



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

**“DESARROLLO DE UNA INTERFAZ PARA CONTROL REMOTO
DE MOTORES USANDO UN DISPOSITIVO EMBEBIDO”**

INFORME DE PROYECTO INTEGRADOR

Previa a la obtención del Título de:

INGENIERO/A EN CIENCIAS COMPUTACIONALES

ANA CAROLINA ARIAS PACHECO

LUIS ENRIQUE PÉREZ LÓPEZ

GUAYAQUIL – ECUADOR

AÑO: 2015

AGRADECIMIENTOS

Quiero agradecer en primer lugar a Dios, por darme una vida hermosa con muchas oportunidades, sobre todo por darme unos padres maravillosos a quienes agradezco el esfuerzo que hicieron durante toda su vida para darme lo mejor y gracias al amor que me dieron soy quien soy en este momento.

Quiero agradecer también a mi esposo, quién llegó a mi vida para completarla, apoyándome en todo sentido y por darme un hijo hermoso que es desde ya la luz en nuestras vidas.

Agradezco a mis profesores quienes me formaron durante la vida académica y me enseñaron mucho más que conocimientos técnicos sino integrales.

Ana Arias

Quiero agradecer a mis padres por todo su apoyo incondicional, por nunca dudar en que alcanzaría mis objetivos y siempre estarme motivando constantemente. Gracias a ellos he cumplido esta meta con mucho éxito.

Quiero agradecer también a mis hermanos y amigos por siempre alegrar mi vida en los momentos más difíciles. Un agradecimiento especial para mi Tío Manuel por ser mi mentor y guía en mi vida profesional.

Luis Pérez

DEDICATORIA

Este proyecto lo dedico a mis padres, a mi esposo y a Alejandrino, quienes son mi motivo de vida.

Ana Arias

Dedico este proyecto a Dios ya que sin él nada es posible, también con mucho cariño a mi familia y amigos.

Luis Pérez

TRIBUNAL DE EVALUACIÓN



Ph. D. Dennis Romero

PROFESOR EVALUADOR

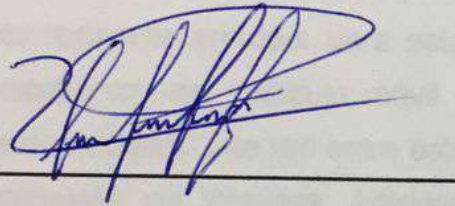


MSIG Ronald Ponguillo

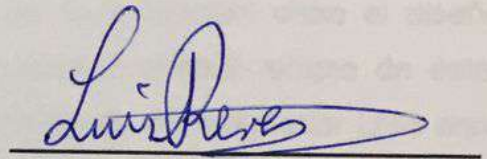
PROFESOR EVALUADOR

DECLARACIÓN EXPRESA

"La responsabilidad y la autoría del contenido de este Trabajo de Titulación, nos corresponde exclusivamente; y damos nuestro consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual"



Ana Arias Pacheco



Luis Pérez López

RESUMEN

El presente trabajo describe la implementación de una interfaz web accesible desde teléfonos móviles para el control remoto de un vehículo móvil utilizando todos los beneficios que ofrece el dispositivo embebido Raspberry Pi.

El dispositivo embebido Raspberry Pi, entre muchas funcionalidades, tiene como una ventaja principal, comunicarse con el mundo exterior utilizándose para una amplia gama de proyectos a través del uso de sensores y módulos que se integran fácilmente a sus entradas y salidas. En este proyecto Raspberry Pi actúa como componente principal para establecer un canal de comunicación entre el diseño electrónico del vehículo móvil y la interfaz web para el control remoto de éste, estableciendo una red entre estas terminales con la ayuda del adaptador USB para conexión vía Wireless. Adicionalmente, proporcionará por medio de una cámara web integrada a su circuito, el streaming de las imágenes capturadas del vehículo controlado remotamente, las cuales son visualizadas en la interfaz web; dado que los beneficios que ofrece este dispositivo embebido permite generar esta solución a bajo costo, se logra apoyar a los sectores productivos con tecnología accesible.

Para llevar a cabo este proyecto se utilizó la Metodología SCRUM, ésta es una metodología ágil y flexible para gestionar el desarrollo de software, mediante la cual se definieron las actividades a realizarse en intervalos controlados de tiempo.

ÍNDICE GENERAL

AGRADECIMIENTOS.....	ii
DEDICATORIA	iii
TRIBUNAL DE EVALUACIÓN	iv
DECLARACIÓN EXPRESA	v
RESUMEN.....	vi
ÍNDICE GENERAL.....	vii
CAPÍTULO 1	1
1 SITUACIÓN ESPECÍFICA DEL PROBLEMA A RESOLVER	1
1.1. Causas	2
1.1.1 Tecnología en la Agricultura Ecuatoriana.....	3
1.2. Efectos	4
1.3. Soluciones Similares	5
CAPÍTULO 2.....	7
2 METODOLOGÍA O SOLUCIÓN TECNOLÓGICA IMPLEMENTADA	7
2.1. Tecnologías y herramientas utilizadas en el proyecto	7
2.1.1. Raspberry Pi, dispositivo embebido	7
2.1.2. Software Node JS.....	13
2.1.3. Software Node Red	14
CAPÍTULO 3.....	19
3. RESULTADOS OBTENIDOS Y ANÁLISIS DE LOS RESULTADOS	19
3.1. Implementación del Hardware del proyecto.....	19
3.1.1. Carcasa del vehículo	19
3.1.2. Pines del Raspberry Pi	20
3.1.3. Tarjeta de conexión entre Raspberry y Placa controladora de motores de vehículo.	23
3.2. Implementación del Software del proyecto.....	26
3.2.1. Instalación de Sistema Operativo Ubuntu	26
3.2.2. Instalación de programas de utilidad para el proyecto	29

3.2.3. Configuración de SD Card y módulos a ser utilizados.....	31
3.2.4. Instalación de Node JS y Node Red.....	39
3.2.5. Implementación de Interfaz web, audio y cámara	41
3.2.6. Implementación de flujo completo en Node Red	43
CONCLUSIONES Y RECOMENDACIONES	55
BIBLIOGRAFÍA.....	57

CAPÍTULO 1

1 SITUACIÓN ESPECÍFICA DEL PROBLEMA A RESOLVER

Los avances tecnológicos a nivel mundial fortalecen los procedimientos realizados en áreas estratégicas que sustentan la economía de los países.

La sed tecnológica está llegando con fuerza al Ecuador, sin embargo los avances tecnológicos a gran escala implementados localmente no disminuyen la demanda tecnológica necesaria, esto se atribuye a varios factores como, falta de materia prima para elaboración de componentes tecnológicos, poca inversión en conocimiento e investigación en tecnologías actuales y sobretodo falta de capital para invertir en el desarrollo e implementación de tecnología que en fusión con el trabajo humano, mejore y facilite notablemente procesos que pueden ser automatizados.

La problemática abordada en este trabajo se centra en dos aspectos de la sociedad actual que pretenden con el tiempo integrar tecnología, buscando principalmente un bajo costo de inversión; ambos enfoques se referirán a áreas en la industria ecuatoriana que requieren el uso de una tecnología que aún no presenta un beneficio costo/valor para ser implementadas.

El presente trabajo busca apoyar al sector Agrícola con un sistema de vigilancia de cultivos que posibilite la supervisión remota de un cultivo por medio de un vehículo con una estructura robusta que permita recorrer los cultivos sin necesidad de movilizarse para realizar esta actividad.

Gracias al dispositivo embebido utilizado en el proyecto, éste enfoque de vigilancia de cultivos se amplía mucho más, hacia vigilancia en hogares, zonas nocturnas peligrosas, estudio de la naturaleza, en fin, miles de ámbitos en los que se puede utilizar las facilidades que ofrece el sistema implementado en este proyecto. Cabe recalcar que las funciones implementadas pueden existir actualmente, sin embargo se pretende generar soluciones tecnológicas considerando un costo menos elevado comparado con las soluciones tecnológicas actuales.

Adicionalmente el dispositivo utilizado en este proyecto permite incorporar más de una funcionalidad en un solo proyecto, siendo sencilla la implementación de innovaciones y reutilización del proyecto para distintos fines. Mencionado esto, el segundo enfoque cuenta como un enfoque alternativo que se da en este proyecto, siendo éste un enfoque publicitario, es decir, el vehículo construido no solo podrá captar imágenes sino reproducir audio de cualquier tipo, de tal manera que se pueda realizar publicidad, ya sea política, comercial, etc. dentro de una zona deseada, controlando el recorrido de la publicidad remotamente.

Por esta razón se explicará a continuación las causas que llevan a la escasez de soluciones tecnológicas en las empresas e incluso el posible rechazo de los empresarios hacia ésta; se mostrará también la importancia de implementar soluciones tecnológicas en este trabajo y automatizar procesos existentes en las empresas.

1.1. Causas

En la actualidad existe cierta dependencia tecnológica. Hoy en día nos podemos dar cuenta que en cierto sentido somos manejados por la tecnología. Cada vez que se crea un nuevo invento tecnológico ahí estamos nosotros, nos dejamos llevar por la tecnología. Son pocos los hogares donde no hay un televisor, un radio, un refrigerador, ventilador, etc. Al hablar de la ciencia y la tecnología debemos meditar en la responsabilidad de quienes la hacen y la ejecutan, resulta inútil comprender la ciencia desvinculada de los sentimientos y valores, de la educación. [2]

Hoy el hombre tiene un gran interés hacia la ciencia como esfera de la actividad humana pues no hay un lugar en el ámbito de la vida de la sociedad donde esta no intervenga, ella es un dominio especial de la actividad humana tanto teórico como práctico. En la "Cumbre de la Tierra" se fortaleció la idea de que el hombre debe seguir interactuando con la naturaleza, pero de forma sostenible es decir tomar de ella lo necesario sin causar daños. [2]

La tecnología nos proporciona felicidad, nos ayuda a resolver problemas, pero muchas veces trae consigo nuevos problemas de difícil solución. Uno de los más grandes y antiguos problemas que ha traído consigo la tecnología es la contaminación ambiental, que hoy en día ha pasado a ser un problema global. Y existen serios problemas que afectan al hombre, y que la ciencia no ha podido hallarle aún solución. [3]

Además que la tecnología actual no es accesible a emprendedores, negociantes, empresas pequeñas y medianas, que fabrican productos, maquinarias, cultivan, y realizan diversas actividades que sustentan en parte la economía del Ecuador; la tecnología que aporta en la automatización de los procesos en las empresas es importada o implementada con un alto costo debido a que esta tecnología utilizada no es desarrollada al interior del país.

Empresas con presupuestos altos son aquellas que pueden contar con tecnología que resuelva problemáticas varias tales como soluciones de vigilancia, soluciones de automatización industrial, biotecnología etc.

1.1.1 Tecnología en la Agricultura Ecuatoriana

Centrándonos en uno de los aspectos de éste estudio, la vigilancia de las zonas de cultivo refleja una gran importancia para la salud del sistema agrícola y el incremento de la población.

El desarrollo de la información y la tecnología en el país no solo son necesarios para el crecimiento del sector industrial o empresarial, sino que deben servir también para mejorar la agricultura y la ganadería. Este fue uno de los planteamientos que se trató en el 'Primer foro internacional de tecnologías de la información y comunicación en el sector agropecuario, que se llevó a cabo en Quito entre el 26 y el 27 de noviembre del 2014. En este encuentro participaron productores, empresarios y expertos en tecnologías aplicadas a la agricultura de otros países. Los expositores comentaron sus experiencias sobre su uso en los

diferentes sistemas de producción. El sector agropecuario que no se tecnifique no va para ningún lado opinan entendidos en el tema. “La gente que trabaja en este sector debe disponer de las mejores herramientas”. Al respecto, Antonio Sánchez, presidente de la Asociación Ecuatoriana de Software, señaló que dentro del área de software ya se está fortaleciendo el sector de tecnologías de la información para que puedan aplicarse en el sector. [4]

La tecnología que se ha implementado en el área agrícola se enfoca en problemas tales como fumigación, biotecnología para mejora de productos, verificaciones de clima, riego, vigilancia, entre otros; teniendo como ejemplo la tecnología de los drones que está de moda en la actualidad, sin embargo tecnología como éstas requieren una inversión de aproximadamente 300000 dólares; capital que no está al alcance de pequeños y medianos agricultores que necesitan también de tecnología para superar los problemas mencionados.

Es notable que la tecnología incorporada en las empresas grandes, aquellas que tienen la capacidad de solventar inversiones tecnológicas, es excesivamente costosa debido a sus funciones a niveles industriales. Lo que no se ha analizado es la situación de las empresas medianas o incluso pequeñas; las funciones que requieren en un sistema de vigilancia son de mucho menor escala y a su vez debería ser de mucho menor precio.

1.2. Efectos

La agricultura comercial de hoy en día se ha vuelto una actividad de alta tecnología, sobre todo con avances tecnológicos importantes en la última década en el manejo agrícola automatizado, genética, nutrición, control de plagas y enfermedades, principalmente. Uno de los elementos que facilitan la transferencia de tecnología alrededor del mundo, es el libre

comercio de estos avances, los cuales se realizan a través de importaciones de maquinaria y equipo especializado, semillas mejoradas, agroquímicos y además por la asesoría especializada de expertos. La adquisición de tecnologías provenientes de diversas partes del mundo genera grandes beneficios para la agricultura en general, pero sobre todo para aquellos países en desarrollo, que enfrentan limitantes para invertir en desarrollo y tecnología agropecuaria. [8]

El sector agropecuario que no se tecnifique no va para ningún lado opinan entendidos en el tema. “La gente que trabaja en este sector debe disponer de las mejores herramientas”. Al respecto, Antonio Sánchez, presidente de la Asociación Ecuatoriana de Software, señaló que dentro del área de software ya se está fortaleciendo el sector de tecnologías de la información para que puedan aplicarse en el sector. [4]

Podría parecer que la tecnología es muy costosa durante la etapa de instrumentación, pero podría ahorrarse dinero en el largo plazo, especialmente cuando se encuentra una solución tecnológica a bajo costo para reemplazar una aplicación arcaica. Muchas soluciones tecnológicas podrían tener costos más elevados que el proceso manual al que reemplazan. [9]

1.3. Soluciones Similares

Actualmente existen varias soluciones similares al proyecto implementado y descrito en este documento. Estas soluciones tecnológicas, involucran varias herramientas, software y hardware que permiten realizar funcionalidades similares.

Existen muchas formas de implementación para éste proyecto, valiéndose de distintos componentes eléctricos y dispositivos embebidos disponibles en el mercado.

Alrededor del mundo se han implementado proyectos similares con distintas herramientas de software y hardware pero ninguna solución

pública combina el uso de Raspberry Pi y Node Red, siendo Node Red la herramienta que permite establecer la funcionalidad del proyecto de una forma muy simple.

Se puede agregar innovación en muchos aspectos a proyectos desarrollados con dispositivos embebidos como Raspberry Pi debido a la cantidad de componentes, módulos y sensores disponibles, fáciles de integrar.

CAPÍTULO 2

2 METODOLOGÍA O SOLUCIÓN TECNOLÓGICA IMPLEMENTADA

Al ser este proyecto, integrador de herramientas en hardware y software, se realizó un análisis de las tecnologías disponibles en el mercado y cuál sería la recomendable para cumplir el objetivo de generar soluciones tecnológicas a bajo costo.

El análisis de los costos y beneficios de la tecnología utilizada en el proyecto obedece los siguientes puntos a tomar en cuenta:

- a) Establecer el propósito que tiene la adquisición tecnológica;
- b) Establecer porqué la adquisición resulta deseable;
- c) Describir el proceso que la tecnología va a reemplazar;
- d) Establecer los problemas o riesgos que podrían presentarse por el uso continuo del proceso actual;
- e) Describir la forma en que la tecnología debería trabajar;
- f) Establecer los beneficios de la tecnología;
- g) Describir los posibles problemas o riesgos que podrían surgir al adoptarla;
- h) Hacer un estimado de todos los costos que implica la introducción de la tecnología, incluyendo el equipo, los programas y la infraestructura necesaria para las comunicaciones, transporte, personal, contratistas externos, mantenimiento y actualizaciones;
- i) Estimar los costos que implicaría un fracaso;
- j) Explicar los costos asociados con el proceso vigente;
- k) Identificar cualquier gasto que podría evadirse con la adopción;
- l) Detallar el plazo para instrumentarla;
- m) Discutir lo sustentable que es e indicar su período de utilidad. [9]

2.1. Tecnologías y herramientas utilizadas en el proyecto

2.1.1. Raspberry Pi, dispositivo embebido

El Raspberry Pi (también RPi o RasPi) es un mini ordenador del tamaño de una tarjeta de crédito que corre bajo Linux y tiene un precio bastante asequible pues cuesta aproximadamente 30 euros, este precio es para la última versión de la RPi que es el modelo B versión 2.0. [10], el cual se observa en la Figura 2.1

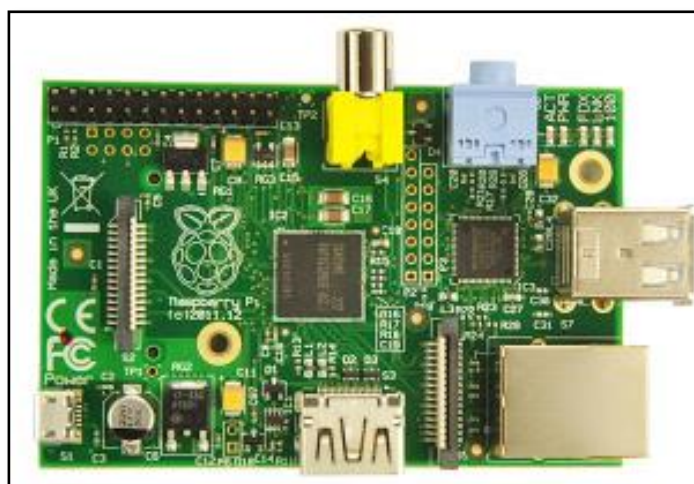


Figura 2.1: Raspberry Pi modelo B revisión 2.0 256 MB.[10]

Es una de las más famosas SBC (*Single Board Computer*) actuales. A pesar de ser muy reciente es la que más unidades ha vendido (no hay cifras oficiales pero seguramente ya han sobrepasado el millón) y ha servido de referencia para otras. La RPi ha sido desarrollada en Reino Unido por la Fundación Raspberry Pi que es una organización benéfica, que tiene como objetivo el de estimular la enseñanza de ciencias de la computación y temas relacionados en las escuelas. [10]

Múltiples usos con Raspberry Pi

Se puede utilizar para muchas de las cosas que hace un PC de escritorio, como navegar por Internet, hojas de cálculo,

procesador de textos y juegos. También permite vídeo de alta definición. Algunas de sus ventajas con respecto a los ordenadores corrientes son su tamaño, bajo coste, bajo consumo y que permite añadir proyectos electrónicos gracias a la cabecera de expansión de pines GPIO lo cual hace que el Raspberry Pi sea el dispositivo perfecto para desarrollar proyectos interesantes y a bajo coste. Alguna desventaja podría ser la imposibilidad de cambiar componentes hardware tal y como podemos hacer en un ordenador de sobremesa. [10]

Especificación técnica Raspberry Pi

Raspberry Pi cuenta con una serie de características técnicas muy modernas al igual que características físicas que hacen de este dispositivo embebido una opción para crear soluciones tecnológicas potentes a pequeña escala. Se pueden observar las características mencionadas en las Figuras 2.2 y 2.3 a continuación.

	Raspberry Pi Model B+	Raspberry Pi 2 Model B
SoC	Broadcom BCM2835	Broadcom BCM2836
CPU	ARM11 ARMv6 700 MHz	ARM11 ARMv7 ARM Cortex-A7 4 núcleos @ 900 MHz
Overclocking	Sí, hasta velocidad Turbo; 1000 MHz ARM, 500 MHz core, 600 MHz SDRAM, 6 overvolt. de forma segura	Sí, hasta arm_freq=1000 sdram_freq=500 core_freq=500 over_voltage=2 de forma segura
GPU	Broadcom VideoCore IV 250 MHz. OpenGL ES 2.0	Broadcom VideoCore IV 250 MHz. OpenGL ES 2.0
RAM	512 MB LPDDR SDRAM 400 MHz	1 GB LPDDR2 SDRAM 450 MHz
USB 2.0	4	4
Salidas de video	HDMI 1.4 @ 1920x1200 píxeles	HDMI 1.4 @ 1920x1200 píxeles

Figura 2.2: Características técnicas de dispositivos Raspberry Pi y Raspberry Pi 2.

Almacenamiento	microSD	microSD
Ethernet	Sí, 10/100 Mbps	Sí, 10/100 Mbps
Tamaño	85,60x56,5 mm	85,60x56,5 mm
Peso	45 g	45 g
Consumo	5v, 600mA	5v, 900mA, aunque depende de la carga de trabajo de los 4 cores
Precio	35 dólares	35 dólares

Figura 2.3: Características físicas de dispositivos Raspberry Pi y Raspberry Pi 2.

Estas características físicas y técnicas de los dispositivos embebidos Raspberry Pi y Raspberry Pi 2 especifican todos los beneficios desde costos hasta potencia que ofrecen.

Módulos de Raspberry Pi para desarrollo del proyecto

Módulo cámara Raspberry Pi

El módulo de la cámara Raspberry Pi se puede utilizar para tomar video de alta definición, así como fotografías. Es fácil de utilizar y tiene mucho que ofrecer a los usuarios avanzados, si se está buscando para ampliar su conocimiento En la figura 2.4 se observa el módulo integrable de la cámara. [11]

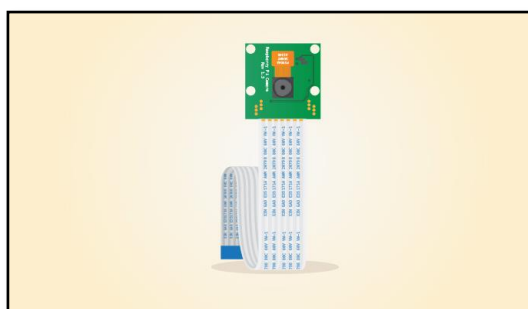


Figura 2.4: Módulo de Cámara web a integrar con Raspberry Pi.

La cámara funciona con todos los modelos de Raspberry Pi 1 y 2. Se puede acceder a través de las API MMAL y V4L. Existen también numerosas bibliotecas de terceros creadas para ello, incluyendo la biblioteca Picamera Python.

El módulo de la cámara es muy popular en aplicaciones de seguridad para el hogar, y en las trampas de la cámara de vida silvestre. [11]

Raspberry Pi USB WIFI dongle (WI - PI)

Wi-Pi es un módulo USB que conecta un Raspberry Pi a una red local Wifi. Wi-Pi es de alto rendimiento, puede transmitir datos de hasta 150 Mbps, mientras que el modelo anterior sólo 54Mbps. En la Figura 2.5 se muestra el módulo integrable para Wifi. [12]



Figura 2.5: Adaptador de Wifi a integrarse en Raspberry Pi

Salida de Audio en Raspberry Pi

El Raspberry Pi tiene dos modos de salida de audio: HDMI y conector para auriculares. Puede cambiar entre estos modos en cualquier momento.

Si su monitor HDMI o TV ha incorporado en los altavoces, el sonido se puede jugar a través del cable HDMI, pero se puede cambiar a un conjunto de auriculares o otros altavoces conectados a la toma de auriculares. Si la pantalla dice tener altavoces, el sonido se emite a través de HDMI de forma predeterminada; si no, se emite a través de la toma de auriculares.

Raspberry SD Card

El Raspberry Pi debería funcionar con cualquier tarjeta compatible con SD, aunque hay algunas consideraciones al respecto:

Tamaño de la tarjeta SD (capacidad). Para la instalación de NOOBS, el tamaño mínimo recomendado tarjeta es 8GB. Para las instalaciones de imagen se recomienda un mínimo de 4 GB.

Clase tarjeta SD. La clase de tarjeta determina la velocidad de escritura sostenida de la tarjeta; una tarjeta de clase 4 será capaz de escribir en 4MB / s, mientras que una clase de 10 debe ser capaz de alcanzar 10 MB / s.

En la Figura 2.6 se muestra la SD Card integrada al dispositivo Raspberry Pi 2

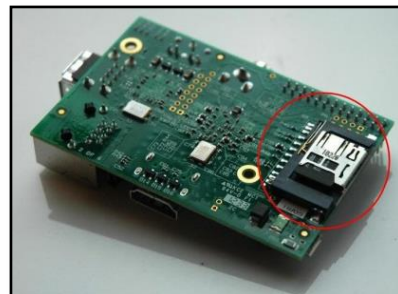


Figura 2.6: SD Card integrada en Raspberry Pi 2.[10]

2.1.2. Software Node JS

Node.js es un entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor, asíncrono, con I/O de datos en una arquitectura orientada a eventos y basado en el motor V8 de Google. [13]

Node.js está basado en el motor V8 de JavaScript de Google. Este motor está diseñado para correr en un navegador y ejecutar el código de JavaScript de una forma extremadamente rápida. Esta arquitectura se muestra en la Figura 2.7 y 2.8 [13]

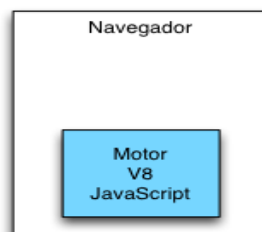


Figura 2.7: Arquitectura de Node JS basado en JavaScript.

La tecnología que está detrás de Node.js permite ejecutar este motor en el lado del servidor abriendo un nuevo abanico de posibilidades en cuanto al mundo de desarrollo se refiere. [13]

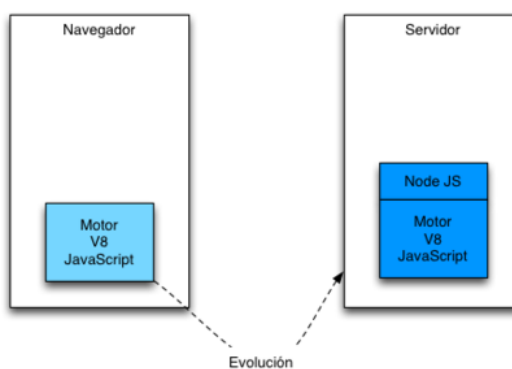


Figura 2.8: Arquitectura y evolución de Node JS basado en JavaScript.

Para ello el entorno de Node.js ha desarrollado un conjunto amplio de librerías que no tienen nada que envidiar a otras plataformas. Por otro lado se han eliminado algunas funcionalidades que en el entorno de servidor no tenían sentido como por ejemplo el uso de Document Object Model. [13]

2.1.3. Software Node Red

Nodo-RED es una herramienta para cablear juntos los dispositivos de hardware, APIs y servicios en línea de maneras nuevas e interesantes. La programación en bloques que se realiza en Node Red se muestra en la Figura 2.9 a continuación [14]

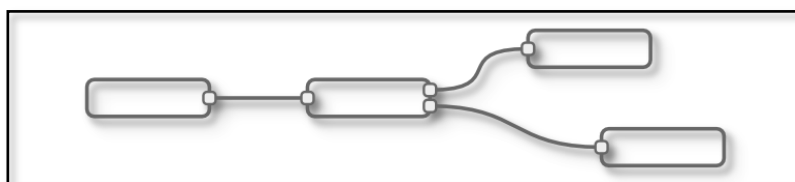


Figura 2.9: Flujo de nodos implementados en Node Red.

Edición de flujo basada en navegador

Node Red ofrece un editor de flujo basada en navegador que hace que sea fácil de cablear juntos los flujos utilizando los nodos de amplio rango en la paleta. Los flujos pueden ser entonces desplegados en el tiempo de ejecución en un solo clic. Funciones de JavaScript se pueden crear en el editor usando el editor de texto enriquecido. Una biblioteca incorporada le permite guardar funciones útiles, plantillas o flujos para su reutilización. En la Figura 2.10 se muestra un

bosquejo del API de Node Red en donde se realiza un flujo a partir de conexiones entre bloques programados. [14]

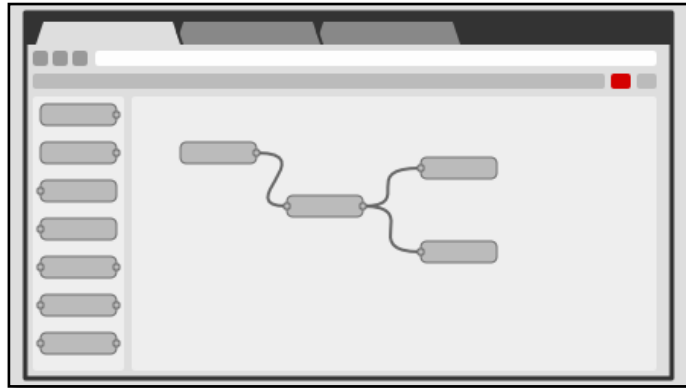


Figura 2.10: Bosquejo de pantalla de la aplicación Node Red, con estructura paleta/pizarrón.[14]

Construido sobre Node.js

El tiempo de ejecución de peso ligero se construye sobre Node.js, aprovechando al máximo de sus eventos, modelo sin bloqueo.

Con más de 120.000 módulos en repositorio de paquetes de Node, es fácil de extender el rango de los nodos de la paleta para agregar nuevas capacidades. [14]

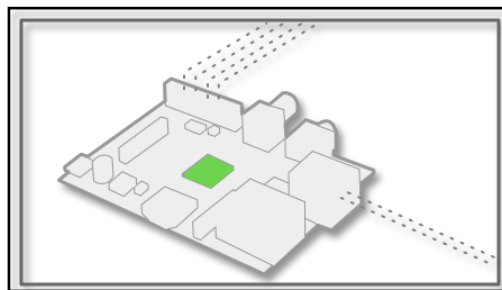


Figura 2.11: Node Red construido sobre Node JS.[14]

Desarrollo Social

Los flujos creados en Nodo-RED se almacenan utilizando JSON que puede ser importado fácilmente y exportado por compartir con los demás. Una biblioteca de flujo en línea le permite compartir sus mejores flujos con el mundo. En la Figura 2.12 se muestra que para cada bloque en un flujo de node red se puede programar un script en distintos lenguajes de programación. [14]

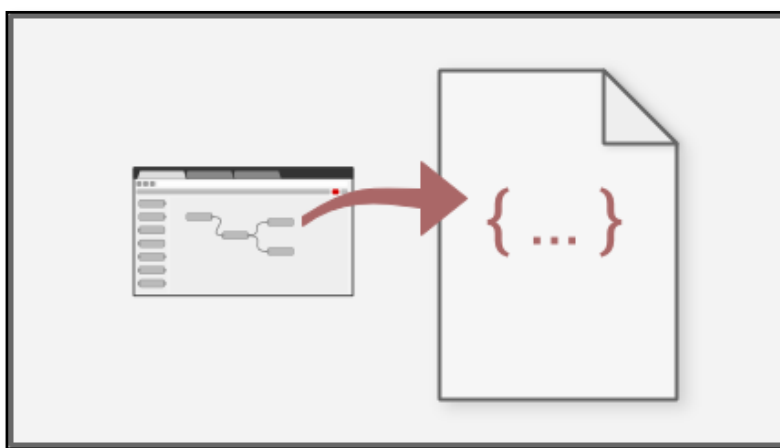


Figura 2.12: Nodos de Node Red ejecutando lenguajes de programación como JavaScript.[14]

Módulos Node Red para desarrollo del proyecto

HTML File

El nodo .html define la forma en que aparece el nodo con el editor. Contiene tres partes distintas, cada uno envuelto en su propia etiqueta <script>.

La definición de nodo principal que se ha registrado en el editor. Esto define las cosas tales como la categoría de la paleta, las propiedades editables (por defecto) y qué icono

para usar. Es dentro de una etiqueta de script JavaScript regular como se muestra en la Figura 2.12. [15]

```
<script type="text/javascript">
  RED.nodes.registerType('node-type', {
    // node definition
  });
</script>
```

Figura 2.12: Script de muestra de una función en JavaScript configurado en nodo de Node Red.

La definición de nodo contiene toda la información sobre el nodo que necesita el editor.

La plantilla de edición para un nodo se describe el contenido de su diálogo de edición. Como se muestra en la figura 2.13 [15]

```
<script type="text/x-red" data-template-name="node-
type">
  <div class="form-row">
    <label form="node-input-name"><i
class="icon-tag"></i> Name </label>
    <input type="text" id="node-input-
name" placeholder="Name">
  </div>
</script>
```

Figura 2.13: Script de muestra de una función en Html configurado en nodo de Node Red.

JavaScript File

Define el comportamiento en tiempo de ejecución del nodo.

Los nodos se definen por una función de constructor que se puede utilizar para crear nuevas instancias del nodo. La

función se ha registrado en el tiempo de ejecución para que pueda ser llamado cuando los nodos del tipo correspondiente se despliegan en un flujo. En la Figura 2.14 se muestra un script de ejemplo para una función en Javascript [16]

```
function SampleNode(config){
    RED.nodes.createNode(this, config);
    // node-specific code goes here
}
RED.nodes.registerType("sample",SampleNode);
```

Figura 2.14: Script de muestra de una función en JavaScript configurado en nodo de Node Red.

Los nodos pueden enviar mensajes a los nodos en dirección descendente del flujo usando la función de envío.

Si el nodo tiene más de una salida, una serie de mensajes pueden ser enviados, a cada una se envía a la salida correspondiente.

Cada vez que se implementa un nuevo flujo, se deberán eliminar los nodos existentes. [16]

Módulo para interactuar con Raspberry Pi

Hay dos formas principales de interactuar con una Raspberry Pi usando Nodo-RED.

Nodos RPI-GPIO (por defecto)

Proporcionada en la paleta para el seguimiento y control de los pines GPIO. Esta es la forma más sencilla y recomendada.

Módulo de cableado-pi (opcional)

Esto proporciona un acceso completo a los pines GPIO, y otros dispositivos, dentro de los nodos de función. Esto da más control y el acceso a otras funciones no en los nodos, pero hay que programar usted mismo.

CAPÍTULO 3

3. RESULTADOS OBTENIDOS Y ANÁLISIS DE LOS RESULTADOS

3.1. Implementación del Hardware del proyecto

La implementación del hardware consiste en el manejo de las entradas y salidas de los puertos disponibles en el Raspberry Pi, además de los módulos integrados a este, como son, el adaptador Wifi, cámara, instalación del sistema operativo a través del SD Card, y adicionalmente se detallarán los circuitos eléctricos independientes utilizados para envío de voltajes hacia la placa interna del vehículo.

3.1.1. Carcasa del vehículo

La carcasa del vehículo corresponde a un prototipo a pequeña escala, con fines de muestra del uso que se puede realizar en el proyecto; cabe recalcar que esta carcasa puede variar según el uso que se le quiera dar al proyecto.

La carcasa del vehículo utilizado en el proyecto, fue reciclada de un vehículo a control remoto por radio frecuencia; ésta carcasa pasó por un proceso de pintado como se muestra en la figura 3.1.



Figura 3.1: Proceso de decoración de carcasa para vehículo controlado remotamente.

El vehículo utilizado tiene un decorado con stickers de las tecnologías utilizadas en la implementación del mismo, entre estas, las mencionadas en el Capítulo 2 del presente documento, además de logos de la Escuela Superior Politécnica del Litoral (ESPOL) y la Facultad de Ingeniería en Eléctrica y Computación (FIEC) como se muestra en la figura 3.2.



Figura 3.2: Diseño final del vehículo controlado remotamente.

3.1.2. Pines del Raspberry Pi

Los pines del Raspberry Pi, los cuales se muestran en la figura 3.3, servirán de intermediarios para enviar voltajes, ya sean altos o bajos hacia el circuito eléctrico que se conecta a la placa controladora de motores del vehículo, este envío de voltajes dependerán de las instrucciones que se envíe desde la interfaz web implementada para uso del usuario que controlará remotamente el vehículo.

No todos los pines serán utilizados, solo aquellos que sirvan para crear funcionalidad en el proyecto.

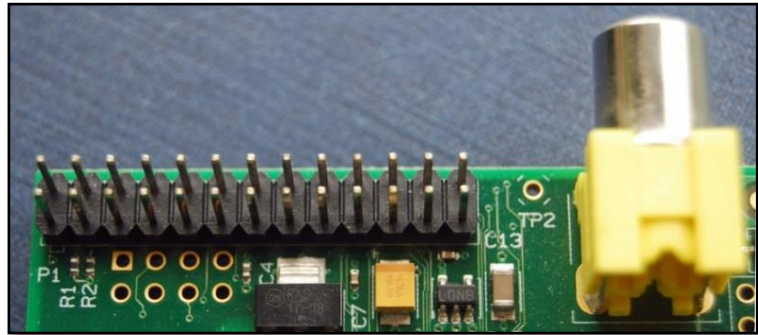


Figura 3.3: Pines de entrada y salida en Raspberry Pi 2.

Los pines del Raspberry Pi utilizados serán el 11 – 12 – 13 – 14 – 15 - 1 - 2 – 6, debido a su específica funcionalidad, las cuales están acordes a lo que se desea lograr en este proyecto.

Cada uno de estos pines será utilizado con un fin específico, ya sean fuentes de alimentación de voltaje o específicamente pines de entrada y salida de información llamados GPIO.

A continuación se detallan los pines utilizados y su utilidad dentro del proyecto; en las figuras 3.4 y 3.5 se especifica el uso de cada pin del Raspberry:

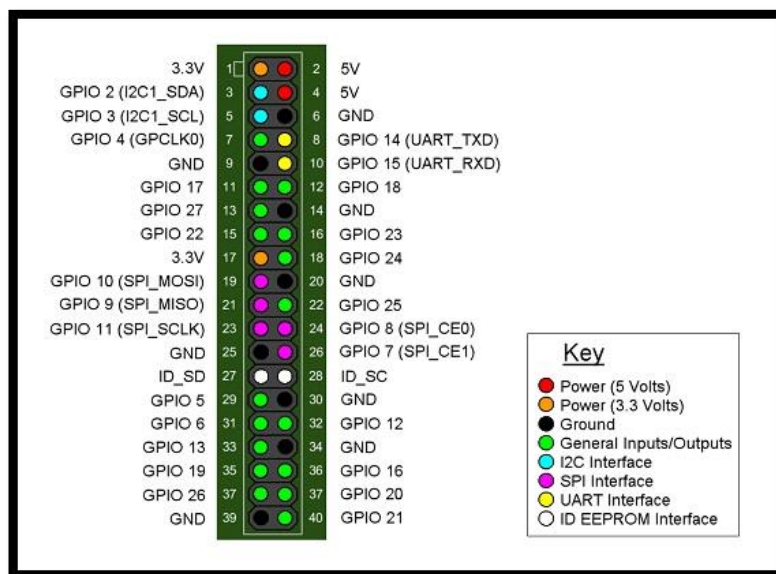


Figura 3.4: Pines de entrada y salida en Raspberry Pi 2.

PIN 2 Utilizado como fuente para alimentación del circuito, este pin provee de un voltaje 5V, para transmitir voltaje y corriente al circuito eléctrico que permite comunicación con placa controladora de motores del vehículo.

PIN 6 Utilizado como referencia a tierra para el circuito eléctrico que permite comunicación con la placa controladora de motores del vehículo.

PIN 1 Utilizado como fuente para la alimentación del circuito, este pin provee de un voltaje 3.3V al circuito eléctrico que permite comunicación con la placa controladora de motores del vehículo.

PIN 11 (GPIO 17) Utilizado para enviar voltaje alto hacia el nodo de la placa controladora de motores que controla el motor derecho

PIN 13 (GPIO 27) Utilizado para enviar voltaje alto hacia el nodo de la placa controladora de motores que controla el motor izquierdo

PIN 15 (GPIO 22) Utilizado para enviar voltaje alto hacia el nodo de la placa controladora de motores que controla el motor delantero

PIN 12 (GPIO 18) Utilizado para enviar voltaje alto hacia el nodo de la placa controladora de motores que controla el motor trasero

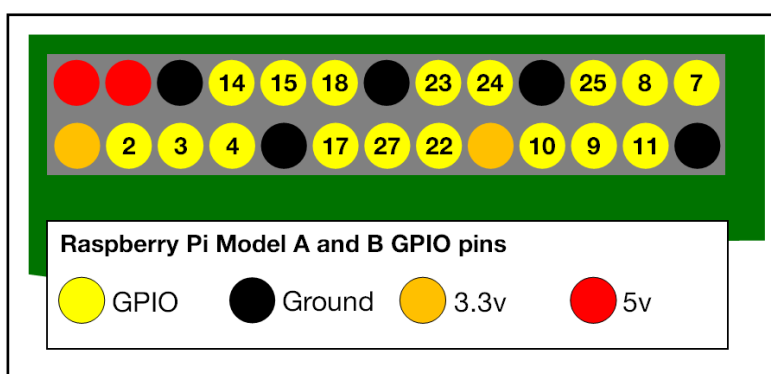


Figura 3.5: Pines de entrada y salida en Raspberry Pi 2.

3.1.3. Tarjeta de conexión entre Raspberry y Placa controladora de motores de vehículo.

Para poder realizar el envío de instrucciones desde los pines del Raspberry Pi, que consisten en voltajes altos y bajos, se diseñó e implementó una placa eléctrica que permite disparar estos eventos cuando la interfaz web controle los movimientos del vehículo.

Para el diseño de la placa se utilizaron las siguientes herramientas:

- Isis:** Es el encargado de realizar el modelo esquemático del circuito, para ello cuenta con una librería de más de 6,000 dispositivos tanto analógicos como digitales.

b) **Ares:** Se encarga de hacer la placa de circuito impreso (PCB) además de que puede posicionar automáticamente los componentes y hacer las pistas. [17]

El diseño de la placa se muestra a continuación:

La figura 3.6 a continuación muestra el esquemático del circuito realizado en Isis:

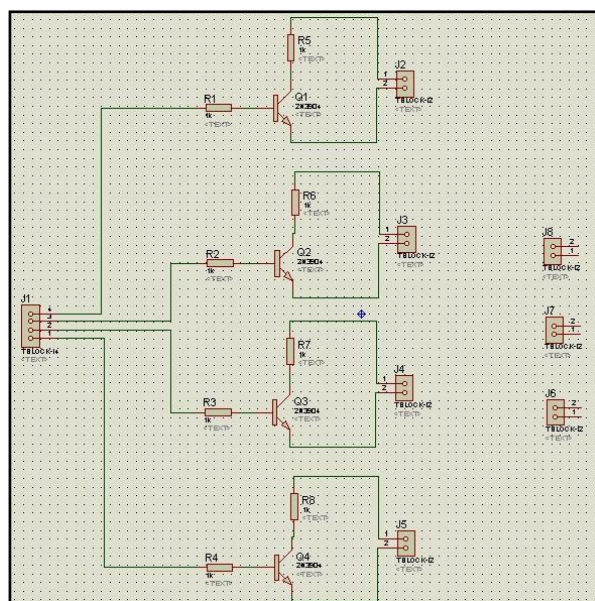


Figura 3.6: Esquemático en Isis de circuito eléctrico para comunicación con placa controladora de motores del vehículo.

El esquemático cuenta con conjunto de 4 circuitos similares los cuales están conformados de un transistor NPN que se utiliza para conmutar una carga de encendido y apagado. Cuando el GPIO es "alto" (lógica 1) la carga se encenderá; cuando es "baja" la carga se apagará. El circuito muestra una barra de alimentación 5V, pero con el transistor 2N3904 se muestra, un suministro de hasta 40V puede ser utilizado. Debe ser adecuado para una corriente de carga de hasta 100 mA. [18]

La figura 3.7 a continuación muestra el esquemático del circuito realizado en Ares:

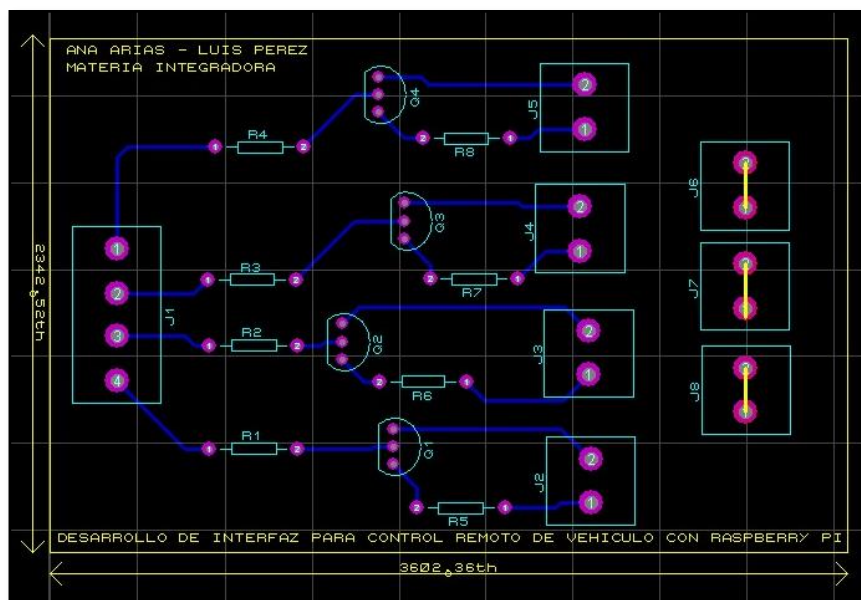


Figura 3.7: Esquemático en Ares de circuito eléctrico para comunicación con placa controladora de motores del vehículo.

En la figura 3.7 se muestra el mismo esquemático de la figura 3.6 realizado en Ares, para la realización de la placa física final que será parte del vehículo controlado remotamente.

Adicionalmente podemos decir de este circuito que para cargas más grandes, tendrá un transistor más grande para cambiar voltajes más altos o corrientes, pero se debe tener en cuenta que necesitará un transistor con una ganancia más alta también. La ganancia es la relación de la corriente en R1 (en este caso, sobre 2,5 mA), a la corriente en la carga: una carga 1A tendrá una ganancia de 400. Los llamados transistores Darlington, tales como el TIP112

(1A, 100V) proporcionan una alta ganancia a altas corrientes. [18]

3.2. Implementación del Software del proyecto

Para la implementación del software del proyecto se utilizaron programas gratuitos que permitieron realizar funciones específicas en el proyecto y permitieron la integración de componentes,

A continuación se listarán los programas utilizados en el proyecto:

3.2.1. Instalación de Sistema Operativo Ubuntu

Ubuntu es una de las versiones más populares del sistema operativo Linux que está disponible de forma gratuita y que se puede ejecutar en casi cualquier computadora. [19]

Los pasos a seguir para la instalación de Ubuntu son los siguientes:

- a) Descarga el archivo ISO de Ubuntu.** Puedes obtener el archivo desde el sitio web de Ubuntu.
- b) Graba el archivo ISO.** Abre el programa de grabación que prefieras. Existen diversos programas gratuitos y pagos que pueden grabar un archivo ISO en un CD o DVD.
- c) Inicia desde el CD.** Una vez que lo hayas grabado, reinicia la computadora y elige iniciar la computadora desde el disco.
- d) Prueba Ubuntu antes de instalarlo.** Luego de haber iniciado la PC desde el CD, podrás elegir la opción de probar Ubuntu sin instalarlo.
- e) Instala Ubuntu.** Para ello, la computadora necesita al menos 4,5 GB de espacio libre, aunque es probable que quieras más espacio para instalar programas y crear archivos.

- f) **Elige qué hacer con tu sistema operativo actual.** Si tienes instalado Windows, puedes elegir alguna de las opciones sobre cómo instalar Ubuntu.
- g) **Determina el tamaño de partición.** Si instalas Ubuntu con Windows, puedes usar el deslizador para ajustar cuánto espacio quieres asignarle a Ubuntu.
- h) **Ingresa la información de inicio de sesión.** Ingresa tu nombre, el nombre de la computadora (el que se mostrará en la red), un usuario y una contraseña.
- i) **Espera a que finalice el proceso de instalación.** La instalación comenzará luego de que hayas ingresado la información de inicio. [19]

En las Figuras 3.8, 3.9 y 3.10 se muestran algunos de los pasos a seguir durante la instalación de Ubuntu.



Figura 3.8: Preparando instalación de Ubuntu.

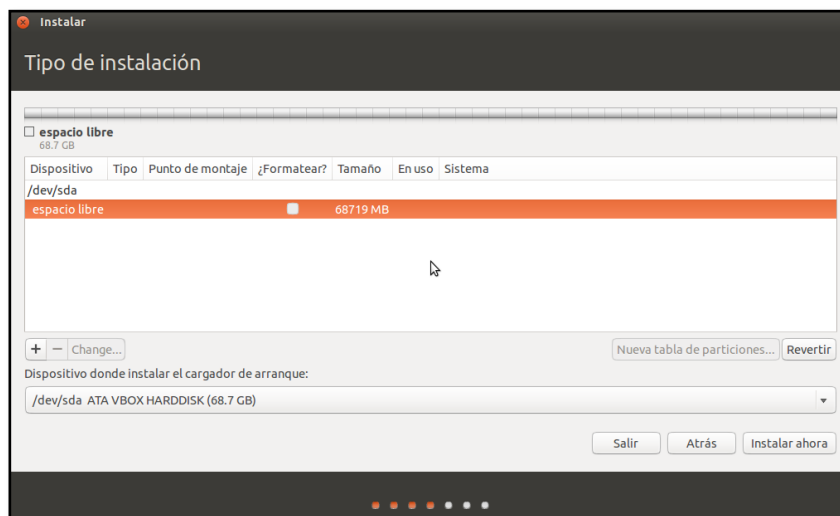


Figura 3.9: Escogiendo tipo de instalación.[19]

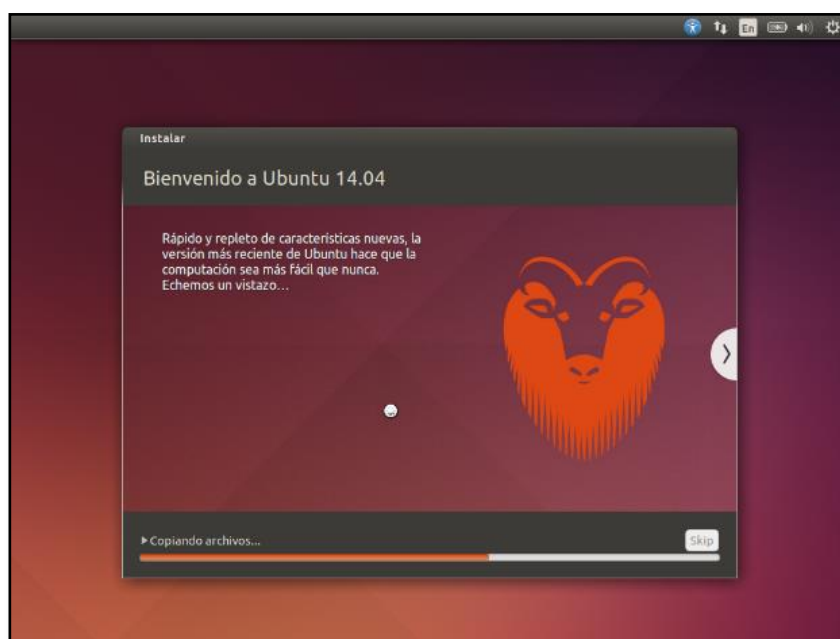


Figura 3.10: Mensaje de Bienvenida a Ubuntu.[19]

Una vez finalice la instalación, se retira el soporte (CD, DVD o USB) y pulsa la tecla “Enter”. El computador se reiniciará y podremos acceder a Ubuntu. [19]

3.2.2. Instalación de programas de utilidad para el proyecto

Instalación de Filezilla en Ubuntu

Puedes descargar e instalar también directamente la aplicación en tan solo un paso utilizando la consola o terminal. Ingresa el siguiente comando y Ubuntu hará todos los procedimientos necesarios para que finalmente quede completamente instalada en el sistema: [20]

```
$ sudo apt-get install filezilla
```

En la Figura 3.11 se puede observar la aplicación de Filezilla instalada y lista para ser utilizada

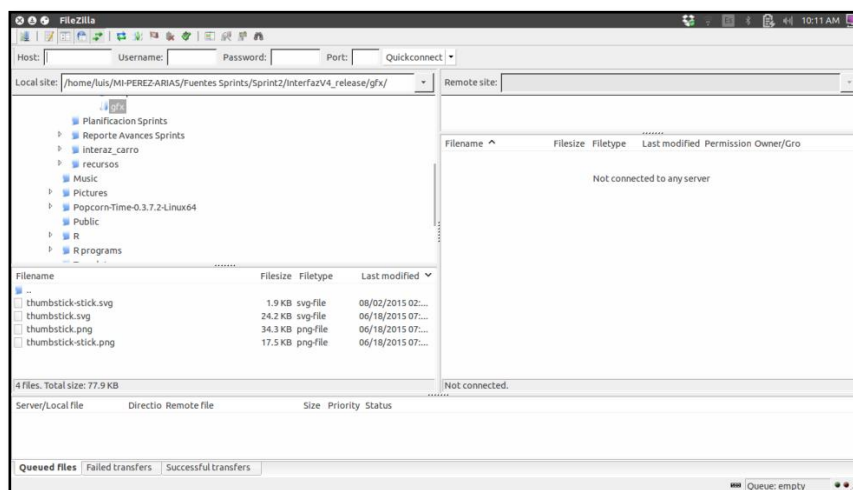


Figura 3.11: Aplicación Filezilla para transferencia de archivos vía ftp.[20]

Instalación de Navegador Firefox en Ubuntu

Lo primero será abrir nuestra terminal de comandos, para ello la buscamos o mejor aún, dejamos pulsadas las teclas **CTRL + ALT + T**.

Lo siguiente que haremos será añadir el repositorio de Firefox 9, para ello escribimos: [21]

```
$ sudo add-apt-repository ppa:mozillateam/Firefox-next
```

A continuación actualizamos la lista de paquetes:

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

Instalamos Firefox 9 en Ubuntu mediante el siguiente comando:

```
$ sudo apt-get install firefox
```

En la Figura 3.12 se muestra el proceso de instalación de Firefox en Ubuntu.

```

juanjo@ubuntu:~$ sudo apt-get install firefox
Leyendo listas de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
El paquete indicado a continuación se instaló de forma automática y ya no es necesario.
 libgladeui-1-11
Utilice «apt-get autoremove» para eliminarlos.
Se instalarán los siguientes paquetes extras:
  firefox-globalmenu
Paquetes sugeridos:
  latex-xft-fonts
Se actualizarán los siguientes paquetes:
  firefox firefox-globalmenu
2 actualizados, 0 se instalarán, 0 para eliminar y 5 no actualizados.
Necesito descargar 18,2 MB de archivos.
Se utilizarán 1716 kB de espacio de disco adicional después de esta operación.
¿Desea continuar [S/n]? s
Des:1 http://ppa.launchpad.net/mozillateam/firefox-next/ubuntu/ oneirc/main fir
efox-globalmenu i386 9.0-b6+build1-0ubuntu0.11.10.1-mfn1 [138 kB]
Des:2 http://ppa.launchpad.net/mozillateam/firefox-next/ubuntu/ oneirc/main fir
efox i386 9.0-b6+build1-0ubuntu0.11.10.1-mfn1 [18,1 MB]
Descargados 18,2 MB en 29seg. (614 kB/s)
(Leyendo la base de datos ... 214065 ficheros o directorios instalados actualmen
te.)
Preparando para reemplazar firefox-globalmenu 8.0+build1-0ubuntu0.11.10.3 (usand
o ../firefox-globalmenu_9.0-b6+build1-0ubuntu0.11.10.1-mfn1_i386.deb) ...

```

Figura 3.12: Proceso de instalación Firefox en Ubuntu.

Instalación de SSH

Un servidor SSH permite acceder de manera remota al equipo. De este modo, se puede acceder a la consola (similar a Telnet, pero mucho más seguro) y transferir archivos (similar a FTP, pero mucho más seguro). Para realizar la instalación de SSH se ejecuta el comando: [22]

```
$ sudo aptitude install ssh
```

```
$ ssh milogin@direccionip
```

3.2.3. Configuración de SD Card y módulos a ser utilizados.

Para la instalación del sistema operativo necesario para operar en Raspberry Pi, se descargó una “Imagen” de un sitio web llamado thethingbox.io, sitio dedicado a implementaciones con dispositivos embebidos tales como Raspberry Pi, Raspberry Pi 2, BeagleBone Black, Arduino; la imagen descargada tendría módulos y configuraciones programadas por defecto.

Herramientas necesarias

Para preparar el Raspberry Pi, se necesitan algunas herramientas:

- a) Una SD Card. En el Raspberry Pi, todo el software está en una tarjeta SD. Así, el primer paso para acceder al interior del dispositivo será escribir la imagen descargada desde ThingBox en la tarjeta SD. Advertencia: la tarjeta SD será totalmente sobre escrita, por lo que esta tarjeta SD se utilizará sólo para el Raspberry Pi.

- b) Un equipo con Windows o Macintosh (con un puerto SD para acceder a la tarjeta SD). Para este proyecto se utilizó un equipo con el sistema operativo Ubuntu.
- c) La imagen descargada desde ThingBox.
- d) Una aplicación particular de la computadora para escribir la tarjeta SD. [23]

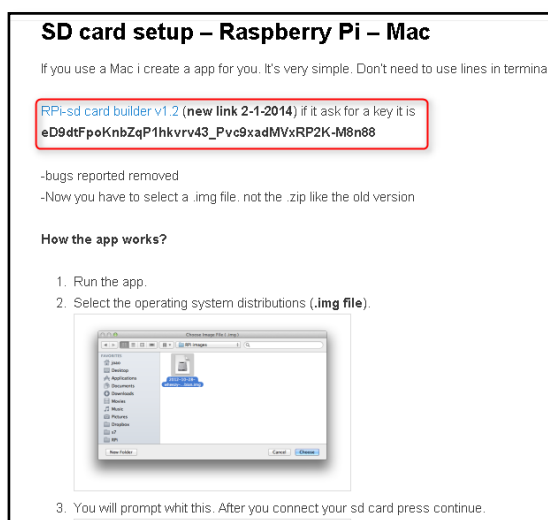


Figura 3.14: Proceso de descarga de Sistema Operativo de Raspberry Pi en SD Card.

Pasos para la instalación

- a) Descargar imagen de ThingBox versión 1.7
- b) Extraer la imagen de ThingBox.
- c) Ejecutar la aplicación Disks para Ubuntu que permite grabar archivos en dispositivos de almacenamiento.
- d) Seleccionar el disco sobre el cuál se quiere grabar la imagen.
- e) Dar clic en el botón configuración, y seleccionar opción Create Image.

- f) Seleccione la imagen ThingBox.
- g) Presionar Start. [23]

En la figura 3.15 se muestra el proceso para quemar el sistema operativo en la SD Card integrada al dispositivo embebido Raspberry Pi, de esta manera se puede acceder a las funcionalidades del dispositivo.

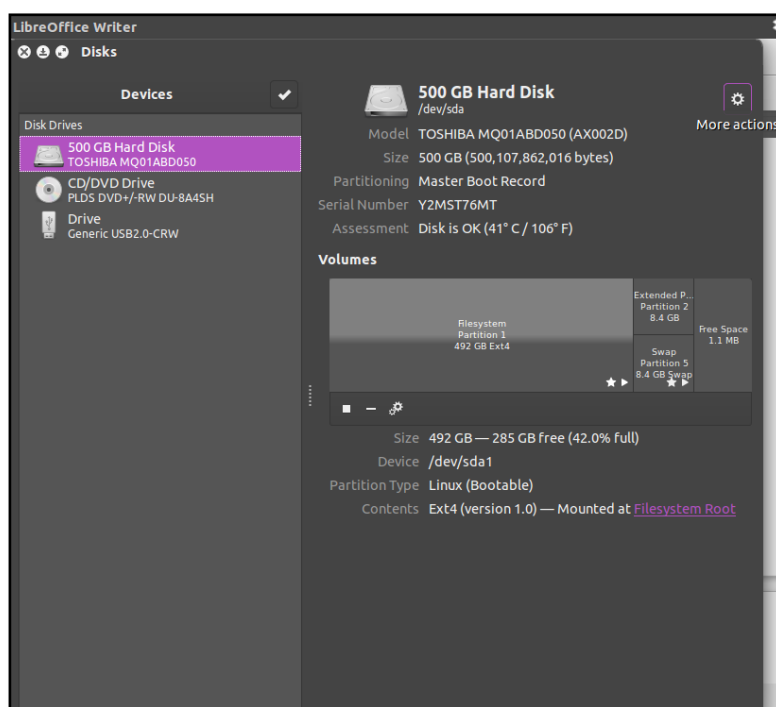


Figura 3.15: Proceso de quemado de Sistema Operativo en SD Card de Raspberry Pi.

Comenzando a utilizar Raspberry Pi

Inserte la tarjeta SD en el Raspberry Pi, conectarlo a la red local y enciéndalo. Espere unos pocos minutos para que arranque el sistema operativo a partir de la imagen instalada en los pasos previos. [24]

Hay dos posibilidades para acceder a la imagen ThingBox:

- a) Utilizar otro ordenador (Windows o Macintosh) para acceder remotamente.
- b) Conectar un teclado, un mouse y una pantalla en el Raspberry Pi y utilizar el navegador integrado.

Para el proyecto se utilizó la opción 2 para poder acceder al sistema operativo del Raspberry y comenzar con las configuraciones necesarias para integrar módulos del proyecto [24]. La instalación finalizada del sistema operativo en las computadoras utilizadas en el proyecto se muestra en las Figuras 3.16 y 3.17.

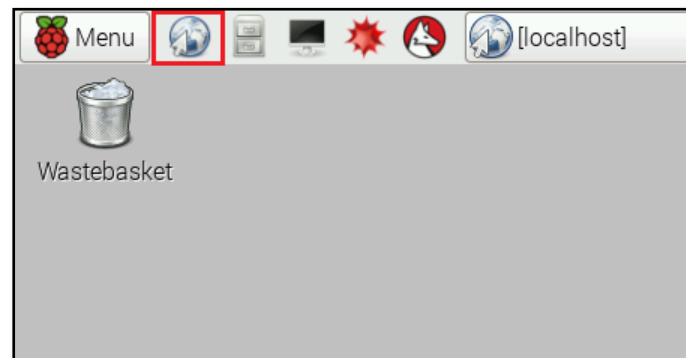


Figura 3.16: Interfaz de usuario para el sistema interno de Raspberry Pi.

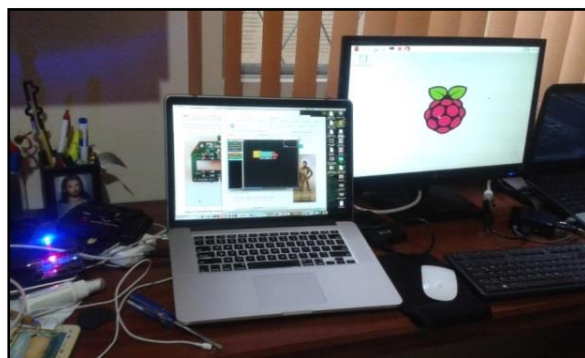


Figura 3.17: Accediendo a interfaz de usuario para el sistema interno de Raspberry Pi desde monitor.

Integrando adaptador WIFI a Raspberry Pi

El primer paso será conectar el Adaptador de Wifi especificado en el Capítulo 2 para poder realizar una conexión inalámbrica desde Raspberry Pi.

Se proporciona una interfaz gráfica de usuario para la configuración de las conexiones Wifi en la interfaz de Raspberry Pi actual.

La conexión inalámbrica vía Wifi se puede hacer a través del icono de red en el extremo derecho de la barra de menú. Si un adaptador Wifi está conectado, dar clic izquierdo en este icono, se abrirá una lista de redes Wifi disponibles como se muestra en la Figura 3.19. [25]

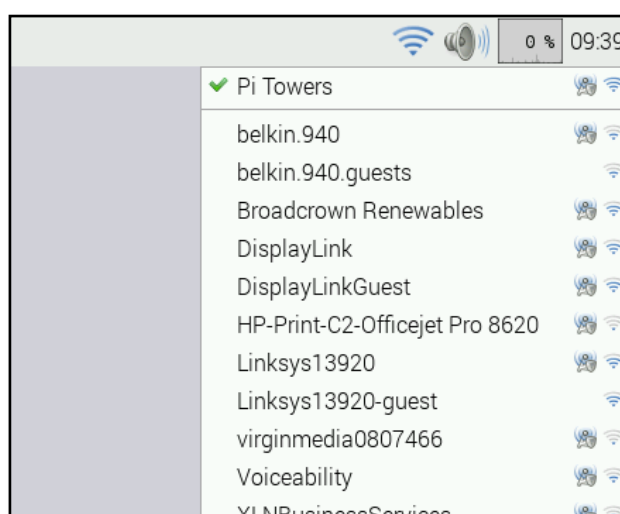


Figura 3.19: Detectando redes disponibles gracias a adaptador Wifi de Raspberry Pi.

Los iconos de la derecha muestran si una red está asegurada o no, y su fuerza de la señal. Haga clic en la red a la que desea conectarse; si está asegurada, un cuadro de diálogo se muestra que le pide que introduzca la clave de red.

Introduzca la clave y presione Aceptar, a continuación, esperar un par de segundos. El icono de red parpadea brevemente para mostrar que se está haciendo una conexión; una vez que esté listo, el icono deja de parpadear y muestra la intensidad de la señal. [25]

Integrando módulo Cámara a Raspberry Pi

El cable flexible se inserta en el conector situado entre los puertos Ethernet y HDMI, con los conectores de plata que enfrenta el puerto HDMI. El conector del cable flex debe abrirse tirando de las pestañas de la parte superior del conector hacia arriba y luego hacia el puerto Ethernet. El cable flexible debe ser insertado firmemente en el conector, con cuidado de no doblar el flex en un ángulo demasiado agudo. La parte superior del conector debe ser empujado hacia el conector HDMI y hacia abajo, mientras que el cable flexible se mantiene en su lugar como se muestra en la Figura 3.20.

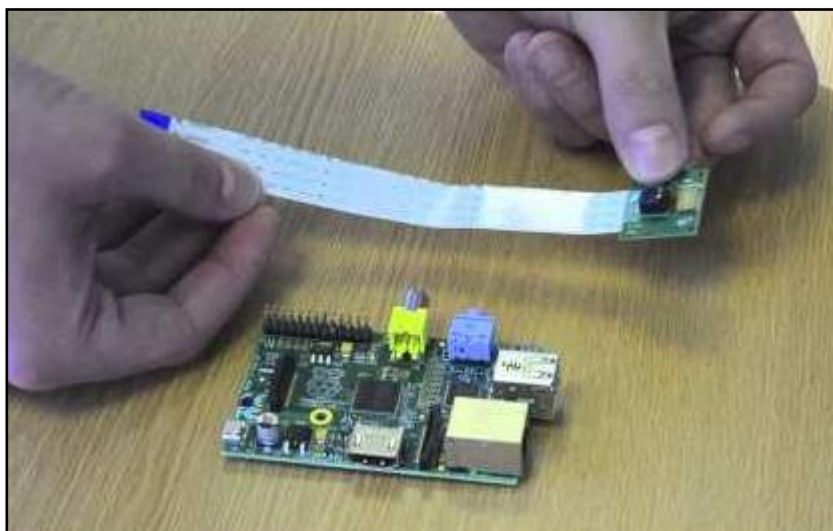


Figura 3.20: Módulo de cámara web a ser integrada en Raspberry Pi.

Habilitar configuración de Cámara en Raspberry Pi

Abrir la herramienta raspi-config desde el terminal con el siguiente comando:

```
$ sudo raspi-config
```

Seleccionar Enable camera y presionar Enter, luego dirigirse a Finish; se reiniciará el Raspberry Pi.

Integración de cámara web en el proyecto

El siguiente comando instala las tres bibliotecas que utiliza MJPG-Streamer:

```
$ sudo apt-get install libjpeg8-dev imagemagick libv41-dev
```

El archivo de cabecera videodev.h que necesita MJPG-Streamer ha sido reemplazado con un videodev2.h. Para hacer MJPG-Streamer óptimo tienes que crear un enlace simbólico:

```
$ sudo ln -s /usr/include/Linux/videodev2.h  
/usr/include/linux/videodev.h
```

El código fuente para MJPG-Streamer está disponible en sourceforge.net, pero es difícil de encontrar el enlace de descarga directa:

```
$ wget http://sourceforge.net/code-snapshots/svn/m/mj/mjpg-  
streamer/code/mjpg-streamer-code-182.zip
```

La descarga de código fuente es un archivo comprimido zip. Ponga el archivo en un directorio personal (o una carpeta temporal) y ejecutar el siguiente para extraer los archivos:

```
$ unzip mjpg-streamer-code-182.zip
```

MJPEG-Streamer viene con varios plugins, pero sólo se necesitan un par de ellos para transmitir vídeo. El comando a continuación sólo construye lo que se necesita:

```
$ cd mjpg-streamer-code-182/mjpg-streamer
$make mjpg_streamer input_file.so output_http.so
```

Para instalar MJPG-Streamer. Los siguientes comandos copian todos los archivos necesarios en los directorios del sistema:

```
$ sudo cp mjpg_streamer /usr/local/bin
$ sudo cp output_http.so input_file.so
/usr/local/lib/
```

Una vez instalado lo necesario, inicializamos la cámara con el siguiente comando:

```
$ mkdir /tmp/stream
$ raspistill -nopreview -w 640 -h 480 -q 5 -o
/tmp/stream/pic.jpg -tl 100 -t 999999 -th 0:0:0 &
```

A continuación inicializar MJPG-Streamer con el siguiente comando:

```
LD_LIBRARY_PATH=/usr/local/lib mjpg_streamer
-i "input_file.so -f /tmp/stream -n pic.jpg" -o
"output_h ttp.so -w /usr/local/www"
```

3.2.4. Instalación de Node JS y Node Red

Instalación de Node JS

Debido a que el Raspberry Pi 2 utiliza un procesador diferente (Arm v7) en comparación con el original de Pi (Arm v6) el método de instalación de Node.js es ligeramente diferente.

Nota: Las versiones de io.js no son compatibles actualmente.

Para instalar Node.js en Pi 2 y otros consejos basados en el procesador Arm7, ejecute los siguientes comandos:

```
curl -sL https://deb.nodesource.com/setup_0.12 | sudo bash -  
sudo apt-get install -y build-essential Python-dev Python-  
rpi.gpio node js
```

Esto también instala algunas dependencias adicionales. [27]

Instalación de Node Red

Para Node Red 0.10.4 o posterior, la forma más fácil de instalar el Nodo Red es utilizar el gestor de paquetes de node, la npm:

```
$ sudo npm install -g --unsafe-perm node-red
```

Nota: Durante la instalación, algunos errores pueden ser reportados por el comando node-gyp. Estos suelen ser los errores no fatales y están relacionados con las dependencias opcionales que requieren un compilador para construirlos. Node Red funcionará sin estas dependencias opcionales, pero

puede encontrar módulos de nodos adicionales que requieren la capacidad de compilar código nativo.

Una vez instalado, usted debe verificar qué versión de las bibliotecas de Python RPi.GPIO se han instalado.

Node Red incluye un nrgpio script específico Raspberry Pi para interactuar con los pines GPIO hardware. Este script también se puede utilizar para comprobar qué versión de la biblioteca subyacente está instalada:

```
<node-red-install-directory>/nodes/core/hardware/nrgpio ver0
```

Una vez que Node Red está ejecutando, ejecutar un navegador en la máquina donde se está ejecutando el Node Red. Una forma de encontrar la dirección IP del Raspberry Pi es utilizar el comando:

```
hostname -I
```

Luego navegar a **http://{Dirección ip devuelta}:1880/**, tal como se muestra en la Figura 3.21; de esta manera se accede al Api de Node Red que permite trabajar con bloques programables.

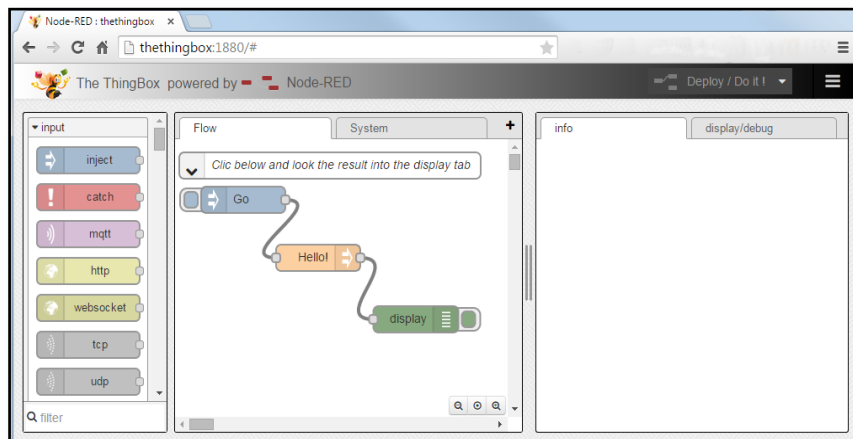


Figura 3.21: Aplicación Node Red con flujo de prueba.

La parte izquierda se llama la paleta. Contiene nodos. Cada nodo da una funcionalidad.

La parte central es el espacio de trabajo, con pestañas para organizar el trabajo. Los nodos pueden ser arrastrados hacia el espacio de trabajo y ser vinculados unos a otros para hacer un flujo.

La parte derecha es la barra lateral. Contiene pestañas que dan informaciones: la ficha de información muestra una documentación cuando se selecciona un nodo. La ficha contiene pantalla de salida o errores del flujo.

Si no aparece la barra lateral, abrirlo con: Ctrl-espacio, o a través del menú desplegable. [27]

3.2.5. Implementación de Interfaz web, audio y cámara

La interfaz tiene un diseño sencillo y concreto para las funcionalidades principales que realiza el proyecto; cuenta con un diseño responsive que se adapta a distintos tamaños de

pantalla, ya que esta interfaz fue diseñada específicamente para dispositivos móviles.

Las herramientas utilizadas para la implementación de la interfaz web son las siguientes:

Html: Sus siglas son Hyper Text Markup Language (lenguaje de marcas de hipertexto). Es un estándar que sirve de referencia para la elaboración de páginas web, define una estructura básica y un código para la definición de contenido de una página web, como texto, imágenes, videos, entre otros. [28]

Css3: Hoja de estilo en cascada es un lenguaje usado para definir y crear la presentación de un documento estructurado escrito en HTML o XML. [29]

JavaScript: Es un lenguaje de programación interpretado. Se define como orientado a objetos, basado en prototipos, imperativo y dinámico. Se utiliza como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas. [30]

Bootstrap: Es el framework en Html, Css y JavaScript más popular para el desarrollo responsive de sitios web. [31]

La implementación de esta interfaz se realizó en su capa visual con HTML y CSS3, y para la interacción con el usuario se implementaron funciones con JavaScript, y finalmente se utilizaron librerías JavaScript de Bootstrap para lograr un diseño responsive; siendo estos los únicos lenguajes utilizados para la implementación de la misma.

En la Figura 3.22 se muestra la interfaz final utilizada para el control remoto de vehículo, mostrando las imágenes del recorrido y habilitando un botón en la parte superior para activación y desactivación del audio.

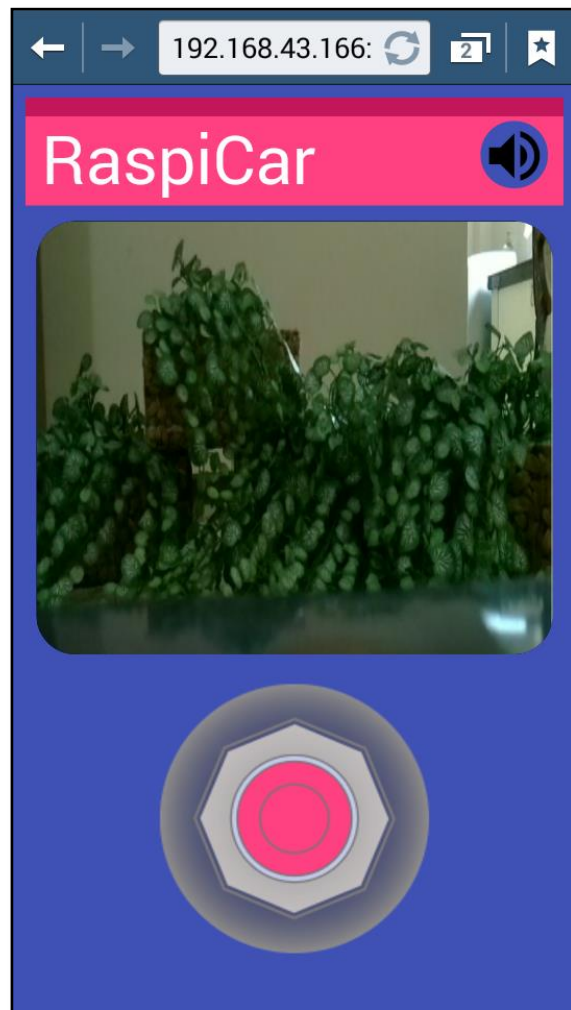


Figura 3.22: Interfaz web responsive para control remoto de vehículo móvil.

3.2.6. Implementación de flujo completo en Node Red

A continuación se explicará la especificación de los nodos utilizados en el proyecto, y los flujos independientes e integrados realizados con la interacción de estos nodos para lograr las funcionalidades del proyecto.

Especificación de nodos utilizados en flujo

Delay: Presenta un retraso en unos límites de caudal o mensajes. Retraso por defecto es de 5 segundos y límite de velocidad de 1 msg / segundo, pero ambos se puede configurar [14]. La configuración de este nodo se muestra en la Figura 3.23.

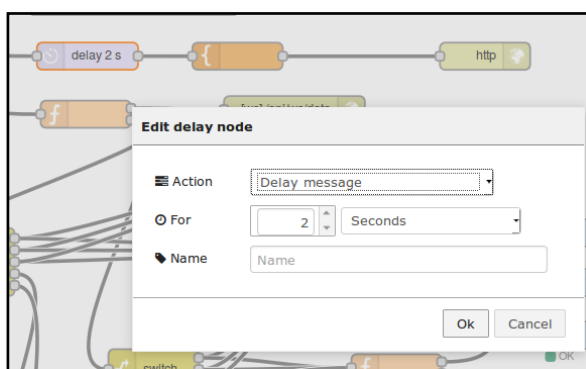


Figura 3.23: Configuración del nodo Delay en el flujo del proyecto.

Http: Proporciona un nodo de entrada para las solicitudes HTTP, permitiendo la creación de servicios web simples. El mensaje resultante tiene las siguientes propiedades:
 msg.req: http petición.
 msg.res: http respuesta. [14]
 La configuración de este nodo se muestra en la Figura 3.24.

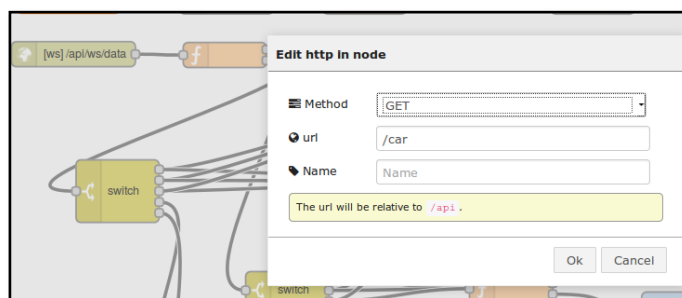


Figura 3.24: Configuración del nodo Http en el flujo del proyecto.

Http Response: Envía las respuestas a las peticiones HTTP recibidas de un nodo de entrada HTTP.

La respuesta se puede personalizar usando las siguientes propiedades del mensaje:

payload, se envía como el cuerpo de la respuesta

statusCode, si se establece, se utiliza como el código de estado de respuesta (por defecto: 200)

cabeceras, si se establece, debe ser un objeto que contiene pares campo / valor que se añade como cabeceras de respuesta. [14]

La configuración de este nodo se muestra en la Figura 3.25.

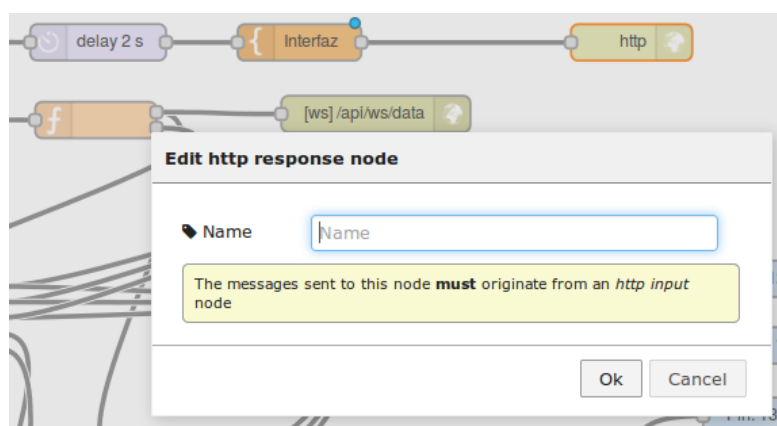


Figura 3.25: Configuración del nodo Http response en el flujo del proyecto.

Function: Un bloque de funciones donde se puede escribir código para hacer las cosas más interesantes. El mensaje se pasa como un objeto de JavaScript llamada *msg*.

Por convención tendrá una propiedad *msg.payload* que contiene el cuerpo del mensaje. [14]

La configuración de este nodo se muestra en la Figura 3.26.

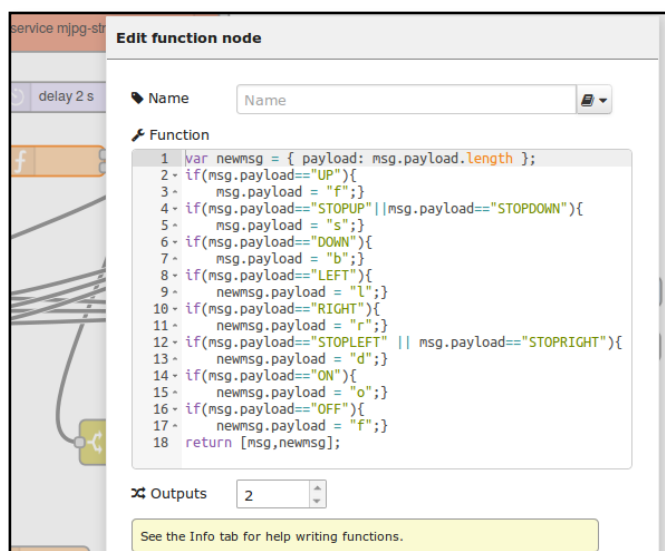


Figura 3.26: Configuración del nodo Function en el flujo del proyecto.

Websocket In: Nodo de entrada WebSocket.

Por defecto, los datos recibidos del WebSocket estarán en msg.payload. El zócalo se puede configurar para esperar una cadena JSON formada, en cuyo caso se deberá analizar el JSON y enviar en el objeto resultante como mensaje completo. [14]

La configuración de este nodo se muestra en la Figura 3.27.

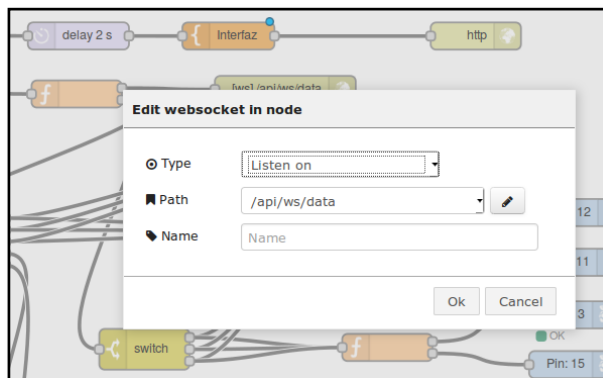


Figura 3.27: Configuración del nodo WebSocket en el flujo del proyecto.

WebSocket Out: Nodo de salida WebSocket.

Por defecto, msg.payload se envía a través de la WebSocket. El socket puede ser configurado para codificar todo el objeto de mensaje como una cadena JSON.

Si el mensaje que llega en este nodo inició en un nodo WebSocket de entrada, el mensaje será enviado de vuelta al cliente que desencadenó el flujo. [14]

La configuración de este nodo se muestra en la Figura 3.28.

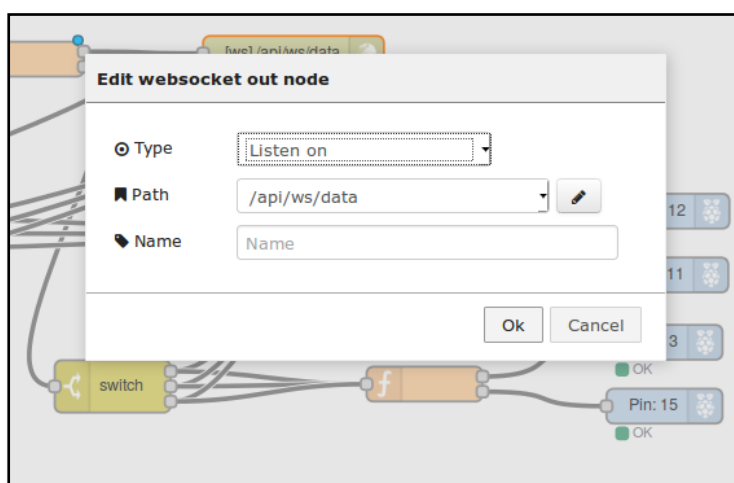


Figura 3.28: Configuración del nodo WebSocket Out en el flujo del proyecto.

Exec: Ejecuta llamada a un comando del sistema. Proporciona 3 salidas: stdout, stderr y código de retorno. Por defecto usa `exec()`, que llama al comando, bloquea a la espera de la finalización, y luego devuelve el resultado completo de una sola vez, junto con los errores. [14]
La configuración de este nodo se muestra en la Figura 3.29.

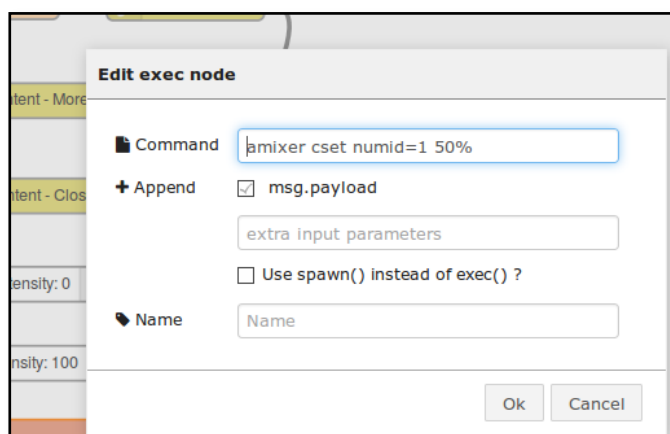


Figura 3.29: Configuración del nodo Exec en el flujo del proyecto.

Gpio: Nodo de salidas de Raspberry Pi. Espera un `msg.payload`, ya sea con un 0 o 1 (o verdadero o falso). Requiere el comando GPIO para trabajar. Establecerá el pin físico seleccionado de alta o baja en función del valor pasado. El valor inicial del pin en el momento del despliegue también se puede establecer en 0 o 1. Al utilizar el modo PWM - espera un valor de entrada de un número 0-100. [14]

La configuración de este nodo se muestra en la Figura 3.30.

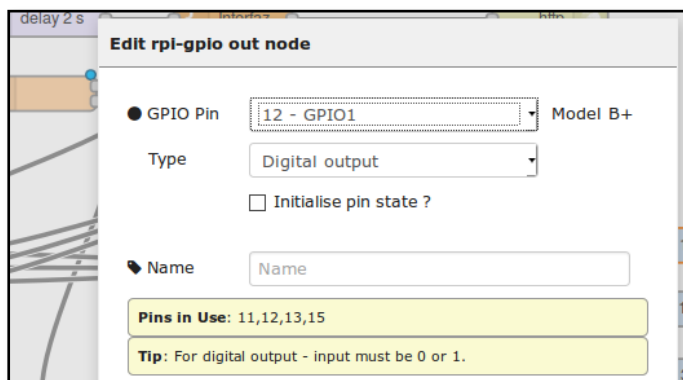


Figura 3.30: Configuración del nodo Gpio en el flujo del proyecto.

Switch: Un nodo de función simple para enrutar mensajes basados en sus propiedades.

Cuando llega un mensaje, la propiedad seleccionada se evalúa cada una de las reglas definidas. El mensaje se envía entonces a la salida de todas las reglas que pasan. [14]

La configuración de este nodo se muestra en la Figura 3.31.

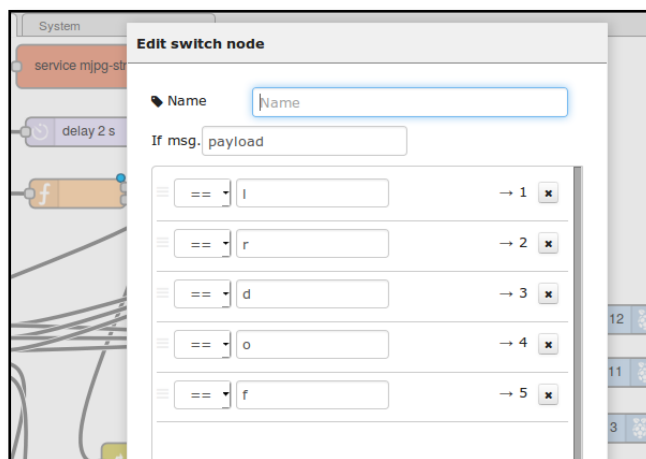


Figura 3.31: Configuración del nodo Switch en el flujo del proyecto.

Audio: Reproducir el archivo mp3 que se encuentra incrustado en el flujo. La primera salida es para cuando se reproduce el sonido. La segunda es para cuando se terminó el sonido.

El nodo toma la URL mp3 en `msg.sound || msg.value || msg.payload` que anula cualquier valor en forma de nodo editable, si está definido.

Entiende las instrucciones "Open", "Close", "More", "Less", "Intensity". [14]

La configuración de este nodo se muestra en la Figura 3.32.

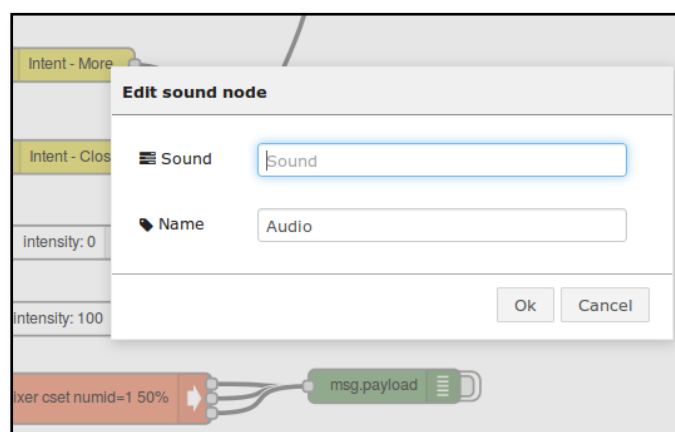


Figura 3.32: Configuración del nodo Sound en el flujo del proyecto.

Intensity: Este nodo le permite seleccionar un valor en un control deslizante. El valor seleccionado se encuentra en el atributo de intensidad. También puede optar por los valores máximos y mínimos en la pestaña de configuración avanzada. Se puede cambiar el atributo utilizado en la pestaña de configuración avanzada. [14]

La configuración de este nodo se muestra en la Figura 3.33.

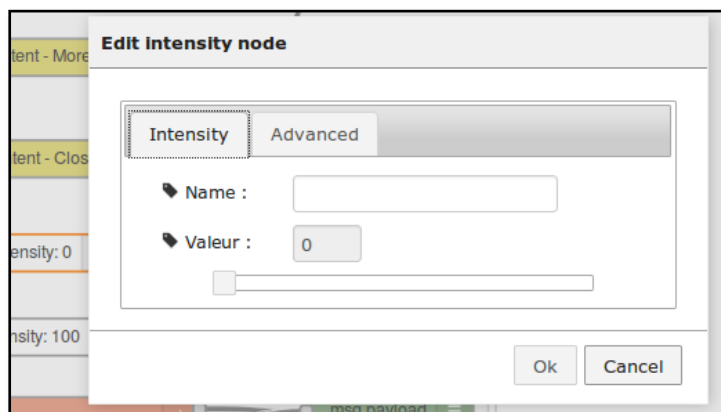


Figura 3.33: Configuración del nodo Intensity en el flujo del proyecto.

Flujo para obtener imágenes captadas por cámara

Este flujo describe la funcionalidad para lograr el streaming de imágenes del módulo cámara; en el flujo encontramos los nodos Http In, Http Out, Delay, Function y el uso de la librería MJPG-Streaming descritos en el punto “Especificación de nodos utilizados en flujo”.

En la Figura 3.34 se muestra el flujo con bloques de Node Red para esta funcionalidad.

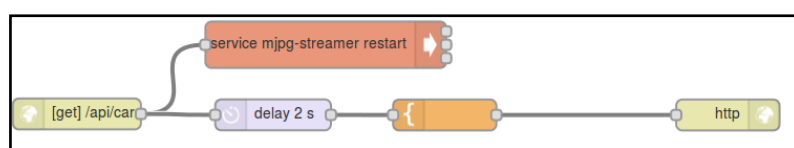


Figura 3.34: Flujo en Node Red para Streaming de cámara.

Flujo para recibo y envío de mensajes vía WebSocket

Este flujo describe la funcionalidad para establecer el canal de comunicación entre Raspberry Pi con la interfaz web mediante

Websocket a través de la red inalámbrica compartida por ambos dispositivos, así como el recibo y envío de mensajes por medio de este canal; en el flujo encontramos los nodos Websocket In, Websocket Out y function descritos en el punto “Especificación de nodos utilizados en flujo”.

En la Figura 3.35 se muestra el flujo con bloques de Node Red para esta funcionalidad.

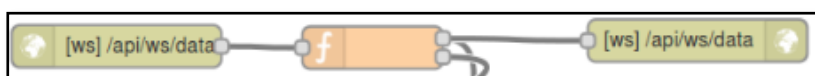


Figura 3.35: Flujo en Node Red para envío de mensajes vía WebSocket.

Flujo para validación de instrucciones de movimientos recibidos por la interfaz web y envío a pines de Raspberry Pi

Este flujo describe la funcionalidad para validación de instrucciones de movimientos recibidos por la interfaz web y se encarga de enviar instrucciones de voltajes altos o bajos a los pines del Raspberry Pi; en este flujo encontramos los nodos Switch, function, Gpio.

En la Figura 3.36 se muestra el flujo con bloques de Node Red para esta funcionalidad.

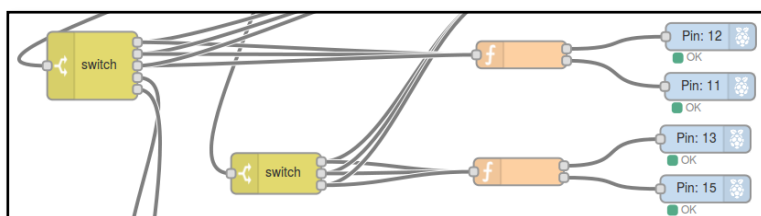


Figura 3.36: Flujo en Node Red para validación de instrucciones hacia pines de Raspberry Pi.

Flujo para activación y desactivación de audio

Este flujo describe la funcionalidad para la activación, reproducción y desactivación del audio a partir de instrucciones enviadas por la interfaz web, adicionalmente permite aumentar la intensidad del audio directamente al Raspberry Pi a través de la ejecución de un comando; en el flujo encontramos los nodos de Audio, Intensity, Go, así como las opciones con las que trabaja el nodo Audio (Open, Close, More, Less).

En la Figura 3.35 se muestra el flujo con bloques de Node Red para esta funcionalidad.

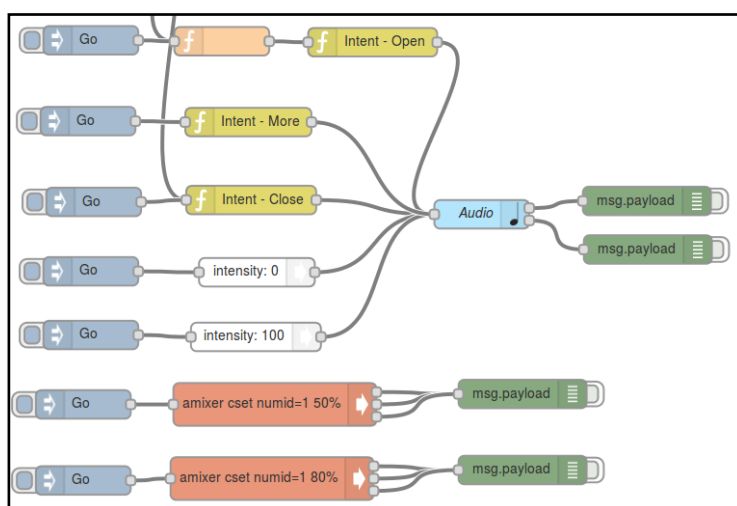


Figura 3.37: Flujo en Node Red para reproducción de sonido.

Flujo completo para funcionalidades del proyecto implementado en Node Red

La Figura 3.38 muestra el flujo completo que integra las diversas funciones que logran los objetivos del proyecto, los cuales son: comunicación con interfaz web móvil para control

de motores de un vehículo, captura de imágenes tomadas por el vehículo y reproducción de audio durante su recorrido.

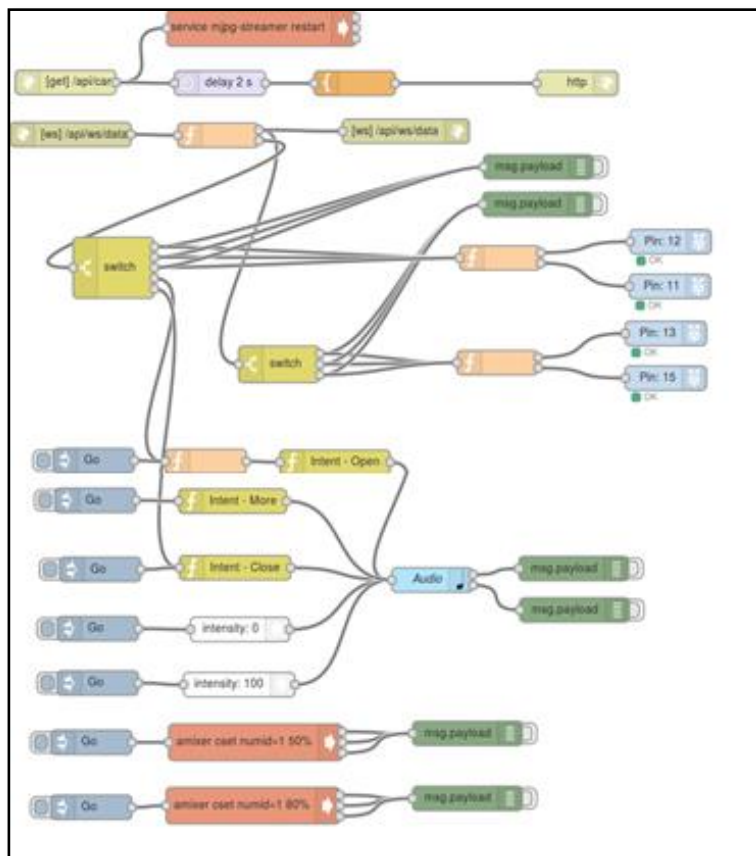


Figura 3.38: Flujo completo en Node Red del proyecto.

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

1. En este proyecto se implementó un sistema que integra hardware y software orientado a la supervisión remota.
2. Con los beneficios de la utilización del interface se logró tener el control del vehículo controlado a través de una conexión wifi que, dependiendo de la intensidad de su señal nos permitirá tener una mayor cobertura. La misma que nos brinda la opción de conectar un dispositivo móvil (teléfono celular) con el dispositivo Raspberry Pi que se encuentra ubicado en la parte superior del vehículo.
3. La herramienta Node-red que se utilizó para este proyecto nos permite una programación por bloques y se puede programar remotamente desde un explorador, convirtiéndose en un aliado estratégico para el proyecto y evitando así la programación en java que haría el proceso más largo.
4. Con el dispositivo Raspberry Pi se tiene la ventaja del soporte de un dispositivo de audio como un parlante que pueden ser implementados en campañas publicitarias incrementando la productividad de ese sector de la economía mejorando los niveles de ingresos de las personas de Ecuador.
5. Existen muchas formas de implementación para éste proyecto, valiéndose de distintos componentes eléctricos y dispositivos embebidos disponibles en el mercado.
6. Alrededor del mundo se han implementado proyectos similares con distintas herramientas de software y hardware pero ninguna solución pública combina el uso de Raspberry Pi y NodeRed, siendo NodeRed la herramienta que permite establecer la funcionalidad del proyecto de una forma muy simple.
7. Se puede agregar innovación en muchos aspectos a proyectos desarrollados con dispositivos embebidos como Raspberry Pi debido a la cantidad de componentes, módulos y sensores disponibles y fáciles de integrar.

Recomendaciones

1. Para este proyecto fue utilizado en el control de la placa de Raspberry Pi, sin embargo sería beneficioso manejar el control de los motores para dar una mayor funcionalidad como es la velocidad del vehículo.
2. El Raspberry Pi al ser un dispositivo de bajo costo tiene alta competencia en precios con dispositivos de similares características, se debería realizar planes estratégicos para lograr colocar su producto y así desplazar el dispositivo utilizado en este proyecto, por lo que se recomienda la actualización constante del mismo.
3. Para agregar valor a los proyectos y generar una mayor competencia en el mercado se pueden agregar sensores y módulos sencillos de fácil implementación en el Raspberry Pi; en el proyecto descrito en este documento se implementó el módulo de cámara web y la reproducción de audio como innovación.
4. Existe un sinnúmero de información en la web acerca del uso de Raspberry Pi, e implementaciones de proyectos con soluciones similares, y la explicación de todo su desarrollo; sin embargo es recomendable buscar información del uso de Raspberry Pi en las páginas oficiales de este producto, ya que en éstas se encontrará la mejor opción de uso de los componentes y del mismo dispositivo embebido.

BIBLIOGRAFÍA

- [1] Softeng, Metodología Scrum (2013, Junio 10). Disponible en: <https://www.softeng.es/eses/empresa/metodologias-de-trabajo/metodologia-scrum.html>.
- [2] Lic. Amarilys Castanedo Alfonso, Efectos de la ciencia y la tecnología en los problemas medioambientales (2013, Mayo 15). Disponible en: <http://www.monografias.com/trabajos75/efectos-ciencia-tecnologia-problemas-medioambientales/efectos-ciencia-tecnologia-problemas-medioambientales2.shtml>.
- [3] Alberto Polanco, Ciencia, Tecnología y Sociedad (2012, Enero 1). Disponible en: <http://www.monografias.com/trabajos5/cienteysoc/cienteysoc.shtml>.
- [4] La Hora, Tecnología al Servicio de la Agricultura (2010, Junio 22). Disponible en: <http://www.lahora.com.ec/index.php/noticias/show/1101431216/-1/Tecnolog>.
- [5] Carlos Espinoza León, Publicidad Auditiva, una delicia (2015, Agosto 11). Disponible en: <http://imagixel.blogspot.com/2010/12/publicidad-auditiva-una-delicia.html>.
- [6] Claudia Romero Ochoa, Música y Publicidad (2015, Junio 14). Disponible en: <http://www.monografias.com/trabajos86/musica-y-publicidad/musica-y-publicidad.shtml>.
- [7] Meli Herrera, Publicidad Auditiva (2008, Junio 2). Disponible en: https://prezi.com/bwnq2_vuyl_y/publicidad-auditiva/.
- [8] César Ocaña Romo, Avances Agrotecnológicos (2009, Febrero 1). Disponible en: <http://www.hortalizas.com/uncategorized/avances-agrotecnicos/>.
- [9] ACE, Elecciones y Tecnología (2013, Junio 3). Disponible en: <http://aceproject.org/ace-es/topics/et/etb/etb01/etb01b>.
- [10] Ing. Víctor Gómez, Experiencias con Raspberry Pi (2014, Diciembre 10). Disponible en: <http://muyraspi.blogspot.com/2013/02/que-es-la-raspberry-pi.html>.

[11] John Gutierrez, Detector de Personas con Raspberry Pi (2013, Junio 10).

Disponible en:

http://proyectossena686497.blogspot.com/2015/04/materiales_64.html.

[12] Kubii, Dongle WI-FI para Raspberry Pi (2013, Octubre15). Disponible en:

<http://www.kubii.es/modulos-extension-raspberry-pi/64-dongle-wipi-modulo-usb-wifi-3170111000385.html>.

[13] Node JS, Documentación. Disponible en: <https://nodejs.org/en/docs/>.

[14] Node Red, Acerca de Node Red. Disponible en: <http://nodered.org/>.

[15] Node Red, Creando Nodos. Disponible en: <http://nodered.org/docs/creating-nodes/node-html.html>

[16] Node Red, Creando Nodos. Disponible en: <http://nodered.org/docs/creating-nodes/node-js.html>

[17] John Hernandez, Proteus simulador de circuitos (2013, Agosto10). Disponible en: <http://www.taringa.net/post/ciencia-educacion/16301044/Proteus-simulador-de-circuitos.html>.

[18] Circuitos RPI GPIO como interfaz (2014, Octubre10). Disponible en:

http://elinux.org/RPi_GPIO_Interface_Circuits.

[19] Ubuntu, Descarga de escritorio de Ubuntu (2012, Agosto26). Disponible en:

<http://www.ubuntu.com/download/desktop>.

[20] Edita Suerias, Cómo instalar Filezilla, un cliente FTP, en Ubuntu (2014,

Diciembre 3). Disponible en: <http://www.lasticenelaula.es/portal/index.php/menu-internet/clientes-ftp/384-filezilla-cliente-de-ftp-tambien-en-linux32>.

[21] Instalar o Actualizar Firefox en Linux. Disponible en:

<http://www.windowlinuxymac.com/linux/21-instalar-o-actualizar-firefox-a-la-%C3%BAltima-versi%C3%B3n-en-linux.html>.

[22] Carlos Vialfa, Cómo instalar un servidor SSH en Ubuntu (2013, Agosto 17).

Disponible en: <http://es.ccm.net/fag/3560-como-instalar-un-servidor-ssh-en-ubuntu>.

- [23] The ThingBox, Preparar en Raspberry Pi. Disponible en: http://thethingbox.io/docs/Flash_Pi_image.html.
- [24] The ThingBox, Descubriendo the ThingBox. Disponible en: <http://thethingbox.io/docs/Discover.html>
- [25] The ThingBox, Agregando Adaptador Wifi. Disponible en: <http://thethingbox.io/docs/WiFi.html>.
- [26] Miguel Grinberg, Instalar y utilizar librería MJPG-Streamer en Raspberry Pi. (2013, Mayo 10) Disponible en: <http://blog.miguelgrinberg.com/post/how-to-build-and-run-mjpg-streamer-on-the-raspberry-pi>.
- [27] Node Red, Instalación. Disponible en: <http://nodered.org/docs/getting-started/installation.html>.
- [28] Sergio Luján Mora, Programación en Internet: Clientes Web , Editorial Club Universitario, 2001
- [29] Hojas de estilo en cascada. Disponible en: https://es.wikipedia.org/wiki/Hoja_de_estilos_en_cascada.
- [30] Flanagan, David; Ferguson, Paula, JavaScript: The Definitive Guide (4ta Edición) , 2002
- [31] Udemy, Desarrollo Responsive con el Framework Bootstrap 3. Disponible en: <https://www.udemy.com/desarrollo-web-frontend/>.