 3. Se empiezan a realizar pruebas de distancia.

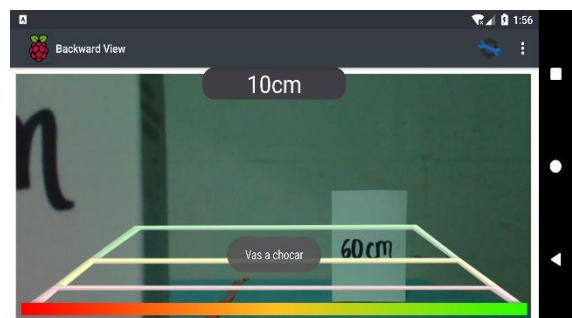
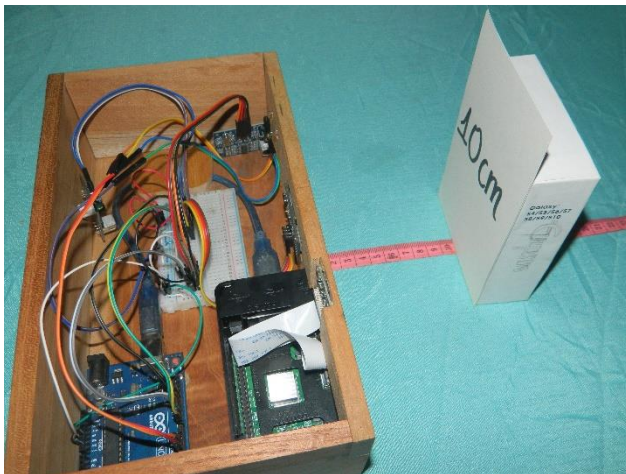


Figura 73. Prueba de funcionamiento con un objeto a 10 cm.

Fuente: Elaboración propia.

En la figura 73 se puede observar que se obtiene el valor de 10 cm en el emulador de la app con un mensaje emergente "VAS A CHOCAR".

Paso 4. Prueba de funcionamiento a una distancia de 30 cm.

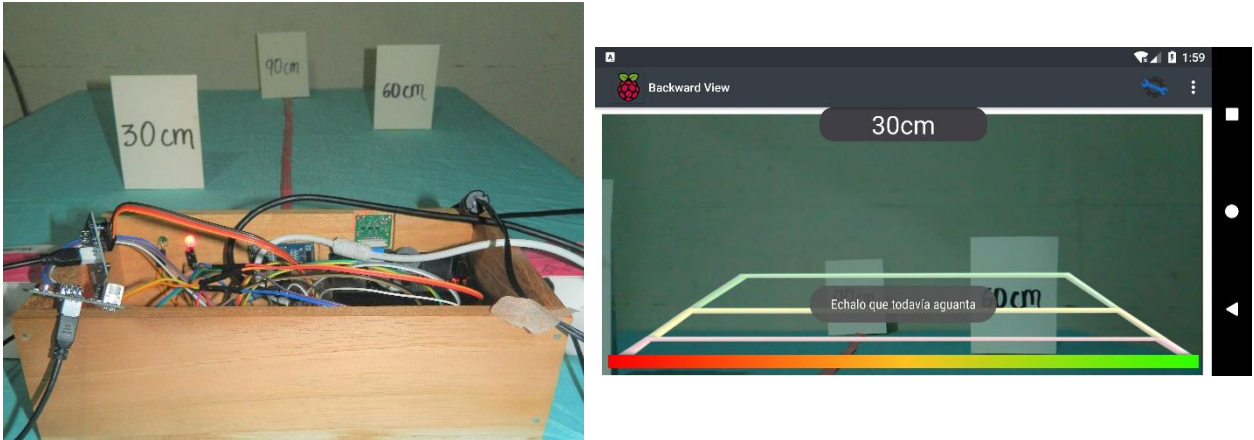


Figura 74. Prueba de funcionamiento a 30 cm.

Fuente: Elaboración propia.

La figura 74 muestra el objeto a 30 cm y la app refleja el mensaje emergente “EHALO QUE TODAVIA AGUANTA”.

Paso 5. Prueba de funcionamiento a una distancia de 60 cm.

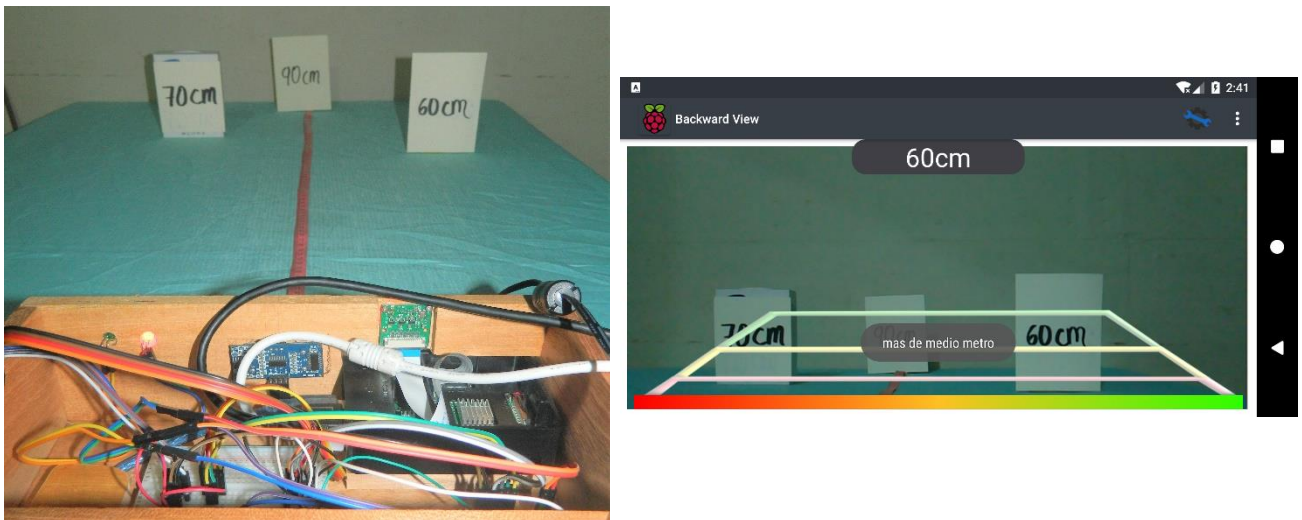


Figura 75. Prueba de funcionamiento a 60 cm.

Fuente: Elaboración propia

Al realizar la prueba de funcionamiento a una distancia de 60 cm el emulador a través de un toast muestra un mensaje emergente en la pagina principal “MÁS DE MEDIO METRO”.

Paso 6. Prueba de funcionamiento a 80 cm.

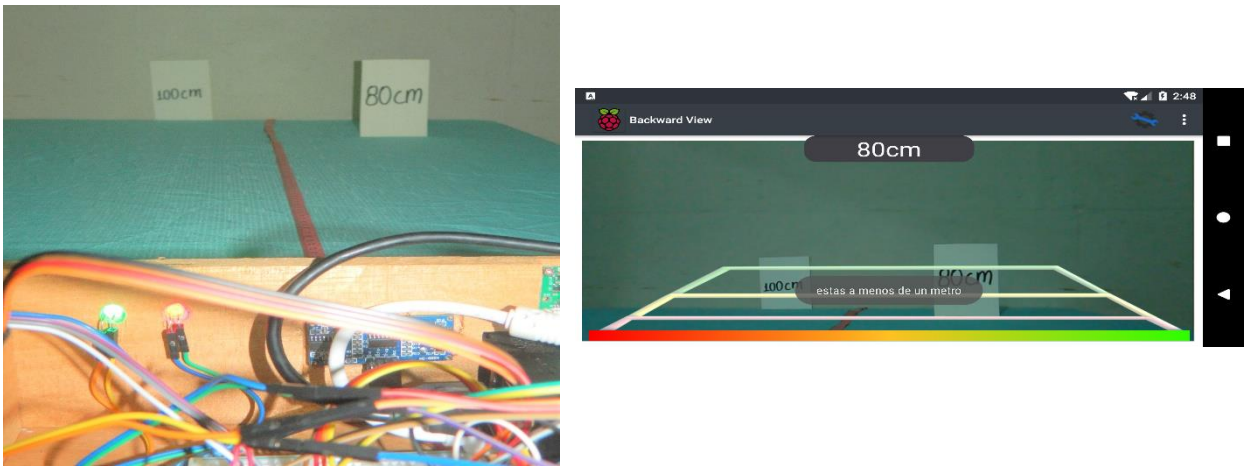


Figura 76. Prueba de funcionamiento a 80 cm.

Fuente: Elaboración propia.

Al colocar un objeto a una distancia de 80 cm se muestra el mensaje emergente “ESTAS A MENOS DE UN METRO”, al mismo tiempo que se puede observar el video en tiempo real.

Paso 7. Prueba de funcionamiento a una distancia de 101 cm.

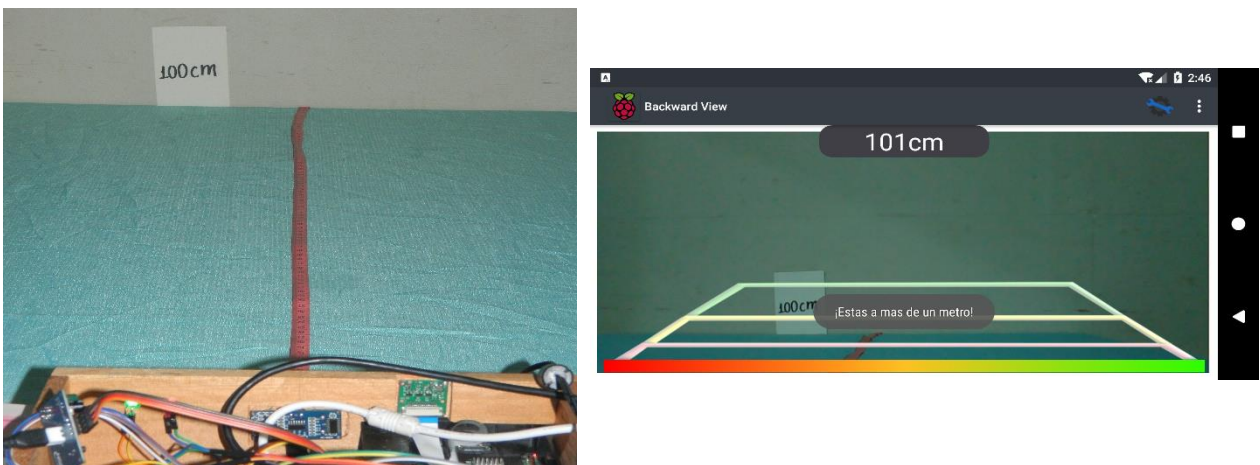


Figura 77. Prueba de funcionamiento a 101 cm.

Fuente: Elaboración propia.

Al realizar la prueba de funcionamiento en tiempo real a una distancia de 101 cm como lo muestra la figura 84 se antepone el mensaje emergente “ESTAS A MAS DE UN METRO”.

XII. Presupuesto de los componentes utilizados para la elaboración del prototipo a escala del sistema de adquisición de video y medición de distancia durante el retroceso en un dispositivo móvil.

Tabla 10. Presupuesto de los componentes utilizados.

Cantidad	Componentes	Costo Unitario	Costo Total
1	Raspberry pi 3 b+	\$ 60	\$ 60
1	Arduino uno pro	\$ 16	\$ 16
1	Módulo de cámara para raspberry pi 3 b+	\$ 14.79	\$ 14.79
1	Tarjeta de memoria microSD 32gb	\$ 7.40	\$ 7.40
1	Encapsulado	\$ 11.83	\$ 11.83
Sensores ultrasónicos			
2	jsn-sr04t	\$ 13.02	\$ 26.04
1	hc-sr04	\$ 8	\$ 8
Led indicadores			
2	led de transmisión (amarillo)	\$ 0.18	\$ 0.36
2	led del sistema encendido (rojo)	\$ 0.18	\$ 0.36
Total			\$ 144.78

Fuente: Elaboración Propia.

XIII. Conclusiones

Se logró identificar la necesidad de crear un sistema de adquisición de video y medición de distancia durante el retroceso en un dispositivo móvil, con la información aportada por parte del personal docente, administrativo y estudiantil de la Universidad Nacional Autónoma de Nicaragua (UNAN - Managua) además de lograr alcanzar los objetivos siguientes:

Se realizó un análisis del sistema de aparcamiento actual vs sistema de aparcamiento propuesto que detalló las ventajas de dichos sistemas.

Se logró describir los componentes que integran el sistema de adquisición de video y medición de distancia durante el retroceso de un vehículo en un dispositivo móvil a través de información esencial que sirvió de base para la realización del siguiente objetivo.

Se diseñó el sistema de adquisición de video y medición de distancia del vehículo, creando el esquema electrónico con su respectivo diagrama de flujo de funcionamiento de los microcontroladores arduino, raspberry pi, también se realizó el diseño de la app Backward view en la plataforma android studio donde se programó en lenguaje java, por otra parte en la raspberry se creó el servidor de video haciendo uso del programa uv4l, además se creó el punto de acceso necesario para la realización del servidor de comunicación socket UDP el que se encarga de transportar los datos proporcionados por el arduino a la app (cliente).

Para finalizar se realizaron pruebas de funcionamiento en un ambiente controlado del sistema de adquisición de video y medición de distancia durante el retroceso de un vehículo en un dispositivo móvil, constituida por 7 pasos (mediciones de valores diferentes), obteniendo como resultado un margen de error del 1%.

XIV. Recomendaciones

Al usuario:

- ❖ Antes de utilizar la app Backward view dirigirse a la opción de ayuda para conocer cómo funciona la misma.
- ❖ En caso de adquirir este sistema abocarse al personal idóneo para esta tarea.
- ❖ Privacidad en el acceso de la app.
- ❖ Garantizar el uso adecuado del equipo para lograr mantenerlo en óptimas condiciones.

A investigadores:

En caso de dar seguimiento e implementar mejoras a este proyecto contar con los siguientes elementos:

- ❖ Conocimientos avanzados en lenguaje de programación java con enfoque a android studio.
- ❖ Conocimientos básicos en lenguaje de programación Python.
- ❖ Conocimiento avanzado en lenguaje de programación C++.
- ❖ Conocimiento básico en servidores creados en sistemas Linux.
- ❖ Conocimiento avanzado en redes de comunicación para crear el Check Point.
- ❖ Si se debe hacer cambio de algún componente cerciorarse de ser el más adecuado para evitar futuras complicaciones al ejecutar el sistema.

Bibliografía

(s.f.). Obtenido de RASPBERRY:

<https://www.raspberrypi.org/forums/viewtopic.php?p=1570820&t=257586&sid=4a20aabed334502dc4957a5154ac173b>

(12 de MAYO de 2018). Obtenido de raspberryparanovatos:

<https://raspberryparanovatos.com/tutoriales/instalar-raspbian-en-raspberry-pi-noobs/>

(2 de septiembre de 2018). Obtenido de taringa: https://www.taringa.net/+hardwarezone/las-4-mejores-carcasas-para-la-raspberry-pi-3-model-b-y-el_1dcyz3

Adictoalcodigo.blogspot.com. (s,f de s,f de 2017). Obtenido de *Adictoalcodigo.blogspot.com*:

<https://adictoalcodigo.blogspot.com/2016/07/ventajas-y-desventajas-de-programar-en-c.html>

Allbiz. (sf de sf de sf). Obtenido de Allbiz: <https://mx.all.biz/sensores-capacitivos-sensores-capacitivos-marca-g7585>

Anonimo. (S,f de S,f de S,f). *Platea.pntic.mec.es*. Recuperado el 17 de 10 de 2019, de

Platea.pntic.mec.es: <http://platea.pntic.mec.es/~lgonzale/tic/imagen/conceptos.html>

Basterra, Berteia, Borello, Castillo y Venturi. (sf de sf de sf). *Android OS*. Obtenido de Android OS:

<https://androidos.readthedocs.io/en/latest/data/caracteristicas/>

blogdelaautoescuela.com. (5 de Mayo de 2016). *Blog de la auto escuela*. Obtenido de Blog de la auto escuela: <http://www.blogdelaautoescuela.com/america/ventajas-de-los-sensores-de-aparcamiento/>

Canales, Alvarado y Pineda. (S,f de S,f de 1994). Metodología de la Investigación. *Publicación de la ORGANIZACIÓN PANAMERICANA DE LA SALUD, 2da. edición*. Washington, D.C., EEUU.

CANSON. (2019). *CANSON INFINITY*. Recuperado el 17 de 10 de 2019, de CANSON INFINITY:

<https://www.canson-infinity.com/es/faq/que-es-la-resolucion-de-una-imagen>

cooperativo. (sf de sf de sf). Obtenido de NAYLAMP: <https://naylampmechatronics.com/sensores-proximidad/326-sensor-ultrasonido-jsn-sr04t.html>

Diarioelectronicohoy.com. (S,f de S,f de S,f). Obtenido de *Diarioelectronicohoy.com*:

<https://www.diarioelectronicohoy.com/modulo-de-camara-para-raspberry-pi/>

electronica2000.com. (s.f.). Obtenido de *electronica2000.com*:

<http://www.electronica2000.com/temas/sensores-proximidad-aplicaciones.htm>

Foto Nostra.com. (s.f.). *Foto Nostra*. Recuperado el 17 de 10 de 2019, de Foto Nostra:

<https://www.fotonostra.com/digital/fomacionimagen.htm>

Guerrero. (21 de septiembre de 2014). *PLUSELECTRIC*. Obtenido de PLUSELECTRIC:

<https://pluselectric.wordpress.com/2014/09/21/arduino-uno-especificaciones-y-caracteristicas/>

- How, K. (15 de Marzo de 2019). *Digital Guide IONOS*. Obtenido de Digital Guide IONOS:
<https://www.ionos.es/digitalguide/servidores/know-how/udp-user-datagram-protocol/>
- Imagenmedia.Blogspot. (8 de Marzo de 2016). *Mundo Digital*. Recuperado el 21 de 10 de 2019, de Mundo Digital: <http://imagenmedia.blogspot.com/2016/03/que-es-una-imagen-digital.html?m=1>
- Intercomunicación y Seguridad en Redes*. (16 de Julio de 2012). Obtenido de Intercomunicación y Seguridad en Redes: <https://lordratita.wordpress.com/2012/07/16/unidad-4-protocolo-de-datagramas-de-usuario-udp-2/>
- Javier, B. S. (2013). Obtenido de monografia.com:
<https://www.monografias.com/trabajos99/medicion-y-control-del-nivel-liquido-sistema-scada/medicion-y-control-del-nivel-liquido-sistema-scada2.shtml>
- lenguajesdeprogramacion.net. (s,f de s,f de s,f). *Lenguajes de programacion*. Obtenido de Lenguajes de programacion: <https://lenguajesdeprogramacion.net/java/>
- Lora, F. (29 de 05 de 2019). *Top Fotografía*. Recuperado el 17 de Octubre de 2019, de Top Fotografía: <http://www.topfotografia.net/Fotografia/teoria-de-la-fotografia/sensor-cmos/sensor-cmos.html>
- LUIS LLAMAS. (septiembre de 2019). Obtenido de LUIS LLAMAS:
<https://www.luisllamas.es/raspberry-pi-instalar-noobs/>
- Marmolejo. (23 de Mayo de 2017.). *HETPRO*. (I. d. HeTPro, Editor) Recuperado el 28 de Noviembre de 2019, de HETPRO: <https://hetpro-store.com/que-es-arduino/>
- masadelante.com. (s,f de s,f de s,f). *masadelante.com*. Obtenido de masadelante.com:
<https://www.masadelante.com/faqs/udp>
- Motor y Racing. (s,f de s,f de s,f). *Motor y Racing*. Obtenido de Motor y Racing:
<https://www.motoryracing.com/coches/noticias/los-sensores-de-aparcamiento-y-su-evolucion/>
- Nylam Mechatronic - Perú*. (sf de sf de sf). Obtenido de Nylam Mechatronic - Perú:
<https://naylampmechatronics.com/sensores-proximidad/403-sensor-de-proximidad-inductivo-lj18a3-5-zax-npn.html>
- Pineda, G. (8 de Marzo de 2016). *MUNDO DIGITAL*. Obtenido de MUNDO DIGITAL:
<http://imagenmedia.blogspot.com/2016/03/que-es-una-imagen-digital.html>
- PNGOCEAN. (s,f de s,f de s,f). *PNGOCEAN*. Obtenido de PNGOCEAN:
<https://www.pngocean.com/gratis-png-clipart-dtyin>
- Porto, J. P. (s,f de s,f de 2009). *Definición.de*. Recuperado el 17 de 10 de 2019, de Definición.de:
<https://definicion.de/video-camara/>
- Porto, J. P. (s,f de s,f de 2012). *Definicion.de*. Recuperado el 17 de 10 de 2019, de Definicion.de:
<https://definicion.de/pixel/>

- Porto, J. P. (s,f de s,f de 2014). *Definicion.de*. Recuperado el 17 de 10 de 2019, de Definicion.de:
<https://definicion.de/red-inalambrica/>
- Programo ERGO SUM. (s,f de s,f de s,f). *Programo ERGO SUM*. Obtenido de Programo ERGO SUM:
<https://www.programoergosum.com/cursos-online/raspberry-pi/244-iniciacion-a-python-en-raspberry-pi/que-es-python>
- Quilapan, L. (26 de septiembre de 2014). *Wiki Inalambrica*. Obtenido de Wiki Inalambrica:
<https://sites.google.com/site/wikiinalambricas/definicion-y-tipos-de-redes/definicionytiposderedes>
- Raspberrypi.org. (S,f de S,f de S,f). *Raspberrypi.org*. Obtenido de Raspberrypi.org:
<https://www.raspberrypi.org/documentation/hardware/raspberry-pi/bcm2837b0/README.md>
- Raúl Diosdado. (abril de 2014-2018). Obtenido de zona market:
<https://www.zonamaker.com/arduino/modulos-sensores-y-shields/ultrasonido-hc-sr04>
- RedUsers. (13 de 02 de 2013). Recuperado el 17 de 10 de 2019, de RedUsers:
<http://www.redusers.com/noticias/que-es-una-red-informatica/>
- Sanchez, K. (2015 de Julio de 10). *MOTORTREND GROUP*. Recuperado el 09 de Octubre de 2019, de MOTORTREND GROUP: <http://www.motortrendenespanol.com/noticias/11506-modelos-con-camara-de-vision-trasera-estandar/>
- SD Memory Card Formatter. (s.f de s.f de s.f). *SD Memory Card Formatter*. Obtenido de SD Memory Card Formatter: https://www.sdcard.org/downloads/formatter_4/
- (sf de sf de sf). Obtenido de STARWARE: <https://tienda.starware.com.ar/producto/raspberry-pi-3-b-kit-stem-element14-base-fan-fuente/>
- sn. (s.f.). Obtenido de TECNOPURA: <http://www.tecnopura.com/producto/modulo-camara-1080p-para-raspberry-pi-webcam-5mp-rev-1-3/>
- Sosa, D. (Ed.). (16 de 03 de 2016). *Autoscosmos.com*. Recuperado el 09 de 10 de 2019, de Autoscosmos.com: <http://noticias.español.autoscosmos.com/2016/03/16/camara-de-vision-trasera-hizo-su-aparicion-hace-15-años>
- Tecnologia Informatica*. (sf de sf de sf). Obtenido de Tecnologia Informatica: <https://tecnologia-informatica.com/que-es-red-inalambrica-seguridad-wifi/>
- Test de Velocidad*. (11 de 08 de 2016). Obtenido de Test de Velocidad:
<https://www.testdevelocidad.es/2016/08/11/conoce-cuales-todos-los-estandares-wi-fi-existen/>
- Vilchez, A. (2 de 04 de 2009). *EcuRed*. Obtenido de EcuRed: <https://www.ecured.cu/Android>
- xataka.com*. (S.f de S,f de S,f). Obtenido de xataka.com:
<https://www.xataka.com/ordenadores/raspberry-pi-3-model-b-analisis-mas-potencia-y-mejor-wifi-para-un-minipc-que-sigue-asombrando>

Anexos

Anexo A. Modelo de Encuesta utilizado en la recolección de datos.



UNIVERSIDAD
NACIONAL
AUTÓNOMA DE
NICARAGUA,
MANAGUA
UNAN-MANAGUA

Encuesta

Esta encuesta, pretende obtener datos necesarios para la realización de un sistema de adquisición de video y medición de distancia durante el retroceso en un dispositivo móvil.

Edad: _____

Sexo:

- Hombre
- Mujer

Pertenece a:

- Estudiante.
- Personal Docente.
- Personal Administrativo.

Modelo de su auto: _____

1. ¿De qué año es el vehículo que usted conduce?

- 2000 al 2017.
- 2017 a la actualidad.
- Otros: _____

2. ¿Su vehículo posee un sistema de ayuda para el retroceso?

- Si
- No

Si su respuesta es negativa conteste la siguiente pregunta. En el caso de ser afirmativa salte a la pregunta #5

3. ¿Le gustaría implementar un sistema de ayuda para el momento de retroceder en su vehículo?

- Si
- No
- Tal Vez

4. ¿Considera usted necesario la implementación de un sistema de ayuda en retroceso para vehículo?

- Si
- No

5. ¿Considera usted poner en marcha un nuevo sistema de ayuda para el retroceso a través de una aplicación móvil?

- Si
- No

6. ¿Le gustaría que este sistema sea de bajo costo?

- Si
- No

7. ¿Cuál de las siguientes ventajas considera usted que obtendría de ser implementado el sistema?

- Limitar el índice de colisiones en la parte trasera del vehículo.
- Modernizar su vehículo a través de esta herramienta.
- Mejor visión en los puntos ciegos al momento de retroceder.
- El costo del sistema no es muy elevado.
- Todas las anteriores

8. ¿Cuál de las siguientes desventajas considera usted que obtendría de ser implementado el sistema?

- No es un sistema integrado.
- No es un sistema automático.