UNIVERSIDAD DE EL SALVADOR
FACULTAD DE INGENÍERIA Y ARQUITECTURA
ESCUELA DE INGENÍERIA ELÉCTRICA

Diseño y construcción de un prototipo de un sistema de seguridad basado en software y hardware de libre distribución con conexión a Internet por medio del protocolo IEEE 802.11

PRESENTADO POR:

MARIO ERNESTO ARGUETA TOBAR
RENÉ ANTONIO IRAHETA REYES
JUAN FRANCISCO MIRANDA TORRES
JULIO CÉSAR PORTILLO FERRUFINO

PARA OPTAR AL TÍTULO DE:
INGENIERO ELECTRICISTA

CIUDAD UNIVERSITARIA, FEBRERO DE 2015

UNIVERSIDAD DE EL SALVADOR

RECTOR :

ING. MARIO ROBERTO NIETO LOVO

SECRETARIA GENERAL :

DRA. ANA LETICIA ZAVALETA DE AMAYA

FACULTAD DE INGENIERIA Y ARQUITECTURA

DECANO :

ING. FRANCISCO ANTONIO ALARCÓN SANDOVAL

SECRETARIO :

ING. JULIO ALBERTO PORTILLO

ESCUELA DE INGENIERIA ELÉCTRICA

DIRECTOR :

ING. JOSÉ WILBER CALDERÓN URRUTIA

UNIVERSIDAD DE EL SALVADOR
FACULTAD DE INGENIERIA Y ARQUITECTURA
ESCUELA DE INGENIERIA ELECTRICA

Trabajo de Graduación previo a la opción al Grado de:

INGENIERO ELECTRICISTA

Título :

Diseño y construcción de un prototipo de un sistema de seguridad basado en software y hardware de libre distribución con conexión a Internet por medio del protocolo IEEE 802.11

Presentado por :

MARIO ERNESTO ARGUETA TOBAR
RENÉ ANTONIO IRAHETA REYES
JUAN FRANCISCO MIRANDA TORRES
JULIO CÉSAR PORTILLO FERRUFINO

Trabajo de Graduación Aprobado por :

Docente Asesor :

ING. JOSÉ WILBER CALDERÓN URRUTIA

San Salvador, febrero de 2015

Trabajo de Graduación Aprobado por:

Docente Asesor :

ING. JOSÉ WILBER CALDERÓN URRUTIA

ACTA DE CONSTANCIA DE NOTA Y DEFENSA FINAL

En esta fecha, 28 de noviembre de 2014, en la Sala de Lectura de la Escuela de Ingeniería Eléctrica, a las 10:00 horas, en presencia de las siguientes autoridades de la Escuela de Ingeniería Eléctrica de la Universidad de El Salvador:

1. Ing. José Wilber Calderón Urrutia
   Director — Firma: Wilber Calderón

2. Msc. e Ing. Salvador de Jesús Germán
   Secretario — Firma:

ESCUELA DE INGENIERÍA ELÉCTRICA
FACULTAD DE INGENIERÍA Y ARQUITECTURA
DIRECCIÓN

Y, con el Honorable Jurado de Evaluación integrado por las personas siguientes:

1- Ing. Jose Wilber Calderón Urrutia — Firma: Wilber Calderón

2- Ing. Gerardo Marvin Jorge Hernández

3- Msc. e Ing. Salvador de Jesús Germán

Se efectuó la defensa final reglamentaria del Trabajo de Graduación:

Diseño y construcción de un prototipo de un sistema de seguridad basado en software y hardware de libre distribución con conexión a Internet por medio del protocolo IEEE 802.11.

A cargo de los Bachilleres:

- Argueta Tobar Mario Ernesto
- Iraheta Reyes René Antonio
- Miranda Torres Juan Francisco
- Portillo Ferrufino Julio César

Habiendo obtenido en el presente Trabajo una nota promedio de la defensa final, de: 9.2

(NUEVE . DOS)

AGRADECIMIENTOS

Primeramente a DIOS TODOPODEROSO, por darme la oportunidad y la bendición de terminar mis estudios, por darme la fortaleza, fe, salud y esperanza para alcanzar este anhelo que hoy se vuelve una realidad, siempre estuviste a mi lado y me dotaste de grandes dones y talentos que hoy puedo utilizar en mi vida, gracias por escuchar mis oraciones y darme la fuerza cada día para seguir por el camino del bien.

Dedico esta Tesis a la memoria de a mi PADRE Ramón (de grata recordación) Y A MI MADRE Reina; quienes permanentemente me apoyaron con espíritu alentador, contribuyendo incondicionalmente a lograr mis metas y objetivos propuestos. Gracias por todo su esfuerzo, al punto muchas veces de sacrificar sus propios sueños por ayudarme a alcanzar los míos. Gracias el apoyo que siempre me brindaron, por su amor que me motivo a alcanzar el éxito, porque sin mis padres, sin sus enseñanzas y consejos, no sería quien ahora soy.

A mi FAMILIA, en primer lugar a mi amada ESPOSA Estelita, gracias por su paciencia, por su comprensión, su dedicación, su fuerza y por su amor, por ser tal y como eres. A mis HIJOS Rodrigo y Mario, porque ustedes son la razón de mi vida, la motivación de mi lucha diaria. Les dedico esta tesis por el tiempo que no pudimos compartir y que hoy se transforma en el mejor de mis logros para ustedes.

A MIS AMIGOS Y COMPAÑEROS. Con los que compartimos nuestro tiempo de estudios y amistad y que en los momentos de dificultad y ansiedad estuvieron pendientes, dándome el ánimo necesario para no flaquear. Y especialmente a mis compañeros de Tesis, Julio, René y Francisco, porque con ustedes compartí muchos momentos, a veces de dificultad y otras veces de alegría al ver como vencíamos esas dificultades. Gracias por su apoyo y su amistad.

A MIS MAESTROS que me han acompañado durante el largo camino, brindándome siempre su orientación con profesionalismo ético en la adquisición de conocimientos y afianzando mi formación como estudiante universitario.

Mario Ernesto Argueta

AGRADECIMIENTOS

Señor, gracias por darme la motivación, convicción y la fuerza necesaria para poder completar una tarea más en mi vida.

Gracias por Iluminarme con tu luz y guiarme de una manera impecable a través de los tantos obstáculos que tuve en mi camino y por mantenerme firme cuando en algún momento el camino se tornó difícil.

Gracias por la Salud y por tu protección a lo largo del camino. Gracias por las bendiciones que hasta ahorita he recibido.

Gracias por regalarme una familia y amigos tan maravillosos que tú has puesto en mi camino, Pidiéndote que por favor los cuides y los guíes de la manera como hasta el momento lo has hecho conmigo.

Agradezco a mi Madre Blanca Miriam Reyes de Iraheta por su apoyo incondicional durante toda mi vida y en mi proceso de formación profesional; agradezco a mi Padre René Iraheta González; a mi hermana Miriam Karina Iraheta por estar conmigo, siendo ellos los que me motivan todos los días para seguir creciendo profesionalmente.

A mis amigos y amigas que siempre estuvieron presentes con esas palabras de aliento y le dieron a mi vida una razón de ser.

Quiero agradecer a los profesores que impartieron y compartieron sus conocimientos durante los años de Formación porque me permitieron terminar con éxito este proyecto. A mis asesores por su colaboración y guiarme en la culminación exitosa de este esfuerzo.

Muchas gracias a todos y que Dios les llene de bendiciones.

René Antonio Iraheta Reyes

AGRADECIMIENTOS

“Aquel que es poderoso para hacer todas las cosas mucho más abundantemente de lo que pedimos o entendemos, según el poder que actúa en nosotros”.

Gracias Padre en el nombre de nuestro señor y salvador Jesucristo por darme el hálito de vida, por fortalecer mi alma mediante su palabra eficaz y poder tener la honra de finalizar mis estudios superiores en la Universidad de El Salvador.

“he peleado la buena batalla, he acabado la carrera y he guardado la Fe“

Palabras donde retomo la fidelidad y Agradecimiento para mi Dios, por lo cual doblo mis rodillas y rindo la gloria, la honra, la alabanza y la adoración a ti señor Jesús.

Agradezco a mis padres Hno. Pablo Francisco Miranda y Hna. María Elena Torres de Miranda; cabeza y ayuda idónea en mi hogar, siendo su herencia y bendición, por darme lo mejor a base de esfuerzo y arduo trabajo, reconozco ese apoyo que solo los padres pueden otorgar. Han sido y son mi primera escuela, los principios y valores, los consejos, sus oraciones. Mis hermanos Elena Elizabeth y Pablo Samuel me han apoyado en gran manera, en cuanto lo he necesitado a lo largo de mis estudios y lo siguen haciendo como hermanos que somos.

A cada profesor de la Escuela de Ingeniería Eléctrica por los conocimientos adquiridos. Con mucha estima Msc. Ing. José Wilber Calderón Urrutia por formar y ser parte de un equipo, estar en toda la disposición de buscar y guiarnos a las mejores soluciones en este trabajo de tesis.

A Reina Isabel Vides, Salvador posada, Juan Olano a cada uno, su labor es muy importante para el desarrollo de los estudiantes de la Escuela de Ingeniería Eléctrica.

Agradecer a mis compañeros de tesis: Mario Ernesto, Rene Antonio, Julio Cesar por tomar la decisión de desarrollar y finalizar este trabajo de tesis de la mejor manera y haber compartido uno a uno la entrega de avances: que al finalizar, vemos la recompensa de nuestro esfuerzo.

Agradecer a cada uno mis amigos y compañeros de la EIE con quienes tuve el honor de compartir tiempo de estudio para prepararnos a aprobar cada materia, quienes forman parte de SEU y durante el desarrollo de la tesis mencionar a: Fredy Marenco, Saúl Rosa, Alexander Ivanov, Johan Morales, Ronald Escobar, Santiago Palma y Pedro Mercado.

A Julia Miranda, Guillerma González, Salvador Miranda y Juan Jesús Torres, quienes les llevo en mi corazón.

Juan Francisco Miranda Torres

AGRADECIMIENTOS:

Al llegar a esta etapa en mi formación profesional; agradezco a Dios todopoderoso que hace lo imposible posible a través de personas maravillosas, a mi madre Reina Aracely que desde donde se encuentre, me supo orientar a tomar buenas decisiones. A mi padre Julio César, por su apoyo constante y buenos consejos. A mis hermanas Ligia y Karina por estar siempre ahí, segundo a segundo. A mi hermano Manuel por ser un apoyo incondicional a lo largo de mi formación. A mis hermanas Marcela y Graciela, a mis sobrinas Emely y Raquel, y mi sobrino Isaac, a todos ellos, por ser la nueva generación de la familia. A mis abuelos Manuel y Graciela, Juan y Emely; que con sus lecciones de vida moldearon de alguna manera mi carácter. A todos ellos gracias.

A mi tía Delis y tía Nelly, por ayudarme también en este camino.

También quiero agradecer a mis compañeros Juan, Rene y Mario por su constante apoyo y perseverancia en la realización de este proyecto.

Agradecer al señor Posada y Juancito por ser apoyo directo en la realización del trabajo.

Agradecimientos especiales al Ing. Wilber Calderón, por creer en el grupo y por sus buenos consejos, recomendaciones y sobre todo; por su paciencia.

A los Jurados Ing. Salvador German e Ing. Marvin Hernández, por sus buenas observaciones y recomendaciones.

A mis compañeros Fredy, Saul, Alex, Ronald, Cesar, Roberto, Pedro, Johan, Mauricio y todos aquellos que de alguna manera me ayudaron en este camino.

A todos los catedráticos de la escuela de ingeniería eléctrica y de la facultad, por formar parte de mi preparación y formación ante esta etapa que se aproxima.

A mis amigas y amigos que estuvieron ahí, ayudándome por sus buenos consejos y palabras acertadas en momentos difíciles.

Y también agradecer a todas aquellas personas que me ayudaron de alguna manera a culminar una etapa en mi vida.

Muchísimas gracias.

Hacia la Libertad por la Cultura.

Julio César Portillo Ferrufino.

RESUMEN

El sistema prototipo de seguridad basado en software y hardware libre a desarrollar, consiste en una Unidad de control independiente capaz de llevar a cabo un monitoreo continuo de un espacio físico concreto por medio de tres tipos de sensores y la transmisión de video en las áreas protegidas.

De modo que una vez detectado algún tipo de movimiento o intrusión, se active el sistema según el nivel de alarma, el cual será monitoreado por el personal de seguridad de la Universidad de El Salvador mediante una red de área local utilizando el protocolo IEEE 802.11 (Institute of Electrical and Electronics Engineers 802.11, Instituto de Ingeniería Eléctrica y Electrónica 802.11).

La aplicación tiene la particularidad de ser un prototipo de bajo costo, el cual es accesible para los posibles usuarios.

The prototype software based security and freedom to develop hardware system consists of a separate control unit capable of performing continuous monitoring of a particular physical space through three types of sensors and video transmission in protected areas.

So once detected some kind of motion or intrusion, the system is activated by level alarm, which will be monitored by security personnel at the University of El Salvador by a local area network using the protocol IEEE 802.11 (Institute of Electrical and Electronics Engineers 802.11, Institute of Electrical and Electronics Engineers 802.11).

The application has the distinction of being a low-cost prototype, which is accessible to potential users.

TABLA DE CONTENIDO

LISTA DE FIGURAS.

LISTA DE TABLAS

INTRODUCCIÓN.

La vigilancia es un tema importante en la actualidad y de alta demanda como servicio, debido principalmente al problema de incrementos delincuenciales que sufre el país, la Universidad de El Salvador, no está exenta de este, y como ente rector superior de educación, debe presentar propuestas concretas sobre las posibles soluciones para palear esta problemática. Una solución técnica de muchas es la implementación de un sistema de seguridad de bajo costo y funcional.

La inseguridad es un problema sistemático e integral causado por muchos factores, entre los cuales se tienen: factores políticos, sociales, económicos, éticos, morales y culturales. Sin embargo, el fenómeno de la delincuencia, no deja de ser un problema porque conlleva a resultados negativos directos como robo y hurto en personas e inmuebles, derivando a su vez en problemas consecuentes como, pérdidas económicas, de información, etc.

Los sistemas de seguridad han tomado gran relevancia en los últimos tiempos, esto es posible en gran medida a que las tecnologías actuales lo permiten porque su desarrollo es continuo. Más específicamente la tecnología de las redes informáticas; es la responsable directa de la expansión de los sistemas de seguridad y ha potencializado de sobremanera el desarrollo de éstos, de tal manera que en la actualidad, es más económico y eficiente utilizar redes de seguridad basados en protocolos de comunicación IP que los sistemas considerados tradicionales.

Además de las ventajas de eficiencia y economía, los sistemas de seguridad detectan ilícitos en tiempo real y los respaldan para ser utilizados en casos legales, al igual que un sistema basado en tecnologías tradicionales.

Como ventaja adicional y no menos importante es el desarrollo de los sistemas de seguridad basados en protocolos de comunicación IP (Internet Protocol, Protocolo Internet) de software libre. Pues, si bien es cierto que los sistemas de seguridad basados en comunicación IP con respecto a la manera tradicional de CCTV (Closed Circuit Television), Circuitos Cerrado de Televisión) su implementación utilizan licencias de software de propiedad privada causando un elevado coste, siendo esta una desventaja que se aumenta considerando que el hardware también es privado. Afortunadamente y dado el avance tecnológico, existe el software y hardware libre; reduciendo significativamente los costos de implementación.

Las tecnologías para la seguridad electrónica y video vigilancia requiere la búsqueda continua de sistemas más útiles y menos costoso para los posibles usuarios. Por esta razón, el desarrollo de nuevas propuestas con características extras a muchos productos en el mercado actual, es el motivo para el desarrollo de mejores y nuevos prototipos.

Muchos de los productos de video vigilancia comerciales presentan dependencia de Internet o de instalaciones dedicadas para poder operar. Por esta razón, se desea proponer los elementos principales de un prototipo de video vigilancia que lo integren puntos de control formados por la placa Raspberry Picon capacidad de operar de forma independiente, sensores de diferente tipo y una cámara web con el fin de obtener una opción de bajo costo con características competitivas ante dispositivos similares, como por ejemplo cámaras IP.

Por lo cual, la motivación es utilizar como hardware libre la placa de desarrollo Raspberry Pi con capacidad de comunicación WLAN(Wireless Local Area Network, Red de Área Local Inalámbrica) en la captura de imágenes mediante la cámara web y del almacenamiento de información en base de datos externa, se debe a la oportunidad de implementar un prototipo con una mini computadora de reciente aparición en el mercado, con capacidades de procesamiento de datos, con poco consumo de energía, con manejo de entradas y salidas, que permiten el desarrollo de proyectos tecnológicos, a un costo moderado.

PLANTEAMIENTO DEL PROBLEMA.

El planteamiento del problema surge a raíz de la inseguridad, la cual no contribuye de ninguna manera al desarrollo de la sociedad.

El alma mater, siendo la institución de formación de profesionales puede dar una solución ante el problema de inseguridad haciendo uso de la tecnología actual, en la Facultad de Ingeniería y Arquitectura se cuenta con el conocimiento que se requiere conveniente para el proceso de desarrollo de este proyecto para fortalecer la seguridad. Para esto se definen dos situaciones que se detallan más adelante. La situación actual se llamará estado A inicial, el cual describe lo que acontece diariamente en el recinto universitario desde la perspectiva de seguridad; y la que sería el estado B final, el cual se pretende alcanzar con el desarrollo del prototipo de sistema de seguridad.

1.1 ESTADO A.

Como se sabe, la situación entorno a la seguridad en El Salvador, no es la mejor, la Facultad de Ingeniería y Arquitectura ha tomado acciones de prevención, como la contratación de seguridad privada[1].En su momento la Universidad de El Salvador también ha dado seguimiento al acceso peatonal y vehicular mediante la adquisición de un sistema electrónico privativo e instalación de plumas en las casetas de control, por medio del DUE y la instalación de cámaras en la zona de parqueo.

En la fecha de 4 de abril del 2012 la Universidad de El Salvador hizo una inversión de 1 millón de dólares[2] en la mejora de su seguridad; pero a pesar de esto, sigue ocurriendo el ingreso de algunas personas que cometen actos ilegales dentro de las instalaciones de la universidad y en el interior de los edificios, violentando el patrimonio de la universidad y la integridad física de la comunidad universitaria.

1.2 ESTADO B.

Con el sistema prototipo de seguridad implementado en este trabajo de graduación se espera llegar al estado B, el cual se describe a continuación:

[1] Comunicado a Facultad de Ingeniería y Arquitectura, 11 de octubre del 2012: https://www.facebook.com/permalink.php?id=239091939469748&story_fbid=443180342394239

[2] https://www.facebook.com/permalink.php?story_fbid=10150532049575202&id=294546065201

- El campus universitario tendría un beneficio al mejoramiento de la seguridad.

- Coordinación de un mejor control en alertas ante los hechos que se presentarían en el campus universitario.

- Un Control eficiente de personas autorizadas a acceder a los edificios en caso de fines de semana, días no laborales, asuetos y horarios fuera de jornada, a través de contraseñas válidas.

- a través del sistema de video vigilancia se tendrían pruebas o evidencias de actos ilícitos.

- Presentar una solución de bajo costo, implementando la filosofía de hardware y software libre ante ofertas limitadas y privativas.

ANTECEDENTES.

Los esfuerzos de la Universidad de El Salvador por mejorar la situación de seguridad se está realizando[3] pero es necesario el aporte de la comunidad estudiantil para el desarrollo de trabajos de investigación en el sentido salvaguardar, la integridad de docentes, personal administrativo y el patrimonio de la Universidad.

En esa línea, el trabajo de graduación pretende sentar una base para cumplir con dicho propósito. Además, dada la reciente experiencia que se presentó en época de vacaciones en la escuela de ingeniería eléctrica en donde se hurtaron bienes de docentes, dispositivos como multímetro, osciloscopio de laboratorio de telefonía móvil[4]. Queda evidenciado el propósito.

JUSTIFICACIÓN.

El trabajo de graduación a desarrollar brinda una aportación al cuido del patrimonio de la Universidad de El Salvador, utilizando como herramientas, el software libre y hardware libre, siendo una ventaja tecnológica para la realización de un sistema de seguridad de bajo coste.

Debido a la experiencia en la Escuela de Ingeniería Eléctrica, poseer un sistema de seguridad sobre este edificio sería un beneficio, una solución ante un acto de delincuencia; evitando así la perdida de bienes inmuebles,

Brindar una herramienta al personal de seguridad de la Universidad de El Salvador.

[3] http://www.eluniversitario.ues.edu.sv/index.php?option=com_content&view=article&id=1136:csu-trabaja-en-proyecto-de-ordenamiento-y-seguridad-del-campus&catid=41:acontecer&Itemid=30

[4] Fotografía publicadas por Ing. Carlos Pocasangre Jiménez
https://www.facebook.com/carlos.pocasangrejimenez/posts/10151919223567149

OBJETIVOS.

OBJETIVO GENERAL:

Diseñar y construir un prototipo de un sistema seguridad basado en software y hardware monitoreado utilizando el protocolo IEEE 802.11

OBJETIVOS ESPECÍFICOS:

- Aplicar protocolos de comunicación WLAN (Wireless local Area Network, Red de Área Local Inalámbrica) y Ethernet así como también el tratamiento digital de la señal para el envío de imágenes y vídeos a través de Internet.
- Desarrollar un servidor usando software de libre distribución para el monitoreo de seguridad, el cual tendrá una capacidad de hasta 30 edificios.
- Implementar un sistema de seguridad que permita detectar la presencia humana, hora, imagen o vídeo para ser transmitido al centro de control por medio del protocolo IEEE 802.11
- Desarrollar un sistema para autenticar que la presencia humana es parte del personal autorizado.
- Utilizar sensores de presencia, sensores de vibración e interruptores magnéticos, sensores de detección de humo y sensores fotosensibles activados por láser para la detección de presencia humana en cada punto protegido.

FUNDAMENTOS TEÓRICOS.

SISTEMA DE SEGURIDAD.

Un sistema de seguridad es, al igual que cualquier red, una interconexión de elementos que buscan un propósito específico, y este es el de salvaguardar la integridad física de las personas y sus bienes ante cualquier acto de robo y vandalismo en las áreas protegidas.

Existen en el mercado, muchos sistemas de seguridad destinados a satisfacer dicho propósito; marcas reconocidas mundialmente están a la vanguardia en el diseño, instalación, gestión y mantenimiento de sistemas de seguridad. Estas marcas, al ser reconocidas mundialmente, el costo de los sistemas es alto debido a que son empresas dedicadas a este rubro. Esto es debido en gran medida a que los componentes tanto en software, como hardware, son muchos de ellos comerciales y su manufactura es específica; por ejemplo, las cámaras, monitores, servidores, dispositivos VCR (Video Cassette Recorder, Videograbadora), grabadores y cualquier dispositivo adicional; son elementos propios. Cualquiera que sea la forma de manufactura, dichos componentes pertenece a las compañías que prestan el servicio tanto en hardware como en software. Existe alternativa también a esos sistemas de seguridad.

Estos se auxilian de una forma particular para su realización y consiste en utilizar software y hardware de libre distribución para cumplir con los objetivos fundamentales que posee cualquier sistema de seguridad. Sea cual sea el caso, en ambos sistemas, es necesario conocer los componentes de un sistema de seguridad, cuáles son las tareas de cada componente, como es la interconexión entre los elementos, así como la terminología que se utiliza.

En la actualidad, los sistemas de vídeo vigilancia realizados son conocidos como video vigilancia basada en IP o Vigilancia IP, y presentan muchas ventajas con respecto a los sistemas de circuito cerrado de televisión o CCTV analógicos tradicionales. Estos sistemas se realizan a través de redes IP (Internet Protocol / Protocolo de Internet) cableadas o inalámbricas; las cuales permiten trasmisiones de vídeo, audio y otros datos a través de la misma infraestructura de red[5].

En la figura 1 se muestran los componentes básicos de un sistema de vídeo vigilancia IP con todos los elementos.

[5] Guía técnica de vídeo IP. Factores y técnicas a considerar para un correcto uso de las aplicaciones de vídeo vigilancia y monitorización remota basadas en IP. Axis comunicaciones. http://www.axis.com/

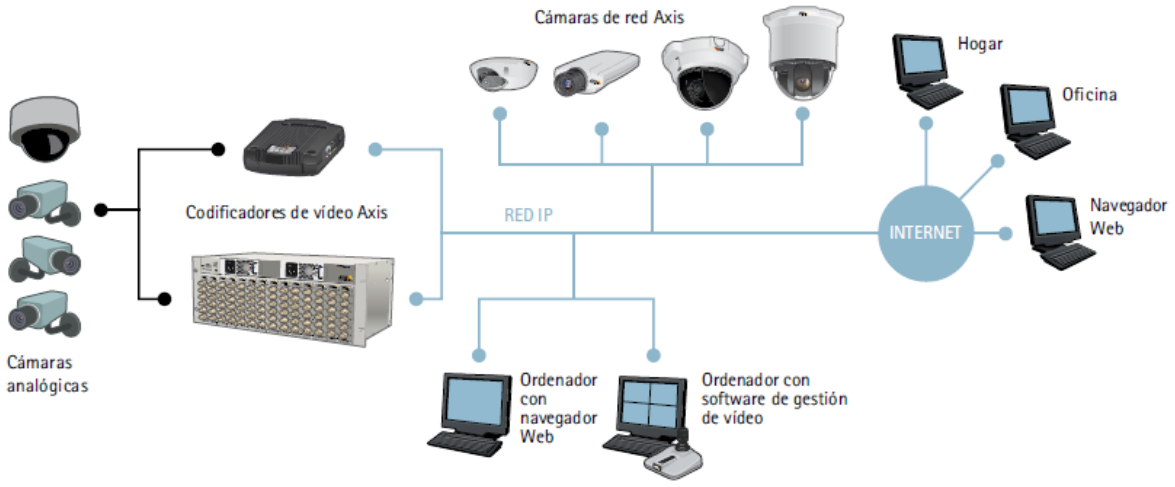

Figura 1.Componentes de un sistema de vídeo en red.[6]

Se observa que un sistema de vídeo vigilancia IP consta de elementos como: Red IP, cámaras IP, cámaras analógicas y codificadores de vídeo, computadoras personales (PC) de escritorio o laptops, navegador web, software de gestión de vídeo e internet.

Además de los elementos de vídeo vigilancia IP; los sistemas de seguridad utilizan redes de sensores individuales o nodos; formando una red hibrida, que consta de una red para vídeo vigilancia y una red de sensores electrónicos teniendo el sistema de seguridad, mayor capacidad de control en las áreas a cubrir. Estas redes de sensores tienen diversos componentes entre los cuales se tienen: sensores de movimiento o presencia, sensores magnéticos, sensores de vibración, sirenas, etc. Formando un conjunto de elementos que se comunican entre sí a través del protocolo de comunicación IP.

Un sensor se puede definir como un dispositivo capaz de recoger información de los distintos parámetros físicos o químicos que controlan; para luego transmitir esa información para su procesamiento en forma de señal eléctrica. En esa dirección, los fabricantes de sistemas de seguridad aprovechan esa característica que poseen los sensores para extender sus sistemas de seguridad. La estructura de una red de sensores va de acuerdo a su tecnología. Estas tecnologías son diversas entre las cuales se mencionan: los estándares IEEE 802.11 /WIFI, Bluetooth, IEEE 802.15.4 /Zigbee, tecnologías inalámbricas sub-GHZ entre otras.

Cada una de ellas presenta ventajas sobre las otras, la tecnología que mayor desarrollo posee es la IEEE 802.15.4 / Zigbee. Sin embargo, en algunos sistemas, utilizan el estándar IEEE 802.11 para transmitir las señales eléctricas de los sensores, en esa línea, el protocolo de comunicación IEEE 802.11 dada las características que presenta, puede ser utilizado como protocolo de comunicación. En la figura 2 se observa un sistema de sensores con sus componentes.

[6]http://www.axis.com/es/products/video/about_networkvideo/network_video.htm

Figura 2. Componentes de un sistema electrónico de seguridad[7].

Al combinar ambas redes se forma una red integral de seguridad o sistema de seguridad integral. Algunos fabricantes incorporan a sus sistemas, redes CCTV; integrándose a la red IP a través de dispositivos codificadores/decodificadores; uniendo tecnologías antiguas.

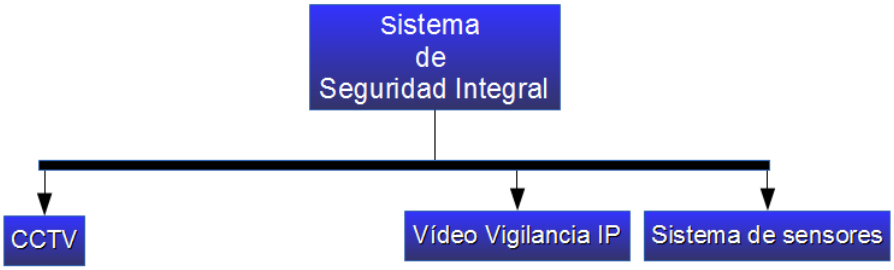

Figura 3. Elementos principales de un sistema de seguridad.

Los sistemas de seguridad actuales, ya sean comerciales o de libre distribución, muchos de ellos utilizan redes IP de tecnología inalámbrica, que dentro de sus ventajas están la implementación de sistemas de una manera flexible, rentable y rápida; pues se elimina la necesidad de utilizar redes alámbricas con cable Ethernet. Dentro del abanico de tecnologías inalámbricas se tiene el protocolo de comunicación IEEE 802.11.

Estándar IEEE 802.11.

El estándar más habitual para redes inalámbricas de área local (WLAN) es la norma IEEE 802.11. Si bien existen otros estándares y otras tecnologías patentadas, la ventaja de utilizar los estándares inalámbricos IEEE 802.11 es que funcionan en un ámbito sin licencia, de manera que no implican ningún coste asociado a la configuración y al funcionamiento de la red. Las extensiones más relevantes del estándar son IEEE: 802.11b, 802.11g, 802.11a y 802.11n.[8]

[7] http://quito.olx.com.ec/sistemas-integrados-de-seguridad-iid-44186249#galleryContainer
[8] http://www.axis.com/es/products/video/about_networkvideo/wireless.htm

En la tabla 1 se presenta un resumen de las variantes en los parámetros técnicos para el estándar de acuerdo a su versión.

Protocolo 802.11	Aprobado	Frecuencia	Ancho de Banda	Tasa de datos por flujo	Alcance aproximado en Interiores		Alcance aproximado en Exteriores	
		[GHZ]	[MHZ]	[Mbits/s]	[m]	[ft]	[m]	[ft]
-	Jun-1997	2.4	20	1, 2	20	66	100	330
a	Sep-1999	5	20	6, 9, 12, 18, 24, 36, 48, 54	35	115	120	390
b	Sep-1999	2.4	20	1, 2, 5.5, 11	35	115	140	460
g	Jun -2003	2.4	20	6, 9, 12, 18, 24, 36, 48, 54	38	125	140	460
n	Oct-2009	2.4/5	20	7.2, 14.4, 21.7, 28.9, 43.3, 57.8, 65, 72.2	70	230	250	820
			40	15, 30, 45, 60, 90, 120, 135, 150				
ac	Nov-2012	5	20	Up to 87.6				
			40	Up to 200				
			80	Up to 433.3				
			160	Up to 866.7				

Tabla 1. Valores nominales de parámetros del protocolo IEEE 802.11[9]

CÁMARAS

Cámaras web o webcam:

"Webcam" significa cámara para uso en red. Es un dispositivo que se conecta al puerto USB de la computadora, y así permite captar video y tomar fotos digitales con resolución baja, por lo que no ofrece una gran calidad de gráficos a diferencia de una cámara fotográfica digital, videocámara digital o un teléfono celular moderno. El video que capta, lo codifica especialmente para enviarlo por Internet (red mundial de redes) en tiempo real (lo más instantáneamente posible), hacia otra computadora dónde otro usuario puede visualizarlo al momento. Son muy utilizadas para conversaciones vía Internet y hacer más personalizada la charla, así como también para actividades de vigilancia. Este tipo de cámaras carecen de sistema operativo[10].

Figura 4. Webcam USB.

[9] Redes Inalámbricas en los Países en Desarrollo. Cuarta edición. Página 32.
[10] http://www.informaticamoderna.com/Camara_web.htm#defi

Cámaras en red o cámaras IP.

Una cámara IP es una combinación de una cámara y una computadora en una sola unidad, la cual captura y transmite imágenes en vivo a través de una red IP, habilitando a los usuarios autorizados a ver, almacenar y administrar el video sobre una infraestructura de red estándar basada en el protocolo IP.(Juan pablo Ycezalaya)[11].

Figura 5. Vista posterior de una cámara IP.[12]

Las cámaras constituyen un elemento principal para un sistema de video vigilancia y por lo tanto, su selección repercute en el nivel de calidad máximo que puedan lograrse en las imágenes que se registran[13].

Las cámaras pueden ser:

- Diurnas o con visión nocturnas (IR).
- Domo, mini domos, tipo box, o tipo bala, ambientes exteriores e interiores.
- PTZ (PANT-TILT-ZOOM), lente fija o variable.
- Con o sin audio.

Figura 6. Diferentes tipos de cámaras IP.[14]

[11] http://www.rnds.com.ar/articulos/031/RNDS_084W.pdf
[12] http://www.rnds.com.ar/articulos/031/RNDS_084W.pdf
[13] Aprenda a instalar cámaras de seguridad. Ing. Sergio Bellechasse Lissabet.
[14] http://www.ipcam.com.mx/

Todas las cámaras, ya sea de vídeo o de fotografías, poseen sensores de imagen. Este es un dispositivo electrónico que capta la luz que compone la imagen y la entrega en una señal eléctrica. Existen dos tipos de sensores mayormente utilizados: El sensor CCD (dispositivo de acoplamiento de carga) y sensor CMOS (semiconductor de óxido metálico complementario).

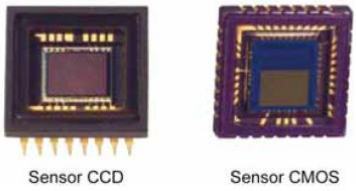

Figura 7. Forma física de sensor CCD y CMOS[15]

En aplicaciones de vídeo vigilancia se deberá considerar las ventajas y desventajas de cada tipo.

Tecnología del sensor	Ventajas	Desventajas
CCD	Alta sensibilidad a la luz y longitudes de onda	Mayor coste económico que los CMOS.
	Bajo nivel de ruido.	Requiere de electrónica de control externa.
	Alta calidad de imagen.	Mayor consumo eléctrico.
	Amplio rango dinámico. (diferencia entre el umbral de iluminación del elemento fotosensible y su nivel de saturación.	Existencia de Blooming: Cuando un píxel recibe demasiada iluminación produce un exceso de carga que es esparcido entre los pixeles adyacentes y se aprecia como área difusa por exceso de iluminación.
CMOS	Consumo eléctrico inferior al CCD.	Menor superficie receptora de la luz por píxel.
	Económico (necesita pocos componentes externos).	Menor uniformidad de los píxeles.
	El conversor digital puede estar integrado en el	Escasa sensibilidad a la luz ultravioleta e infrarroja.
	Escaso Blooming o inexistente.	Menor rango dinámico.
	Mayor flexibilidad en la lectura. Lectura simultánea de pixeles, lo que posibilita la previsualización más rápida.	
	Muy alta frecuencia de imagen en comparación a un CCD del mismo tamaño.	

Tabla 2. Ventajas y desventajas de los sensores CCD y CMOS[16]

Además del tipo de sensor de imagen en las cámaras, existen otras características técnicas tales como: Sensibilidad, formato del sensor, resolución.

La sensibilidad cuantifica la iluminación (iluminancia) mínima necesaria para producir una imagen en el sensor. Mientras menor sea su valor, mayor sensibilidad tendrá el dispositivo y viceversa. Su unidad es el LUX (lx). Las dimensiones (Largo x Ancho) del área sensible del

[15] Guía técnica de vídeo IP. Factores y técnicas a considerar para un correcto uso de las aplicaciones de vídeo vigilancia y monitorización remota basadas en IP. Axis comunicaciones. http://www.axis.com
[16] Aprenda a instalar cámaras de seguridad. Ing. Sergio Bellechasse Lissabet.

sensor de imagen se denomina formato óptico y de manera estándar se expresa en fracciones de pulgadas (Longitud diagonal) figura. Los formatos estándares más utilizados en sistemas de vídeo vigilancia son: 1/2", 1/3", 1/4", aunque existen otros formatos; mientras mayor sea el formato de sensor, más cantidad de luz recibe y su sensibilidad es mayor[17].

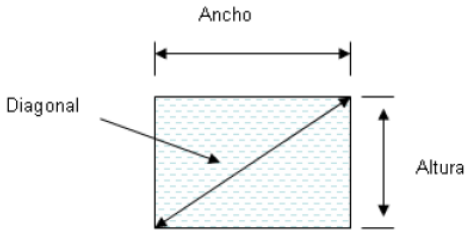

Figura 8. Formato de Óptico del sensor: Área sensible del sensor de imagen[18]

Resolución de cámaras.

La resolución en un mundo digital o analógico es parecida, pero existen algunas diferencias importantes sobre su definición. En el video analógico, una imagen consta de líneas o líneas de TV, puesto que la tecnología de vídeo deriva de la industria de la televisión. En un sistema digital, una imagen está formada por pixeles cuadrados. Algunas de las más populares para sistemas de vídeo vigilancia IP son: NTSC, PAL, VGA, megapíxel y HDTV. Se mencionan brevemente las primeras tres.

RESOLUCIONES NTSC Y PAL.

Las resoluciones NTSC (National Television System Comite, Comité Nacional de Sistemas de Televisión) y PAL (Phase Alternating Line, Línea de Alternancia de Fase) son estándares de vídeo analógico. Son relevantes para el video en red, ya que los codificadores de video proporcionan dichas resoluciones al digitalizar señales de cámaras analógicas. Las cámaras de red PTZ actuales y las cámaras domo de red PTZ también ofrecen resoluciones NTSC y PAL, puesto que hoy en día utilizan un bloque (que incorpora la cámara, zoom, enfoque automático y funciones de iris automático) hecho para cámaras de video analógico, conjuntamente con una tabla de codificación de vídeo integrada.

En Norteamérica y Japón, el estándar NTSC es la norma de vídeo analógico que predomina, mientras que en Europa y en muchos países de Asia y África se utiliza la norma PAL. Ambos estándares proceden de la industria de la televisión. El NTSC tiene una resolución de 480 líneas y utiliza una frecuencia de actualización de 60 campos entrelazados por segundo (o 30 imágenes completas por segundo)[19].

Resoluciones VGA

Con los sistemas 100% digitales basados en cámaras de red se pueden proporcionar resoluciones derivadas de la industria informática y normalizada en todo el mundo, de

[17]Aprenda a instalar cámaras de seguridad. Ing. Sergio Bellechasse Lissabet.

[18]Idem

[19] Guía técnica de vídeo IP. Factores y técnicas a considerar para un correcto uso de las aplicaciones de vídeo vigilancia y monitorización remota basadas en IP. Axis comunicaciones. http://www.axis.com

modo que la flexibilidad es mayor. VGA (Tabla de Gráficos de Video) es un sistema de pantalla de gráficos para PC desarrollado originalmente por IBM. Esta resolución es de 640 x 480 pixeles, un formato habitual en las cámaras de red que no disponen de megapíxeles. La resolución VGA suele ser más adecuada para cámaras de red, porque el video basado en VGA produce pixeles cuadrados que coinciden con los de las pantallas de las PC's. Los monitores de PC manejan resoluciones en VGA o múltiplos de VGA[20]. Tabla.

Formato de visualización	Píxeles
QVGA (SIF)	320x240
VGA	640x480
SVGA	800x600
XVGA	1024x768
4x VGA	1280x960

Tabla 3. Formatos de visualización VGA o múltiplos[21]

COMPRESIONES DE VÍDEO.

Cuando se está desarrollando una aplicación de vídeo vigilancia los desarrolladores consideran inicialmente algunas interrogantes: ¿Son necesarias imágenes estáticas o en movimiento?, ¿Cuál es el ancho de banda de la red?, ¿Qué nivel de degradación de imágenes resulta aceptable?, ¿A cuánto asciende el presupuesto para él sistema?

Cuando se digitaliza una secuencia de vídeo analógica de acuerdo al estándar CCIR 601[22] puede consumir aproximadamente 165 Mbps (Megabit por segundo), es decir 165 millones de bits cada segundo. Aunque la mayoría de las aplicaciones de vigilancia rara vez comparte la red con otras aplicaciones intensivas en datos, es realmente infrecuente encontrar este ancho de banda disponible. Para solventar este problema una serie de técnicas, denominadas técnicas de compresión de vídeo e imágenes, han sido creadas para reducir este elevado ratio de bits. Su capacidad para realizar esta tarea se cuantifica por el ratio de compresión, es decir, el menor consumo de ancho de banda que consigue. En todo caso hay que pagar un precio por esta compresión porque el aumento de la compresión genera una mayor degradación de la imagen. A esto se le denomina artifacts[23].

Las técnicas de compresión más comúnmente utilizadas son: JPEG, Motion JPEG y MPEG. Estas tres técnicas son muy importantes en sistemas de vídeo vigilancia; porque reducen el ancho de banda necesario significativamente para la transmisión de las imágenes.

[20]Guía técnica de vídeo IP. Factores y técnicas a considerar para un correcto uso de las aplicaciones de vídeo vigilancia y monitorización remota basadas en IP. Axis comunicaciones. http://www.axis.com.

[21]Idem

[22] Estándares de televisión CCIR (Comité Consultivo Internacional de Radiocomunicaciones - International Radio Consultative Committee) hoy UIT-R (Unión Internacional de Telecomunicaciones- Sector de Normalización de las Radiocomunicaciones)

[23] Técnicas de compresión de video. https://www.casadomo.com/images/CASADOMO/media/content/axis_tecnicas_de_compresion_de_video.pdf

JPEG (Joint Photograpic Experts Group) es un estándar diseñado para imágenes digitales estáticas; esté presenta variantes para la manipulación de imágenes en movimiento; el estándar JPEG-2000.
MPEG (Motion Picture Expert Group) es un estándar para la codificación de imágenes en movimiento y audio. Los estándares producidos son MPEG-1, MPEG-2, MPEG-4. Ambos estándares buscan reducir sobrecargas en los medios de transmisión teniendo resultados aceptables en las imágenes y vídeos generados. Estas técnicas son basadas en los siguientes criterios:

- Reducir matices de color en la imagen
- Reducir la resolución de color respecto a la intensidad de luz prevaleciente
- Reducir partes pequeñas, invisibles de la imagen
- En el caso de una secuencia de vídeo, las partes de una imagen que no cambian se dejan como están.

Todas estas técnicas están basadas en un conocimiento preciso y exhaustivo de cómo el cerebro y los ojos trabajan en combinación para formar el complejo sistema visual humano. Como resultado de estas sutiles modificaciones se produce una reducción significativa del tamaño del fichero para secuencias de vídeo sin prácticamente ningún efecto para la calidad visual. La posibilidad de que esas modificaciones sean apreciables por el ojo humano depende típicamente del grado de la técnica de compresión que se utilice[24].

Para comprender un poco el funcionamiento de estas técnicas de compresión, se genera un vídeo con tres imágenes. En la figura. Se tienen tres imágenes distintas. La figura 9.a utiliza la técnica JPEG. Mientras que en la Figura 9.b la técnica MPEG.

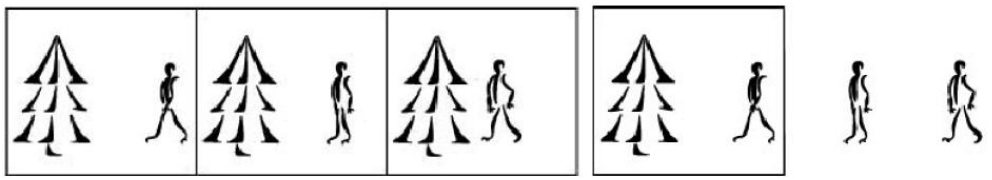

Figura 9. a) Técnica JPEG con tres imágenes estáticas. b) Técnica MPEG con tres imágenes estáticas.

La imagen de la izquierda es la primera imagen en la secuencia seguida por la imagen del medio y después la imagen de la derecha. Cuando se muestra, la secuencia de vídeo muestra a un hombre caminando de derecha a izquierda con un árbol que permanece estático. En las técnicas JPEG cada imagen de la secuencia se codifica como una única imagen separada ofreciendo como resultado una secuencia igual a la original.

[24]Técnicas de compresión de video.
https://www.casadomo.com/images/CASADOMO/media/content/axis_tecnicas_de_compresion_de_video.pdf

En MPEG video sólo las partes de la secuencia de vídeo se incluye junto con la información de las partes que ofrecen movimiento. Sin embargo esto es sólo real durante la transmisión de la secuencia de vídeo para limitar el consumo de ancho de banda. Cuando se visualice aparecerá nuevamente como la secuencia de vídeo original.
En las figuras se nota claramente cuál es el objetivo principal que buscan las técnicas de compresión. Existen otros estándares más evolucionados que describirlos todos estaría fuera del alcance de este documento. Lo que deberá quedar claro que las aplicaciones de vídeo vigilancia utilizan estas técnicas.

Además de todos los elementos utilizados en redes de vídeo vigilancia, se pueden poner a disposición de la red, sensores electrónicos, complementando al sistema de vídeo vigilancia, teniendo un sistema de seguridad en donde la red de video IP se complemente con estas redes de sensores y viceversa.

SENSORES.

Los sensores o detectores son dispositivos capaces de recoger información de los distintos parámetros que controlan (sensores de presencia, magnéticos, vibración etc…) y de transmitir esa información para su procesamiento[25].
Los tipos de sensores mayormente utilizados en un sistema de seguridad electrónica son:

- De movimiento: estos detectan la presencia de un intruso dentro de un área específica.
- De contacto comúnmente llamados magnéticos: detectan la apertura de puertas o ventanas.
- De quebradura de cristales: detectan la vibración y sonido que genera un cristal al romperse.
- De pánico: Dispositivos activados manualmente en una situación de pánico.
- Sirenas: Dispositivos de anunciación audible local para alertar de una posible intrusión.

Los sistemas de alarmas tienen dos formas de avisar al detectar la presencia de un intruso. Aviso local y aviso remoto.
El aviso local se realiza por medios sonoros al detectar algún sensor activado y alertar al personal de vigilancia que se encuentre cerca.
El aviso remoto se puede ser transmitido por distintos medios como línea telefónica, celular, radio e internet y avisar a una central de monitoreo de la existencia de una alarma. El sistema identificará la alarma, se identificará la zona y el tipo de sensor y el personal de monitoreo se encargará de tomar una decisión basándose en la información enviada por la alarma. Por ejemplo si se reporta una señal de apertura de puerta, el centro de monitoreo se encargará de llamar al personal de seguridad[26].

[25] Libro de domótica.
[26] Siscom.pdf

El monitoreo por internet, utiliza el protocolo TCP/IP (Transmission Control Protocol/Internet Protocol, Protocolo de Control de Transmisión/Protocolo de Internet), por lo tanto, es compatible con el estándar de comunicación IEEE 802.11; en esa línea, se puede adaptar fácilmente a un sistema de vídeo vigilancia IP.
Al igual que en redes de vídeo vigilancia, la instalación de los sensores puede ser cableada o inalámbrica. La ubicación de los sensores deberá ser en lugares estratégicos. En la figura se muestra alguna posible ubicación.

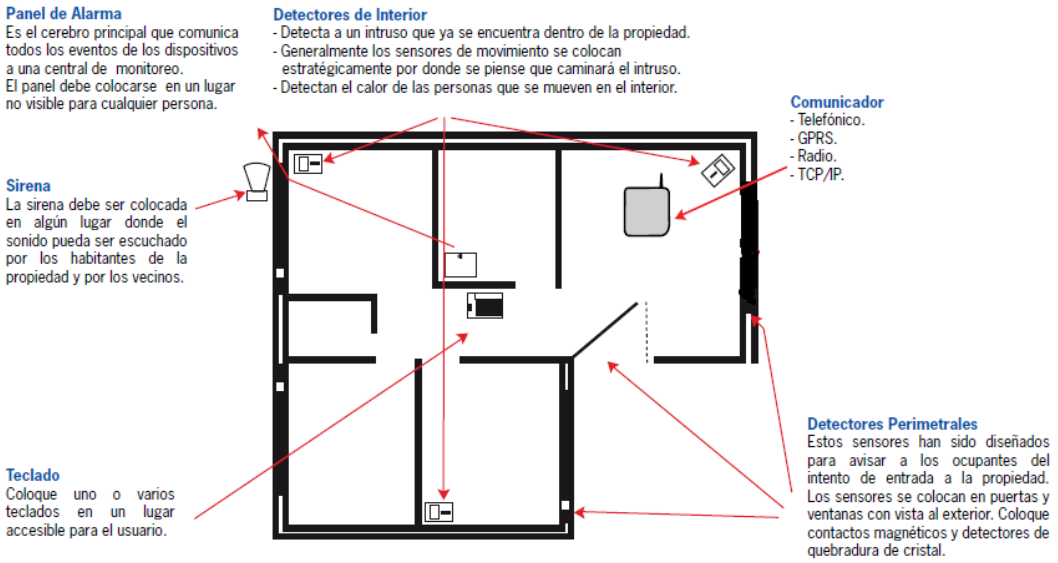

Figura 10. Ubicación de dispositivos de sistema de seguridad.

MÉTODOS DE TRANSPORTE DE DATOS[27].

Direcciones IP. Una dirección IP (dirección de Protocolo de internet) es un número exclusivo utilizado por los dispositivos para poder identificarse y comunicarse entre sí a través de una red utilizando el estándar de Protocolos de Internet. Una dirección IP está formada por cuatro números separados por un punto ".", cada número se encuentra en un rango de 0 a 255. Por ejemplo una dirección IP podría ser "192.168.1.92".
La dirección IP se divide posteriormente en una parte de red y una parte de host. El límite entre ambas partes se decide mediante una máscara de red o una longitud de prefijo. Una máscara de red de 255.255.255.0 significa que los tres primeros bytes corresponden a la dirección IP y el último byte corresponde a la dirección host. Una longitud de prefijo es una forma distinta de proporcionar el límite. Por ejemplo, la misma dirección IP que el ejemplo citado anteriormente posee una longitud de prefijo de 24 bits (p.ej., 192.168.1.92/24).

En direcciones IP se reservan algunos bloques de direcciones de uso privado:

[27]http://www.axis.com/es/products/video/about_networkvideo/data_transport_methods.htm

10.0.0.0/8 (máscara de red 255.0.0.0)
172.16.0.0/12 (máscara de red 255.240.0.0)
192.168.0.0/16 (máscara de red 255.255.0.0)

IPv6, o la versión 6 del Protocolo de Internet, ha sido diseñado como una actualización evolutiva del Protocolo de Internet y, de hecho, coexistirá con el antiguo IPv4 durante cierto tiempo. IPv6 ha sido diseñado para permitir que Internet crezca a un ritmo constante, tanto en términos del número de hosts conectados como de la cantidad total de tráfico de datos transmitidos.

La mejora más evidente de IPv6 respecto a IPv4 es que las direcciones IP se amplían de 32 bits a 128 bits. Esta ampliación anticipa el considerable crecimiento futuro de Internet, proporcionando un número ilimitado (a todos los efectos y propósitos) de redes y sistemas. Por ejemplo, IPv6 tiene previsto facilitar a cada teléfono móvil y dispositivo electrónico móvil su propia dirección.

Protocolos de transporte de datos para vídeo IP

El protocolo más habitual para transmitir datos en redes informáticas en la actualidad es el conjunto de protocolos TCP/IP. TCP/IP actúa de "portador" para muchos otros protocolos. Un buen ejemplo es HTTP (Protocolo de transferencia de hipertexto) empleado para navegar por páginas Web en servidores de todo el mundo a través de Internet.

Protocolos TCP/IP y puertos utilizados para el vídeo IP

Los protocolos habituales y sus números de puerto utilizados para la transferencia de vídeo IP incluyen, se muestran en la tabla 4

IP utiliza dos protocolos de transporte: Protocolo de Control de Transmisión (TCP) y el Protocolo de Datagramas de Usuario (UDP). TCP ofrece un canal de transmisión fiable basado en la conexión, y gestiona el proceso de convertir grandes bloques de datos en paquetes más pequeños, adecuados para la red física que se utiliza y garantiza que los datos enviados desde un extremo se reciben en el otro. UDP, por otro lado, es un protocolo sin conexión que no garantiza la entrega de los datos enviados, dejando así todo el mecanismo de control y comprobación de errores a cargo de la propia aplicación.

Protocolo	Protocolo de transporte	Puerto	Uso común	Uso vídeo en red
FTP File Transfer Protocol	TCP	21	Transferencia de ficheros a través de Internet/intranets	Transferencia de imágenes o vídeo desde una cámara de red o servidor de vídeo a un servidor FTP o a una aplicación
SMTP Send Mail Transfer Protocol	TCP	25	Protocolo para el envío de e-mails	Una cámara de red o servidor de vídeo puede enviar imágenes o notificaciones de alarma utilizando su cliente integrado de e-mail
HTTP Hyper Text Transfer Protocol	TCP	80	Utilizado para navegar en la web, p.e. para recibir páginas web de servidores web	El modo más común de transferencia de vídeo desde una cámara de red o servidor de vídeo donde el dispositivo trabaja como un servidor web, proporcionando vídeo al usuario o servidor de aplicación
HTTPS Hypertext Transfer Protocol over Secure Socket Layer	TCP	443	Utilizado para acceder a páginas web de forma segura utilizando encriptación	La transmisión de vídeo desde una cámara de red o servidor de vídeo puede ser utilizada para autenticar los envíos de la cámara utilizando certificados digitales X.509
RTP Real Time Protocol	UDP/TCP	No definido	Formato de paquetes estandarizado RTP para el envío de vídeo y audio a través de Internet. A menudo utilizado en sistemas multi-media o de video conferencia	Un modo común de transmitir vídeo en red MPEG La transmisión puede ser unicast (uno a uno) o multicast (uno a varios)
RTSP Real Time Streaming Protocol	TCP	554	Utilizado para configurar y controlar sesiones multimedia a través de RTP	

Tabla 4. Protocolos utilizados en internet.

En general, TCP se utiliza cuando se prefiere una comunicación fiable durante el tiempo de espera del transporte. La fiabilidad de TCP a través de la retransmisión puede producir retrasos significativos. Por otro lado, UDP no ofreceré transmisiones de datos perdidos y, en consecuencia, no produce mayores retrasos.

DISEÑO METODOLÓGICO DEL ESTUDIO.

INTRODUCCIÓN AL DISEÑO DEL SISTEMA DE SEGURIDAD.

Básicamente, y considerando todos los elementos tanto en software y hardware para implementar el sistema de seguridad basado en el protocolo de comunicación IEEE 802.11, se tienen los siguientes criterios de diseño.

Como punto de partida, y dadas las características de los sistemas de seguridad, se necesitan elegir, todos aquellos elementos que estarán a cargo del monitoreo continuo de los periodos de tiempo a lo largo del año en donde la afluencia de personas es mínima, estos periodos de tiempo ocurren en horarios no laborales, días de asueto para empleados y alumnos, fines de semana, vacaciones, eventos inesperados en los edificios; con el propósito de identificar aquellos eventos que representen algún indicio delincuencial en las áreas controladas. Estos elementos son los sensores magnéticos, de presencia o movimiento, de vibración y elementos sonoros como la sirena. Además, ante la presencia de una señal de alarma, se pueden tener dos casos qué pueden ser; el atentado delictivo como tal, o que trabajadores que, por alguna razón, estén dentro de la institución en los periodos de poca afluencia. En este último caso, se necesita un medio de identificación como teclado.

Por otra parte en las características de los sistemas de seguridad, el monitoreo continuo de las áreas destinadas a controlar, deberá ser provisto por sensores de imagen que poseen las cámaras digitales. Esto con el propósito de cubrir todo el tiempo posible y sin interrupción alguna las áreas vigiladas. Además, de elegir dispositivos que permitan agregar dinamismo a la cámara, con el objetivo fundamental de dar mayor capacidad de cobertura.

El siguiente punto a considerar, y utilizando las tecnologías desarrolladas en hardware libre, se necesita algún dispositivo capaz de comunicarse remotamente con servidores web y que sea fácilmente adaptable al protocolo de comunicación IEEE 802.11. Con el propósito de centralizar las señales eléctricas que estarán siendo emitidas por los sensores, así como la generación de señales eléctricas de aviso y control, utilizando técnicas de programación.

Que tenga la capacidad de funcionar las 24 horas del día, los 7 días de la semana. Además, elegir el elemento que estará siendo utilizado como servidor web, siempre manteniendo la condición de ser hardware de libre distribución y que sea capaz de permanecer en funcionamiento las 24 horas del día, los 7 días de la semana.

Para la elección del software necesario en el diseño, se deberá utilizar herramientas de libre distribución, es decir; todas las herramientas informáticas utilizadas, deben presentar la licencia free software; abarcando las clasificaciones de: software de sistema, software de programación y software de aplicación. Esto con el propósito específico de cumplir con los objetivos planteados en el estudio y diseño del sistema de seguridad. Para lograr con los propósitos fundamentales de la investigación, se tiene la libertad de elegir el software que presente mejor rendimiento en alguna función en particular del sistema.

Adicionalmente a los componentes en software y hardware, se debe diseñar una interfaz electrónica que permita acondicionar señales para los voltajes suministrado por los sensores y teclados, así como aquellos voltajes suministrados para los dispositivos dinámicos y elementos sonoros.

IMPLEMENTACIÓN EN HARDWARE Y SOFTWARE.

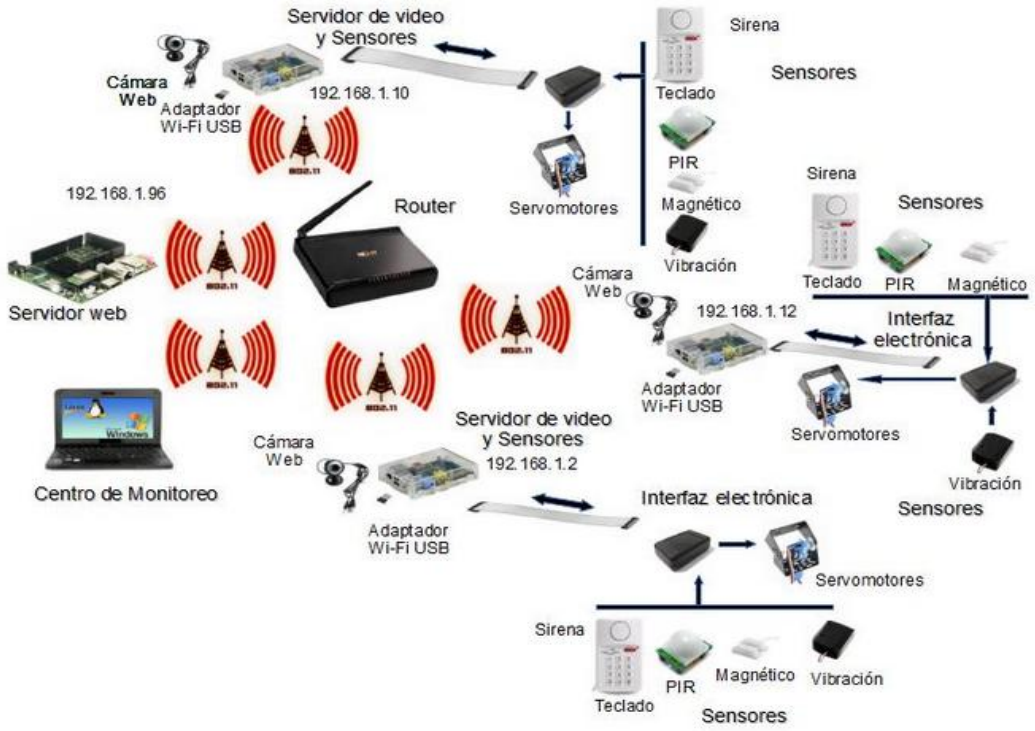

Figura 11. Esquema del Sistema de Seguridad.

El diseño a implementar estará formado por tres Computadoras Raspberry PI que representan los puntos de Vigilancia, cada punto de vigilancia interconecta diferente tipos de sensores que envían información de su estado a la Raspberry PI, esta información se envía través de sus puertos GPIO utilizando una etapa previa de acondicionamiento de

señal para poder trabajar con los niveles de corriente y voltaje aceptados por los puertos GPIO. Esta etapa de acondicionamiento de señal está formada por circuitos opto acopladores que aíslan un nivel de tensión de otro nivel de tensión menor aceptado por la Raspberry PI. Los puertos GPIO al ser de propósito general, deberán ser administrados de manera eficiente para evitar que las señales tengan configuraciones erróneas. Por esa razón sé utiliza el lenguaje de programación Python que se encarga de administrar los puertos, el primer paso es colocar los puertos GPIO que leerán el estado de los sensores como entradas y luego, con una resistencia Pull-down programada y administrada por el lenguaje para que este detecte los niveles de voltaje alto; esta parte, que configura los puertos como entrada y poder leerlos se realiza mediante las funciones de las librerías RPI GPIO y SYS pertenecientes a Python, luego la información leída del sensor se envía a través de la red utilizando el método POST de las funciones propias que contiene la Librería URLLIB2 de Python. La información es enviada al servidor web apache instalado y configurado en la computadora UDOO la cual tiene, además, el Interprete del Lenguaje PHP y Mysql como gestor de Bases de datos, en donde los datos son capturados y procesados por archivos que contienen funciones en lenguaje PHP; estos archivos se encargan de filtrar los datos e interactuar con la base de datos diseñada para este sistema.

Dentro de las tablas existentes en la base de datos, existe una tabla para cada edificio que tenga un punto de vigilancia y donde la tupla de estas tablas esté representada por un sensor que tiene asignado un ID único, que son:

- El estado de su bandera actual.
- El tipo de sensor.
- La zona y su descripción.

La idea del sistema es que el estado de los sensores reales se esté monitoreando todo el tiempo y que el sistema cambie el campo bandera de las tupla que representa los sensores de las tablas de los edificios que, dentro de la semántica de la base de datos, estas tuplas mantengan el estado correspondiente a la señal que tenga el sensor real en ese instante de tiempo, dichos cambios tienen que hacerse de manera automática, siempre y cuando el filtro de la configuración de los horarios establecidos como válidos permita escribir la base de datos.

La información captada por la base de datos que constantemente está cambiando servirá para ser mostrada a través de un centro de monitoreo; el cual, está formado por un conjunto de páginas escritas en PHP, JavaScript, CSS, XML y HTML y en donde cada una tiene una función específica. Es por eso que se utiliza la técnica de programación AJAX (JavaScript And XML por sus siglas en ingles) para hacer que el sistema Cliente-Servidor se asemeje lo más que pueda a una aplicación de escritorio; ya que al elegir máquinas

embebidas de Bajo Costo como la Raspberry PI y UDOO; se debe tratar de optimizar el recurso lo mejor posible. En este caso HTML sirve para crear las páginas web de tal manera que con elementos básicos se pueda mostrar la información necesaria, pero como es del conocimiento, estos elementos deberán mostrarse con una buena presentación para el usuario y además deben de ser capaces de ser interpretados por cualquier navegador y en diferentes resoluciones posibles, es por eso que las páginas web utilizadas, tienen aplicado Hojas de Estilo en Cascada (CSS) para lograr dar una presentación aceptable. En cuanto a la funciones disponibles para los Usuarios del centro de Monitoreo, JavaScript hace peticiones asíncronas con el servidor de tal manera que estas peticiones se ejecutan en segundo plano sin interferir con la interfaz que en el momento que este siendo utilizada por el usuario, pero también que permita enviar solo la información necesaria evitando la saturación de la red con información innecesaria, estos datos que van a ser en segundo plano, se envían por el método GET o POST y luego son procesados por funciones en PHP interactuando estos con la base de datos y regresando estos datos en formato XML el cual permite clasificar y estructurar la información en forma ordenada y clara para la programación que se está realizando.

También hay que decir que cada punto cuenta con una cámara web conectada a la Raspberry pi la cual es administrada por el servidor de video ZoneMinder, el cual pone a disposición de los usuarios la video vigilancia en cada uno de los puntos requeridos. Estas cámaras tienen movimiento y es realizado por servo motores; estos servo motores reciben señales PWM que le indican la posición de la cámara web, cada cámara tiene dos servo motores que permiten movimientos TILT(movimiento vertical) y PAN(movimiento horizontal).

Los servo motores son controlados de forma remota por medio JQUERY de tal manera que desde el centro de monitoreo se pueda mover la cámara con la posición que se necesite, desde el centro de monitoreo, la aplicación que permite el movimiento de los servomotores se llama PI Blaster.

MANEJO DE LOS PUERTOS GPIO DE LA RASPBERRY.

El uso de las computadoras se ha desarrollado a gran escala en nuestros tiempos dado que gran cantidad de trabajo se resume en solo una computadora, que actualmente se ha hecho un instrumento básico en todas las áreas de desarrollo de la humanidad (laboral, entretenimiento, educación, etc.).

En la actualidad existen varias tarjetas para el desarrollo de proyectos de aplicación tecnológicas; una de ellas es la tarjeta Raspberry pi desarrollada en Reino Unido por la fundación Raspberry Pi es básicamente un ordenador creado con el objetivo de estimular la enseñanza de ciencias computacionales en las escuelas.

PUERTOS GPIO

El puerto GPIO de la Raspberry Pi está localizado en la parte superior izquierda de la tarjeta de circuito, etiquetada como P1 tal como se muestra en la figura 12. Este es un puerto de 26 pines, el cual está dispuesto en dos filas de 13 pines cada una de tipo macho de 2.54 mm de largo de fábrica. La longitud de 2.54 mm de estos pines es importante por que es muy común ver esa longitud de los pines en electrónica, y este es el espacio estándar para placas de prototipo.

Cada pin del puerto GPIO tiene su propósito específico, con varios pines trabajando juntos se pueden formar circuitos particulares. Esos 26 pines tienen funciones definidas para SPI o I2C, sin embargo, estos pueden ser reconfigurados para que todos sean puertos digitales; además se pueden utilizar 17 de estos pines y configurar cada uno de ellos ya sea como puerto de entrada o de salida. Los 9 pines restantes son pines que entregan voltaje ya sea a 3.3V (2 pines), 5V (2 pines) o conexión a tierra (5 pines).

Los números de pines para la GPIO son divididos en dos filas, en la que la fila inferior toma los números impares y la superior los números pares. Esto es muy importante tener en cuenta al momento de trabajar con los puertos GPIO de la Raspberry Pi porque la mayor parte de los dispositivos usa un sistema diferente de numeración de pines. Además; estos no están marcados sobre la tarjeta lo que puede causar confusión.

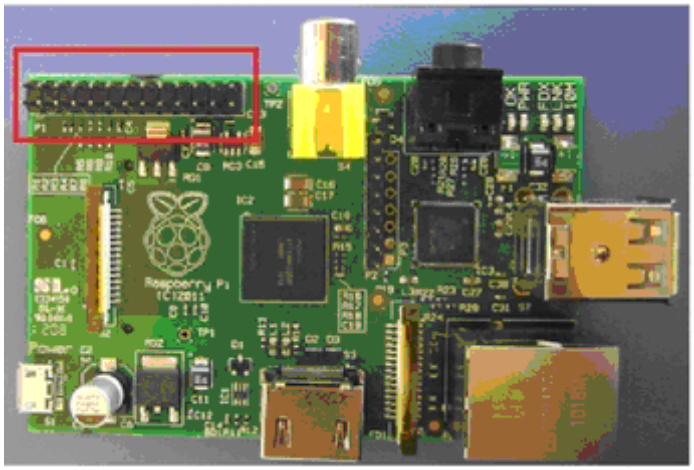

Figura 12. Localización de los puertos GPIO.

Aunque la raspberry Pi proporciona una fuente de alimentación, por medio de la fuente de alimentación de la placa, en el conector micro-USB, en el pin 2, internamente la Raspberry Pi trabaja con 3.3 V como voltaje lógico. Esto significa que las lecturas de entradas y salidas accionadas por los diferentes sensores utilizados, deben trabajar con este nivel de voltaje y si no, hay que acondicionar las señales; debido a que los puertos GPIO no son tolerantes a voltajes mayores.

El puerto GPIO proporciona 23 pines; cada uno con funciones específicas como se muestra en la figura 13.

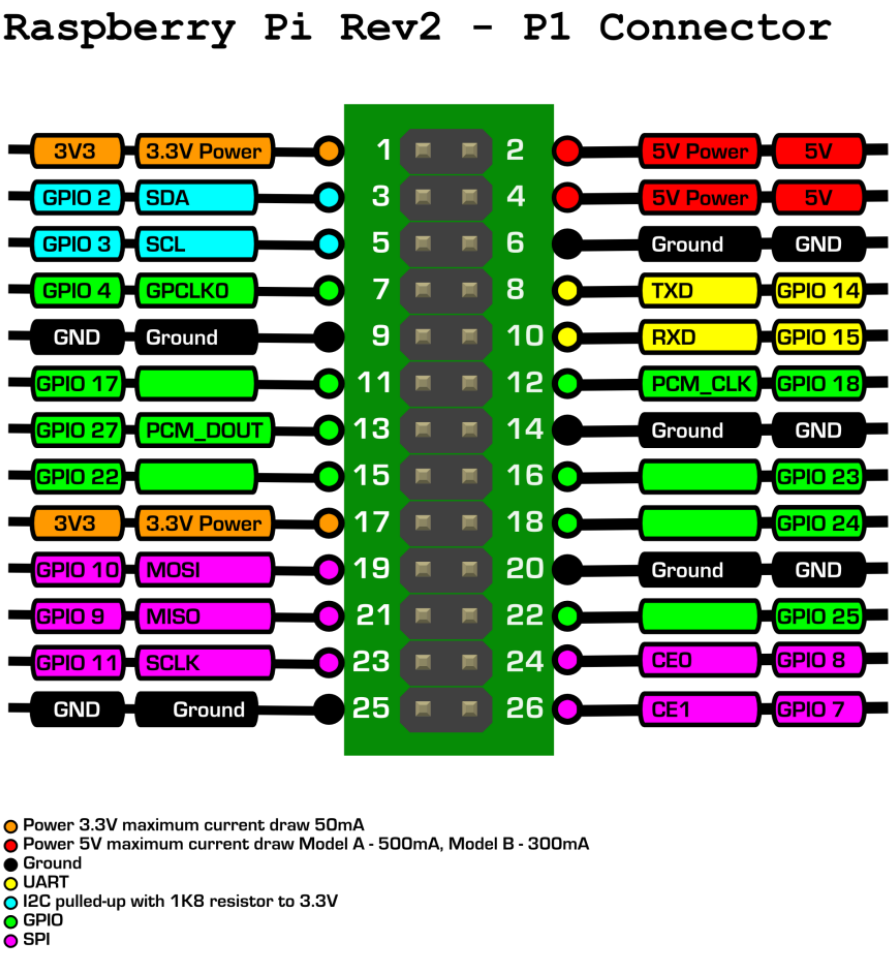

Figura 13. Descripción de los pines GPIO REV.2.

En el presente proyecto se han utilizado los siguientes pines:

Número del pin GPIO	Descripción y Utilidad
1	Salida de alimentación de 3.3 voltios
6	Conexión a tierra
7	Salida para activación de la sirena
9	Conexión a tierra
11	Teclado de activación y desactivación del sistema
13	Sensor para detección de violación de ventanas
15	Sensor magnético para puerta de acceso.
16	Salida PWM para control de servo motor
17	Salida de alimentación de 3.3 voltios
18	Salida PWM para control de servo motor
19	Sensor de movimiento
21	Sensor de movimiento

Tabla 5. Números de los pines utilizados en el sistema de seguridad.

EL SOFTWARE UTILIZADO

Para utilizar y manejar los puertos anteriormente descritos, se ha utilizado el lenguaje Python; puesto que este es un lenguaje que facilita mucho el trabajo con los GPIO, además viene como parte del sistema operativo utilizado en la Raspberry PI; por lo que no es necesario instalar software adicional y basta con actualizar la versión de python e instalar el módulo RPi.GPIO, desde la línea de comando, como se muestra en la figura 14:

```
1. $sudo apt-get update
2. $sudo apt-get install python-dev
3. $sudo apt-get install python-rpi.gpio
```

Figura 14. Instalación de librerías de Python.

Este módulo de Python, ofrece varias funcionalidades que facilita mucho el proceso de lectura y escritura de cada uno de los puertos y solo es necesario importarlo junto con los demás módulos al principio del programa, como se muestra en la figura 15.

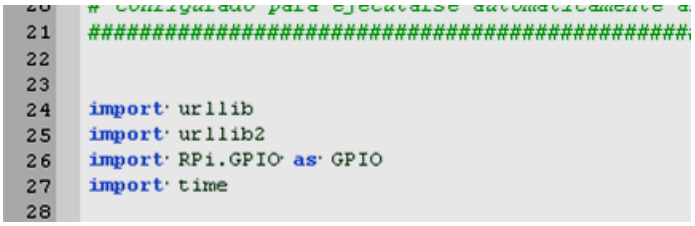

Figura 15. Importación de los módulos de python necesarios para el sistema.

En la línea 26 se hace la importación de este módulo, con lo cual se está listo para comenzar a trabajar sobre los GPIO. Las líneas 24 y 25, se importan dos librerías muy importantes en el script; dado que el sistema trabaja junto con una base de datos llamada "sistemavigilancia"; que se encuentra en el servidor MySQL. Al importar las librerías urllib y urllib2, se tiene acceso a los métodos con los que se pueden insertar los datos y registros en la base de datos.

En la línea 27, se llama el módulo time; porque en el programa se utilizan algunos métodos y funciones de tiempo.

Luego, se debe elegir la forma de identificar los pines, debido a que existen básicamente dos maneras de reconocerlos dentro del script de python; la primera es la forma BOARD, en la cual se reconoce la numeración de los pines según se numeran en la placa de la Raspberry Pi, es decir el conector de dos filas de 26 pines, con los pines impares a la izquierda y los pares a la derecha. El otro modo es el BCM, en este modo los pines se identifican por el número del puerto, según lo reconoce le procesador, tal como se muestra en la figura 16.

Figura 16. Índice de pines de la Raspberry Pi.

USO DE INTERRUPCIONES

La última actualización en el mundo de la Raspberry Pi y la programación de los puertos GPIO con Python, es el realizado por Ben Croston; el cual ha realizado una mejora al módulo RPi.GPIO, a partir de la versión 0.5.1. Esta mejora introduce el uso de interrupciones.

Una interrupción es precisamente detener la ejecución del programa, es esperar a que ocurra algún evento sin la necesidad de estar verificando constantemente si el evento paso o no.

Entonces, para el sistema de seguridad resulto adecuado; en lugar de estar viendo la activación de algunos de los sensores conectado a los puertos GPIO por medio de un lazo WHILE infinito; se utiliza esta nueva característica de las interrupciones del módulo RPi.GPIO, de modo que el programa se mantiene en forma de stand-by, esperando a que el evento de la activación de un sensor por la intrusión de una persona a una zona protegida, dispare la respectiva interrupción y se realice el bloque de código de programa respectivo.

Este cambio estructural del programa permite mayor rapidez de reacción, por que utiliza muy pocos recursos del procesador de la computadora Raspberry Pi para la lectura de los GPIO y enfocar el uso del CPU a otras tareas; por ejemplo en el caso, la transmisión de video.

CONFIGURACIÓN PULL-DOWN O PULL-UP DE LOS PUERTOS GPIO[28]

Una ventaja que ofrece Python y el módulo RPi.GPIO, es la capacidad de declarar y configurar cada uno de los GPIO como PULL-DOWN o PULL-UP, lo que facilita el manejo de los pines y la circuitería extra necesaria para la conexión de los sensores a los puertos.

Cuando un puerto GPIO es declarado como PULL-UP, es equivalente a haberle conectado una resistencia entre VCC y el pin respectivo de ese puerto; es decir que vía software se puede establecer que ese puerto mantendrá un valor lógico alto (True ó 1); hasta que se reciba una entrada que ponga a tierra (False ó 0) en ese puerto; por ejemplo por la activación de un sensor que coloque es puerto a nivel lógico bajo.

De manera similar se puede configurar cualquier puerto, como PULL-DOWN; lo que equivale a haber conectado una resistencia entre el puerto respetivo y tierra (GND).

Nuevamente, es decir que por medio de software se establece que ese puerto mantendrá un valor lógico bajo (False ó 0); hasta que ese puerto reciba una entrada que lo ponga a nivel lógico alto.

En las líneas 33 a 40 del programa, se declara la forma en la que cada uno de los puertos utilizados para los sensores será tratado por el programa Monitoreo4.py. Aquí se configuran los puertos como PULL-DOWN, como se muestra en la figura 17

```
28
29  ################################################################################
30  # Configuracion de los puertos GPIO como entradas opcion pull_down
31  ################################################################################
32
33  GPIO.setmode(GPIO.BOARD)   # configuramos el modo de los pines en modo BOARD
34  GPIO.setwarnings(False)    # Desactivamos los wanings
35  GPIO.setup(5, GPIO.IN, pull_up_down=GPIO.PUD_DOWN)    # salir del programa
36  GPIO.setup(19, GPIO.IN, pull_up_down=GPIO.PUD_DOWN)   # PIR
37  GPIO.setup(15, GPIO.IN, pull_up_down=GPIO.PUD_UP)     # MAGNETICO
38  GPIO.setup(13, GPIO.IN, pull_up_down=GPIO.PUD_DOWN)   # VENTANA
39  GPIO.setup(11, GPIO.IN, pull_up_down=GPIO.PUD_DOWN)   # TECLADO
40  GPIO.setup(7, GPIO.OUT, pull_up_down=GPIO.PUD_DOWN)   # SIRENA
41
```

Figura 17. Configuración de los puertos como PULL-DOWN

En la figura 18, se muestra parte del código Python, en las que se establece las respectivas interrupciones para cada puerto. Como se mencionó, la interrupción espera a que ocurra el evento para disparar la interrupción. En las líneas 179 a 182 del programa se aplica el método GPIO.add_event_detect, del módulo RPi.GPIO, el cual dispara la interrupción. Los parámetros que recibe este método son, el número del Pin, la forma de disparo del

[28] http://sourceforge.net/p/raspberry-gpio-python/wiki/Inputs/

evento; las cuales pueden ser RISING, es decir el evento a detectar es el flanco de subida, cuando el puerto cambia de nivel lógico de Bajo a Alto (0 → 1); puede ser FALLING; el cual es detectar el flanco de bajada, cuando el puerto cambia de nivel lógico de Alto a Bajo (1 → 0) ó puede ser BOTH, es decir detectar ambos flancos tanto el de subida como el de bajada.

```
175 ################################################################################
176 # Configuracion de las Interrupciones en los pines de loa sensores
177 ################################################################################
178
179 GPIO.add_event_detect(19, GPIO.RISING, callback=SENSOR_1, bouncetime=1200)
180 GPIO.add_event_detect(15, GPIO.RISING, callback=SENSOR_2, bouncetime=1200)
181 GPIO.add_event_detect(13, GPIO.RISING, callback=SENSOR_3, bouncetime=1200)
182 GPIO.add_event_detect(11, GPIO.RISING, callback=RESET, bouncetime=12000)
183
```

Figura 18. Declaraciones de las Interrupciones para cada puerto

Otro parámetro de este método es la llamada a la función específica una vez sea disparada la interrupción; en este caso el CALLBACK se hace a las funciones SENSOR_1, SENSOR_2 o SENSOR_3, dependiendo de cuál sensor se ha activado y disparado la señal alarma. En estas funciones se define el código del programa a ejecutar para el tratamiento de las alarmas y finalmente se tiene el parámetro BOUNCETIME por medio del cual se puede configurar el tiempo expresado en milisegundos en los que la interrupción y la detección del evento ignorará el rebote de señal que usualmente, suele estar presente en los puertos GPIO al trabajar con sensores o botones que producen rebote de las señales enviadas a los pines, lo que introduce falsas lecturas en los datos leídos por los puertos GPIO y en este caso falsas alarmas.

DECLARACION DE VARIABLES

Entre las líneas 46 a 76 de la figura 19, se definen las variables utilizadas en el script. Las variables tiempo_anterior y reset son variables globales que podrán ser modificadas por las funciones del programa, la variable retardo establece el tiempo que transcurre entre el envío de la bandera naranja y la bandera roja a la base de datos; las variables control se inicializan en cero y son utilizadas en el ámbito particular de la función respectiva.

La variable ip en la línea 62, guarda la dirección IP del servidor en donde se encuentra la base de datos “sistemavigilancia” y finalmente, la variable edificio debe ser seleccionada y declarada, al descomentar la línea correspondiente al edificio donde se instalará el punto de vigilancia controlado por ésta Raspberry Pi. Como se muestra en la figura 19, se ha

descomentado la línea 66 y por lo tanto éste punto de seguridad queda asignado al edificio de eléctrica; además junto con la variable edificio, se designan igualmente las variables sen1, sen2 y sen3 con el valor correspondiente al identificador respectivo para cada uno de los sensores, tal y como se encuentran en la base de datos.

LLAMADA DE LAS FUNCIONES (CALLBACK)

Una vez detectado el evento que dispara la interrupción, ésta hace un llamado a una función específica para cada una de las interrupciones, como muestra la figura 21. Para este fin se han escrito tres funciones llamadas sensor_1, sensor_2 y sensor_3, las cuales ejecutan básicamente el mismo bloque de instrucciones, cambiando únicamente el registro que se envía a la base de datos.

```
42  ################################################################################
43  # DECLARACION Y ASIGNACION DE VARIABLES AL INICIO
44  ################################################################################
45
46  tiempo_anterior = time.time()      # Establece la variable con el tiempo actual
47  retardo = 90                       # tiempo entre bandera NARANJA y ROJA
48  reset = 1
49  control1 = 1
50  control2 = 1
51  control3 = 1
52  control4 = 0
53
54  ################################################################################
55  # DIRECCION IP DEL SERVIDOR Y NOMBRE DEL EDIFICIO
56  # desmarcar la linea correspondiente al edificio que protejera este punto.
57  # Es necesario escribir en cada funcion el numero identificador de cada sensor
58  # tal y como se encuentre en la base de datos, para que los registros los
59  # guarde apropiadamente en cada zona.
60  ################################################################################
61
62  ip = "http://192.168.1.4/tesis2/prueba.php"
63
64  ################################################################################
65
66  edificio = "electrica", sen1 = "001", sen2 = "002", sen3 = "003"
67  #edificio = "administracionfia", sen1 = "022", sen2 = "023", sen3 = "024"
68  #edificio = "bibliotecafia", sen1 = "029", sen2 = "030", sen7 = "031"
69  #edificio = "medicina", sen1 = "071", sen2 = "072", sen3 = "073"
70  #edificio = "complejodepotivo", sen1 = "064", sen2 = "065", sen3 = "066"
71  #edificio = "oficinascentrales", sen1 = "050", sen2 = "051", sen3 = "052"
72  #edificio = "potencia", sen1 = "078", sen2 = "079", sen3 = "080"
73  #edificio = "cian", sen1 = "015", sen2 = "016", sen3 = "017"
74  #edificio = "ciensalud", sen1 = "043", sen2 = "044", sen3 = "045"
75  #edificio = "bibliotecacentral", sen1 = "036", sen2 = "037", sen3 = "038"
76  #edificio = "libreriaues", sen1 = "057", sen2 = "058", sen3 = "059"
```

Figura 19. Declaración de las variables utilizadas en el programa.

En la línea 93 se define la función con el nombre de SENSOR_1, a continuación se declaran dos variables globales importantes, la primera tiempo_anterior, la que se utiliza para reducir el efecto de rebote en las lecturas de los puertos por medio de comandos y sentencias de python y las variables de reset y control1 que se utilizan para condicionar el envío del registro con bandera ROJA. Es imprescindible declarar estas variables como globales, puesto que estas serán modificadas dentro de la función y de no declararlas así, se producirá un error.

En la línea 95, se establece una nueva variable, esta vez de tipo local llamada tiempo_actual en la que se guarda el tiempo tomado del sistema por medio de la función time.time() perteneciente al módulo time importado al principio.

La línea 96 muestra la primera decisión que se estableció como ya se mencionó para reducir el rebote de las señales en las lecturas de los puertos; si la condición por la cual la diferencia entre el tiempo_actual y el tiempo_anterios es >= 1, se ejecutarán las líneas de código 97 a la 100; además incluye la variable control1 dentro de la condición debido a la necesidad de no repetir el envío del registro con la bandera naranja, una vez se haya enviado la bandera roja.

En la línea 98 se llama la función enviar; la cual se define en la línea 83, como se muestra en la figura 20.

```
78  ################################################################################
79  #   FUNCION ENVIAR REGISTROS
80  ################################################################################
81
82
83  def enviar(flag, iden):
84      dato = urllib.urlencode({"bandera": flag, "edificio": edificio, "identificador": iden})
85      respuesta = urllib2.urlopen(ip, dato)
86      print (respuesta.read())
87
```

Figura 20. Definición de la función enviar.

Con esta función se realiza el envío del registro a la base de datos ubicada en la IP asignada anteriormente, además envía el nombre de la tabla que en este caso es el nombre de la variable edificio.

En esta función enviar(), se establece la variable local llamada dato, el cual es codificado al formato de envío URL de forma que pueda ser reconocido y utilizado por el script de comprobar.php; esto se realiza por el método urllib.urlencode().

En esta variable llamada dato se colocan las tuplas de registro de datos pertinentes que se necesitan guardar en la base de datos “sistemavigilancia” de MySQL, estas tuplas de datos son:

- La bandera y su color
- Edificio y su nombre.
- Identificador y el número del sensor activado.

Luego en la línea 85 se abre la dirección URL por medio de la sentencia urllib2.urlopen(); función que recibe dos parámetros, en este caso la dirección IP del servidor en donde se encuentra la base de datos junto con el nombre del archivo que recibirá el valor del dato, el cual se pasa como segundo argumento para la función.

En la línea 100 del programa, dentro de la función sensor_1, se utiliza el retardo de tiempo que debe existir entre el envío de la bandera naranja que es la primera prevención de alarma si alguno de los sensores se activa por cualquier razón. Si después de transcurridos la cantidad de segundos almacenados en la variable retardo de la línea 47, luego del envío de la bandera naranja y no se registra la señal de reset procedente desde el teclado numérico para la desactivación de las alarmas; entonces se envía un nuevo juego de registros, esta vez con el color de bandera ROJO, el cual disparará la alarma y activará la sirena como forma de aviso al personal de seguridad de la Universidad de El Salvado. Este proceso se establece en las líneas 102 y 103 de la figura 21.

```
88  ################################################################################
89  # FUNCION DEL SENSOR 1  ---  PIR
90  ################################################################################
91
92
93  def SENSOR_1(channel):
94      global tiempo_anterior, reset, control1
95      tiempo_actual = time.time()
96      if (((tiempo_actual - tiempo_anterior) >= 1) and (control1 == 0)):
97          tiempo_anterior = tiempo_actual
98          enviar("naranja", sen1)
99          print ("Sensor 1 con bandera.......NARANJA")
100         time.sleep(retardo)  # Retardo de tiempo entre bandera naranja y roja
101         if(reset == 0):
102             enviar("rojo", sen1)
103             alarma()  # Activamos la alarma sonora
104             print ("Sensor 1 con bandera.......ROJA")
105             control1 = 1
106
```

Figura 21. Función llamada por CALLBACK al detectar la interrupción.

Las otras dos funciones SENSOR_2 y SENSOR_3 son una copia de la función SENSOR_1 descrita en el párrafo anterior, y la única diferencia estriba en el dato que se envía como registro a la base de datos, en el cual se modifica el identificador del sensor por el número respectivo del sensor 2 ó del sensor 3. Por lo demás, estas funciones realizan exactamente lo mismo.

La función de RESET mostrada en la figura 22, sí varía respeto a las funciones de los sensores, debido a que esta función es llamada al detectarse la señal proveniente desde el teclado numérico hasta el pin 11 de la Raspberry Pi y es la que debe desactivar las alarmas registradas en ese punto de vigilancia. Esto significa colocar todas las banderas en color verde; lo que indica que los sensores son desactivados en ese momento.

```
145  ################################################################################
146  # LECTURA DEL TECLADO DE RESET
147  ################################################################################
148
149
150  def RESET(channel):
151      global reset, control1, control2, control3, control4
152      if (reset == 0):
153          (reset, control1, control2, control3, control4) = (1, 1, 1, 1, 0)
154      else:
155          (reset, control1, control2, control3, control4) = (0, 0, 0, 0, 1)
156      if (control4 == 0):
157          print ("senal de RESET detectada")
158          enviar("verde", sen1)
159          enviar("verde", sen2)
160          enviar("verde", sen3)
161          print ("Todas las alarmas ya han sido desactivadas... SISTEMA DESARMADO \n")
162      print ((("reset ="), reset, (" control4 ="), control4))
163
```

Figura 22. Función RESET para desactivar las alarmas.

Para esta función se envía la tupla de registros con la bandera verde como se muestra en las líneas 158, 159 y 160 de la figura anterior. En esta ocasión se deben enviar tres datos, uno por cada sensor conectado a la placa Raspberry, dado que cada una tiene un identificador diferente, en este ejemplo son el 001, 002 y 003. De haber más sensores conectados, habrá que incluirlos.

Finalmente se define la función alarma, la cual enviará un pulso de activación para que suene la sirena que se encuentra instalada junto con el teclado. Basta con enviar un pulso de medio segundo para activar la sirena. La sirena solamente podrá ser desactivada introduciendo el código correcto en el teclado numérico. La función queda definida entre las líneas 180 a 184, como lo muestra la figura 23.

```
175  ######################################################################
176  # SALIDA Y ACTIVACION DE LA SIRENA
177  ######################################################################
178
179
180  def alarma():
181      GPIO.output(7, GPIO.HIGH)
182      time.sleep(0.3)
183      GPIO.output(7, GPIO.LOW)
184      return
185
```

Figura 23. Definición de la función para activar la salida de alarma.

GENERAR PAGINA DE VIDEO.

Para generar la página de video. Luego de haber hecho las configuraciones respectivas de zoneminder, solamente se necesita del streaming de video el div que proporciona el servidor zoneminder.

```
<div id="imageFeed">
<img id="liveStream" src="http://192.168.1.8/cgi-bin/nph-zms?mode=jpeg&amp;monitor=1&amp;scale=100&
amp;maxfps=5&amp;buffer=1000&amp;user=admin&amp;connkey=549855&amp;rand=1401837945" alt="PlantaBaja
" width="320" height="240"/>
</div>
```

Figura 24. Etiqueta DIV en HTML que contiene el DIV de transmisión Video.

Del código de la figura 24 se observa que se tienen dos div. Uno con id = “imageFeed” y otro con id = “liveStream”, básicamente el div de interés es liveStream debido a que este div va pasando las imágenes que se obtienen del demonio nph-zms. Dependiendo del tipo de opción así será la configuración que deberá tener el id “liveStream”.

Este archivo se guarda con extensión .HTML el resultado se muestra en la figura 25.

Figura 25. Captura de transmisión de video.

CONTROL HORIZONTAL Y VERTICAL DE LA APLICACIÓN.

Las cámaras que se utilizan para sistemas de seguridad además de la resolución; vienen diseñadas para ser fijas o dinámicas; estas últimas vienen provistas de 3 movimientos que en los sistemas comerciales se les denomina como: PAN (movimiento horizontal), TILT (movimiento vertical) y ZOOM (acercamiento) respectivamente. Y se encuentran en las especificaciones de los fabricantes como PTZ en las hojas de datos.

Al sistema de seguridad de software y hardware libre, se le dotará, al control, las características de Pan y Tilt, puesto que son cámaras webs las que están siendo configuradas como cámaras de seguridad. Dicho mecanismo debe de ser diseñado y construido de manera eficiente y funcional con hardware de bajo costo y software libre.

El sistema consta de 2 servomotores que están montados sobre Brackets[29] que permiten el movimiento vertical y horizontal. Dicho control está siendo manejado remotamente desde la página web de la aplicación con elementos visuales sliders, cada slider está siendo programado para que la cámara se mueva horizontalmente o verticalmente en correspondencia con los servomotores; es decir, un servomotor tendrá la función de moverse de manera horizontal, y el otro servomotor lo hará de manera vertical.

Para darle al sistema el comportamiento de pant y tilt, se utiliza un proyecto de libre distribución que habilita los puertos GPIO de la raspberry pi como salidas para señales PWM, llamado pi-blaster30.

Pi-blaster habilita 8 puertos GPIO como señales PWM si no se hace ninguna modificación al archivo fuente. Particularmente para el sistema de seguridad, se necesitas 2 pines para ser configurados

GPIO number	Pin in P1 header
4	P1-7
17	P1-11
18	P1-12
21	P1-13
22	P1-15
23	P1-16
24	P1-18
25	P1-22

Figura 26. Lista de pines GPIO configurables como salidas PWM.

[29] Pieza mecánica en donde se montan los servomotores para permitir movimiento.
[30] https://github.com/sarfata/pi-blaster

Para el caso del control, solamente se necesitan 2 pines del puerto GPIO. Para el caso particular del sistema de seguridad, se usarán los números de GPIO 23(pin 16) y GPIO 24 (pin 18). Estos pines serán los encargados de generar las señales PWM que moverán al motor hacia una ubicación específica dentro del rango de grados a las que giran los motores. Dichos motores tienen un rango de giro de aproximadamente 0 a 270 grados.

Pi-blaster es un software diseñado para generar señales PWM con una frecuencia específica. Este software tiene como característica principal, generar señales PWM en donde se puede modificar su ciclo de trabajo desde un porcentaje bien pequeño como puede ser 1% hasta el 100% del ciclo de trabajo.

Para configurar pi-blaster correctamente, antes de compilarlo, se hace una pequeña modificación al archivo fuente pi-blaster.c de la figura 27.

```
38
39  // Created new known_pins with raspberry pi list of pins
40  // to compare against the param received.
41  static uint8_t known_pins[] = {
42          4,      // P1-7
43          17,     // P1-11
44          18,     // P1-12
45  #if REVISION == 2
46          27,     // P1-13
47  #else
48          21,     // P1-13
49  #endif
50          22,     // P1-15
51          23,     // P1-16
52          24,     // P1-18
53          25,     // P1-22
54  };
55
56  // pin2gpio array is not setup as empty to avoid locking all GPIO
57  // inputs as PWM, they are set on the fly by the pin param passed.
58  static uint8_t pin2gpio[8];
59
```

Figura 27. Código fuente de Pi-blaster.

En dicho archivo existen 2 vectores para los pines. El arreglo uint8_t known_pins y el arreglo static uint8_t_pin2gpio. Estas variables configuran los 8 pines sin hacerles ninguna modificación al código fuente de la figura 27. Para lograr configurar solamente 2 de ellos, basta con dejar dentro del vector known_pins los pines necesarios.

En la variable vector de la figura 27 uint8_t pin2gpio, se elimina el índice del vector y se iguala a la variable uint8_t pin2gpio. Ambos vectores deberán quedar iguales como se muestra en la figura 28.

```
39  // Created new known_pins with raspberry pi list of pins
40  // to compare against the param received.
41  static uint8_t known_pins[] = {
42          23,     // P1-16
43          24,     // P1-18
44  };
45
46  // pin2gpio array is not setup as empty to avoid locking all GPIO
47  // inputs as PWM, they are set on the fly by the pin param passed.
48  static uint8_t pin2gpio[] = {
49          23,     // P1-16
50          24,     // P1-18
51  };
52
```

Figura 28. Modificación al código fuente de Pi-blaster.

Al modificar el archivo fuente exactamente como se describe en la figura 28, solamente los pines 16 y 18 del pinout de los puertos serán señales PWM.

Para instalar el software, desde la Shell de Raspbian, o algún otro sistema operativo basado en debían, se ejecuta la sentencia descrita en la figura 29.

```
sudo apt-get install autoconf
```

Figura 29. Instalación de paquete autoconf.

Luego, siempre desde la Shell se pasa a construir y configurar, basta con escribir las siguientes líneas, como las presentadas en la figura 30.

```
./autogen.sh
./configure
make
```

Figura 30. Configuración desde la terminal.

Y luego se instala con la sentencia descrita en la figura 31.

```
sudo make install
```

Figura 31. Instalación de paquete.

Para arrancar pi-blaster manualmente se ejecuta la sentencia de la figura 32.

```
sudo ./pi-blaster
```

Figura 32. Iniciar programa Pi-blaster desde terminal.

Luego, después de dicha sentencia, queda el programa funcionando. Esto se puede ver en la información que el programa devuelve, en la figura 33 detalla la información.

```
sudo ./pi-blaster
Using hardware:                 PWM
Number of channels:               8
PWM frequency:                100 Hz
PWM steps:                      1000
Maximum period (100  %):      10000us
Minimum period (0.100%):         10us
```

Figura 33. Información de programa de Pi-blaster.

Para saber cómo girarán los motores, es necesario conocer cuál es el avance más pequeño entre posición y posición de motor, para determinar la resolución del avance dentro del rango de movimiento se ejecuta por prueba y error hasta determinar dicho valor; en la figura 34 se muestra la forma de ejecutar pi-blaster, se ejemplifica con el puerto GPIO 17 todos los posibles estados.

Examples: Turning PWM pins ON

- To completely turn off GPIO pin 17:

  echo "17=0" > /dev/pi-blaster

- To completely turn on GPIO pin 17:

  echo "17=1" > /dev/pi-blaster

- To set GPIO pin 17 to a PWM of 20%

  echo "17=0.2" > /dev/pi-blaster

Examples: Turning PWM pins OFF (releasing a pin so it can be used as digital GPIO)

- To release previously turned ON GPIO pin 17:

  echo "release 17" > /dev/pi-blaster

Figura 34. Ejemplos de utilización de Pi-blaster mediante PWM.

El valor de ciclo de trabajo para límite inferior del servomotor que hace el movimiento de pan es de 4.5% y para el límite superior de 25%, mientras que el valor del servomotor de movimiento tilt es de 4.5% límite inferior y 12.5% límite superior. La resolución entre el rango de movimiento en ambos casos es de 0.001.

Para realizar la interfaz web de la aplicación, es necesario tener instalada Jquery en cualquier versión. Jquery es una librería realizada en javascrip que tiene funciones bien

potentes para realizar aplicaciones webs dinámicas. También se usa el control range de HTML5 para realizar los slider que controlaran los servos de manera horizontal y vertical. Otra herramienta sera css para darle el aspecto al control, y php que recibirá el parámetro de entrada y ejecutará el comando que permite el movimiento de los motores.

La figura 35 muestra el resultado de tener los divs de vídeo streaming y los slider horizontal y vertical, en ella se muestra estudiantes de la Escuela de Ingeniería Eléctrica en el monitor Electrica_Planta_1 a las 08:34:48 del día 21 de agosto de 2014 .

En la figura 36 se muestra el código fuente de la página que formará parte del sistema.

Figura 35. Captura del control y transmision de video.

```
<!DOCTYPE html>
<html lang=es>
<head>
<link rel="stylesheet" href="ajax.css"/>
<script type="text/javascript" src="jquery-1.8.3.js"></script>

<script>...
</head>

<body id="principal">
<p id="coordenada">Coordenadas</p>

<div id="bloque">
    <p class="leyenda">Vertical</p>
    <p id="#valorleido" class="muestra">45</p>
    <p class="leyenda">Horizontal</p>
    <div id="display2" class="muestra">0</div>
</div>

<div id="marco">
    <div id="submarco1">
    <input type="range" name="envio" id="envioV" step = "0.0001" min="0.045" value="0.1475" max="0.25" value="0" href="javascript:;"
    onclick="realizaProceso($('#envioV').val());return false;"//>
    </div>

    <div id="imageFeed">
    <img id="liveStream" src="http://192.168.1.8/cgi-bin/nph-zms?mode=jpeg&amp;monitor=1&amp;scale=100&amp;maxfps=5&amp;buffer=1000&amp;
    user=admin&amp;connkey=549855&amp;rand=1401837945" alt="PlantaBaja" width="320" height="240"/>
    </div>
</div>

<input type="range" name"envio2" id ="envioH" step="0.0001" min="0.045" value="0.1475" max="0.25" href="javascript:;" onclick="
realizaProceso2($('#envioH').val());return false;"/>
<p id="valorleido"></p>
<p id="valorleido2"></p>

</body>
</html>
```

Figura 36. Código fuente para la generación de vídeo en tiempo real.

RESULTADOS Y DISCUSIÓN.

Dentro de la elección del software para mostrar páginas web se utiliza el servidor web Apache; que está instalado en la placa de desarrollo UDOO, UDOO es un sistema embebido de muy bajo costo y es capaz de soportar sistemas operativos Ubuntu y Android, la placa tiene instalado Ubuntu como sistema operativo recomendado por el fabricante. Apache en su instalación configura la ruta o path por defecto en la raíz de los directorios /var/www/; por lo tanto, cada vez que se reciba una petición de cualquier programa cliente buscará el recurso en la ruta /var/www/. Esta ruta contiene los archivos que gestionan, configuran y administran las peticiones del sistema de Seguridad cliente-servidor creadas.

Una característica a destacar del sistema es que la mayoría de archivos fuentes que entablan la comunicación con el programa cliente, están escritos en lenguaje PHP, necesitando instalar el intérprete de PHP para que estos archivos puedan hacer la comunicación de manera eficiente con él cliente. Además es necesario un gestor de bases de datos que almacene la información, para que sea usada cuando se considere necesario; el sistema gestor MySQL cubre esa necesidad. Tomando en cuenta los detalles anteriores se explica en qué consiste la programación de cada uno de los archivos:

DOMOBASE

Es el archivo que permite la comunicación con la base de datos. Este archivo define una clase llamada GestionarDB, esta clase permite crear objetos y aplicar los métodos que en ella se encuentran, se definen variables globales (figura 37, línea 4 a 16) que serán utilizadas por los métodos de la clase.

```
1   <?php
2   class GestionarDB {
3   /* Variables Utilizadas para la connexion de la Base de Datos */
4   var $servidor;
5   var $usuario;
6   var $clave;
7   var $base;
8   var $conectado;
9   var $Bandera;
10  var $Tabla;
11  var $consultaUpdate;
12  var $Id;
13  var $consultarU;
14  var $consultaSelect;
15  var $consultarS;
16  var $horario = array();
```

Figura 37. Clase GestionarDB.

El primer método es el constructor llamado GestionarDB figura 37, este recibe cuatro parámetros que son: El nombre de la base de datos, el nombre del servidor, el nombre de usuario con permisos para manipular la base de datos y la contraseña de acceso.(Figura línea 18), estos parámetros se pasan como argumento para iniciar las variables globales que se encargan de realizar la conexión con la base de datos (figura línea 19 a 23), resaltando que este método simplemente sirve para inicializar las variables globales que son propiedades fundamentales de la clase como se muestra en la figura 38:

```
18  function GestionarDB($db="", $host="localhost", $user="nobody", $passwd="") {
19          $this->servidor = $host;
20          $this->usuario = $user;
21          $this->clave = $passwd;
22          $this->base = $db;
23  }
```

Figura 38. Función constructor GestionarDB.

Esta clase también contiene un método llamado conectar (la figura 39 línea 25), recibe como argumentos las variables globales inicializadas con el método GestionarDB y son: Él nombre del servidor, el nombre de usuario y clave de acceso. Estas variables son los PARAMETROS que recibe la función mysql_connect para establecer la conexión con el gestor de base de datos MySQL, devolviendo, un manejador que permite tener una sesión abierta con el Gestor MySQL; esta sesión es controlada por el manejador devuelto, (en la línea 26); de tal manera que el manejador es importante para la comunicación con la base de datos. También decir que el símbolo de @ antepuesto a la función mysql_connect; sólo sirve para que no se muestren los errores en la salida estándar, que para el caso es el cliente web. Conectar también utiliza la función mysql_select_db. Dicha función permite seleccionar la base de datos que se está utilizando (línea 29), mysql_selec_db recibe como parámetros la variable global en la que se almacena el nombre de la base de datos que se gestiona, y también recibe otro parámetro que corresponde al manejador encargado de interactuar con MySql. La estructura de este método se muestra a continuación:

```
25  function conectar() {
26          $this->conectado = @mysql_connect($this->servidor, $this->usuario, $this->clave);
27          if($this->conectado){
28                  //echo("se ha conectado la base de datos");
29                  mysql_select_db($this->base,$this->conectado);
30          } else {
31                  echo("No se pudo conectar al servidor");
32          }
33  }
```

Figura 39. Función conectar.

También hay que destacar que todos aquellos métodos de la clase que pretendan realizar una gestión con una base de datos, deben de invocar primero a los dos métodos antes mencionados para luego proceder con cualquier gestión. Continuando con la descripción de los métodos que componen la clase GestionarDB; se encuentra el método actualizarB. Este método recibe tres parámetros que corresponden a: el nombre de la tabla a actualizar, el valor a cambiar en el campo bandera que corresponda a un ID específico y el ID como tercer argumento (línea 34 de la figura 40). Este método sencillamente inicializa las tres variables globales tabla, Bandera y consultaUpdate (línea 35 a la línea 44 figura 40); de las cuales las primeras dos variables tabla y Bandera forman parte de una cadena que escribe el tipo de consulta de actualización que se quiere realizar, a una de las tablas de la base de datos; cabe mencionar que esta cadena es guardada en la última variable inicializada para luego ser ejecutada por la función mysql_query; que recibe como parámetro la variable que contiene la cadena de la consulta y el manejador que interactúa con la base de datos(variable Global llamada conectado). La respuesta de esta función es almacenada en una variable global llamada consultarU(línea 45 figura 40), aunque el propósito de esto es, solamente para efecto de controlar los posibles errores que se puedan dar, que serán gestionados por otro método. La estructura de actualizarB es la que se muestra en la figura 40.

```
34  function actualizarB($tablaP="Electrica", $banderaP="verde", $idP="001") {
35          $this->Tabla = $tablaP;
36          $this->Bandera = $banderaP;
37          $this->Id=$idP;
38          $this->consultaUpdate="UPDATE ";
39          $this->consultaUpdate.=$this->Tabla;
40          $this->consultaUpdate.=" SET bandera='";
41          $this->consultaUpdate.=$this->Bandera;
42          $this->consultaUpdate.="' WHERE idsensor='";
43          $this->consultaUpdate.=$this->Id;
44          $this->consultaUpdate.="';";
45          $this->consultarU=mysql_query($this->consultaUpdate,$this->conectado);
46
47  }
```

Figura 40. Función actualizarB.

Otro Método de la Clase GestionarDB es saberhora; (figura 41 línea 48), recibe como parámetros dos argumentos; la tabla que contiene los Horarios de cada uno de los edificios que tienen implementados los tipos sensores y el nombre del campo que contiene el edificio donde se desea saber la hora a la cual esté fue configurado.

Luego se inicializa la variable global Tabla como se muestra en la figura 41, donde guarda el nombre de la tabla pasada como argumento (la línea 49), para que este forme parte de la cadena que contiene la variable ConsultaSelect que es utilizada como consulta de

selección para la base de datos (línea 50 a la línea 54 figura 41), luego se realiza la consulta en la línea 55 con la función mysql_query y la respuesta a la consulta es guardada en la variable consultarS, y por medio de la función mysql_result, se captura en un arreglo, los campos de los horarios que se necesitan; que para que son la hora y minutos de inicio de activación de las alarmas y también los de desactivación, para luego retornar el arreglo horario con los horarios solicitados. (Desde línea 56 a línea 60 figura 41).

```
48  function saberhora($tablaH="HorarioLaboral", $Edificio="Electrica"){
49          $this->Tabla = $tablaH;
50          $this->consultaSelect="SELECT * FROM ";
51          $this->consultaSelect.=$this->Tabla;
52          $this->consultaSelect.=" WHERE Edificio='";
53          $this->consultaSelect.=$Edificio;
54          $this->consultaSelect.="';";
55          $this->consultarS=mysql_query($this->consultaSelect,$this->conectado);
56          $this->horario[0]=mysql_result($this->consultarS, 0, "HoraI");
57          $this->horario[1]=mysql_result($this->consultarS, 0, "minutoI");
58          $this->horario[2]=mysql_result($this->consultarS, 0, "HoraF");
59          $this->horario[3]=mysql_result($this->consultarS, 0, "minutoF");
60          return $this->horario;
61  }
```

Figura 41. Función saberhora.

Ahora se describe aquellos métodos que entregan información acerca de los sensores instalados en los pines GPIO de la Raspberry PI. Comenzando con los métodos el primero de ellos es AlarmaRoja (la figura 42 línea 62). Este método recibe un solo argumento y es el nombre de la tabla del edificio donde están instalados los sensores, de los cuales se necesita obtener la información; estos sensores representan tuplas en esta tabla que es recibida como argumento. Y estas tuplas contienen la información del estado actual de los sensores así como su respectiva descripción. Como se observa, este método hace lo siguiente; primero coloca el nombre de la tabla que representa un edificio de la Universidad de el Salvador en la variable global Tabla, la cual forma parte de una cadena que va ser usada como consulta, tal y como se muestra en la línea 63.También se inicializa la variable banderaR con la cadena Rojo para llamar a los registros con alarmas que tengan bandera roja (línea 64).

Luego guarda la consulta de selección en la variable ConsultarR y luego ejecuta la consulta con mysql_query, la respuesta es guardada en la variable ConsultarRR (la línea 65 a 70), después el número de tuplas que se obtienen dé la consulta con la función mysql_num_row; la cual se guarda en la variable registrosR (línea 71), luego, con una sentencia condicional If se le indica al método la condición siguiente: si el número de tuplas es diferente de cero, ejecute lo que está dentro de las llaves (línea 72 y línea 80); si es cero el número de tuplas. Ejecute las líneas que estén entre la línea 82 y 85.

Cuando el número de tuplas es diferente de cero se usa la función mysql_fetch_array configurada de tal manera que devuelva un arreglo asociativo con los campos de los registros de los sensores que tenga como estado una bandera Roja recorrido por un lazo while, y se guarda en un arreglo multidimensional llamado arregloR; para ser retornarlo. como se muestra de la línea 73 a la línea 81 de la figura 42.

```
62  function AlarmaRoja($tablaR="Electrica"){
63          $this->Tabla = $tablaR;
64          $banderaR="rojo";
65          $consultaR="SELECT * FROM ";
66          $consultaR.=$this->Tabla;
67          $consultaR.=" WHERE bandera='";
68          $consultaR.=$banderaR;
69          $consultaR.="';";
70          $consultarRR=mysql_query($consultaR,$this->conectado);
71          $registrosR=mysql_num_rows($consultarRR);
72          if($registrosR != 0){
73                  $acumuladorR=0;
74                  while($entregarR=mysql_fetch_array($consultarRR, MYSQL_ASSOC)){
75                          $arregloR[$acumuladorR][0]=$entregarR["Tsensor"];
76                          $arregloR[$acumuladorR][1]=$entregarR["zona"];
77                          $arregloR[$acumuladorR][2]=$entregarR["idsensor"];
78                          $arregloR[$acumuladorR][3]=$entregarR["descripcion"];
79                          $acumuladorR=$acumuladorR+1;
80                  }
81                  return $arregloR;
82          } else{
83                  $arregloR = array();
84                  return $arregloR;
85          }
86  }
```

Figura 42. Función AlarmaRoja.

Antes de seguir explicando los métodos, hay que mencionar que todos los métodos de la clase **GestionarDB** son por defecto públicos dado que al no especificar en PHP si un método es público o privado PHP interpreta de que este método va a ser público y van a ser llamados por otros archivos y funciones externas a la Clase **GestionarDB** y si estos fueran de tipo privado no podrían ser llamados externamente a la clase solo podrían ser usados por los métodos internos de la Clase.

Siguiendo con la explicación de la Clase **GestionarDB** otro método que hace la misma función que el método **AlarmaRoja** es el método **AlarmaNaranja** que se diferencia del método **AlarmaRoja** porque este en vez de retornar todos aquellas Tuplas que contengan el Estado de Alarma Roja en el campo Bandera de cualquiera de las tablas que representen Edificios de nuestra base de datos, **AlarmaNaranja** Retorna en un Arreglo multidimensional con todas las Tuplas que contengan el estado de Alarma Naranja en el

Campo Bandera, es por eso que como su función es similar no se explica y se muestra mejor la estructura interna de este método como se ve en la línea 87 a la línea 111 de la figura 43 la cual es mostrada a continuación:

```
87  function AlarmaNaranja($tablaN="Electrica"){
88          $this->Tabla = $tablaN;
89          $banderaN="naranja";
90          $consultaN="SELECT * FROM ";
91          $consultaN.=$this->Tabla;
92          $consultaN.=" WHERE bandera='";
93          $consultaN.=$banderaN;
94          $consultaN.="';";
95          $consultarNN=mysql_query($consultaN,$this->conectado);
96          $registrosN=mysql_num_rows($consultarNN);
97          if($registrosN != 0){
98                  $acumuladorN=0;
99                  while($entregarN=mysql_fetch_array($consultarNN, MYSQL_ASSOC)){
100                         $arregloN[$acumuladorN][0]=$entregarN["Tsensor"];
101                         $arregloN[$acumuladorN][1]=$entregarN["zona"];
102                         $arregloN[$acumuladorN][2]=$entregarN["idsensor"];
103                         $arregloN[$acumuladorN][3]=$entregarN["descripcion"];
104                         $acumuladorN=$acumuladorN+1;
105                 }
106                 return $arregloN;
107         } else{
108                 $arregloN = array();
109                 return $arregloN;
110         }
111 }
```

Figura 43. Función AlarmaNaranja.

Otro método que sirve para saber el estado y la descripción de los sensores es el método **AlarmaRojaZ** este método es similar al método **AlarmaRoja**, y se explica de manera superficial teniendo en cuenta que la principal diferencia de este método con el método **AlarmaRoja** es que **AlarmaRojaZ** va retornar a todas aquellas tuplas que contengan el estado de Alarma Roja y para la zona que se desea saber, es por eso que este método va a recibir como argumento dos PARAMETROS los cuales son, la tabla del nombre del edificio que contiene el estado y la descripción de los sensores de los cuales se necesita obtener la información y la zona en que estos se encuentran tal como se ve en la línea 112 de la figura 44.

luego en el cuerpo del método lo primero que realiza es inicializar la variable **Tabla** con el nombre de la tabla como argumento luego se genera una cadena de la cual forma parte el contenido de la variable **Tabla** y esta guardada en una variable con el nombre de **consultaRZ** para luego ejecutar la consulta y guardar su respuesta en la variable **consultarRRZ** luego la función **mysql_num_rows** permite obtener de la variable

consultarRRZ el número de tuplas que fue enviado en la respuesta a la consulta hecha por este método como se muestra de la linea 113 a la línea 123. Saber la cantidad de tuplas es necesario para saber si los registros son diferentes de cero y si estos son diferentes de cero recorrerá con un lazo while todos los registros obtenidos de la repuesta a la consulta y los guarda en un arreglo el cual será retornado por la variable **arregloRZ** ahora si no hubieran registros se mandaría el arreglo que para este caso es **arregloRZ** pero de forma vacía como se muestra de la línea 124 a la línea 138 de la figura 44:

```
112  function AlarmaRojaZ($tablaRZ="Electrica", $zonaRZ="1"){
113          $this->Tabla = $tablaRZ;
114          $banderaRZ="rojo";
115          $consultaRZ="SELECT * FROM ";
116          $consultaRZ.=$this->Tabla;
117          $consultaRZ.=" WHERE bandera='";
118          $consultaRZ.=$banderaRZ;
119          $consultaRZ.="' AND zona='";
120          $consultaRZ.=$zonaRZ;
121          $consultaRZ.="';";
122          $consultarRRZ=mysql_query($consultaRZ,$this->conectado);
123          $registrosRZ=mysql_num_rows($consultarRRZ);
124          if($registrosRZ != 0){
125                  $acumuladorRZ=0;
126                  while($entregarRZ=mysql_fetch_array($consultarRRZ, MYSQL_ASSOC)){
127                          $arregloRZ[$acumuladorRZ][0]=$entregarRZ["Tsensor"];
128                          $arregloRZ[$acumuladorRZ][1]=$entregarRZ["zona"];
129                          $arregloRZ[$acumuladorRZ][2]=$entregarRZ["idsensor"];
130                          $arregloRZ[$acumuladorRZ][3]=$entregarRZ["descripcion"];
131                          $acumuladorRZ=$acumuladorRZ+1;
132                  }
133                  return $arregloRZ;
134          } else{
135                  $arregloRZ = array();
136                  return $arregloRZ;
137          }
138  }
```

Figura 44. Función AlarmaRojaZ.

Otro de los métodos es **AlarmaNaranjaZ** prácticamente hace la misma función que **AlarmaRojaZ** con la única diferencia que en vez de retornar las tuplas que contienen banderas con estados de Alarmas Rojas este método retorna aquellas banderas que tienen el estado de alarma Naranja en el campo bandera claro está que para la zona especificada en el argumento del método, como se muestra en la **figura 45** de la línea 139 a la línea 165 la estructura que se muestra es casi la misma de **AlarmaRojaZ** el único cambio que hay es que la variable **banderaNZ** contiene la cadena “naranja” en vez de la Cadena “Rojo” y que las variables finalizan con NZ en vez de con RZ como en el método

AlarmaRojaZ haciendo énfasis en que el método hace lo mismo pero para las banderas de tipo Naranja tal y como se muestra en la figura 45.

```
139  function AlarmaNaranjaZ($tablaNZ="Electrica", $zonaNZ="1"){
140          $this->Tabla = $tablaNZ;
141          $banderaNZ="naranja";
142          $consultaNZ="SELECT * FROM ";
143          $consultaNZ.=$this->Tabla;
144          $consultaNZ.=" WHERE bandera='";
145          $consultaNZ.=$banderaNZ;
146          $consultaNZ.="' AND zona='";
147          $consultaNZ.=$zonaNZ;
148          $consultaNZ.="';";
149          $consultarNNZ=mysql_query($consultaNZ,$this->conectado);
150          $registrosNZ=mysql_num_rows($consultarNNZ);
151          if($registrosNZ != 0){
152                  $acumuladorNZ=0;
153                  while($entregarNZ=mysql_fetch_array($consultarNNZ, MYSQL_ASSOC)){
154                          $arregloNZ[$acumuladorNZ][0]=$entregarNZ["Tsensor"];
155                          $arregloNZ[$acumuladorNZ][1]=$entregarNZ["zona"];
156                          $arregloNZ[$acumuladorNZ][2]=$entregarNZ["idsensor"];
157                          $arregloNZ[$acumuladorNZ][3]=$entregarNZ["descripcion"];
158                          $acumuladorNZ=$acumuladorNZ+1;
159                  }
160                  return $arregloNZ;
161          } else{
162                  $arregloNZ = array();
163                  return $arregloNZ;
164          }
165  }
```

Figura 45. Función AlarmaNaranjaZ

método **cambiarhora** el cual recibe como parámetro el nombre de la tabla que contiene la horarios de los edificios en los cuales van a estar funcionando cada uno de los sensores. Este método inicializa la variable **Tabla** con el parámetro que recibe, luego esta variable forma parte de una cadena que es utilizada como consulta de selección y que es guardada en la variable **consultarHorario** para luego ser ejecutada por la función **mysql_query** y su respuesta es guardada en la variable **consultarHorarioT.** Luego como se hace en los demás métodos se recorre con un lazo while todas las tuplas por medio de la función **mysql_fetch_array** retornando así el registro de cada uno de los nombres de los edificios que contienen sensores en un arreglo llamado **arregloHT** como se muestra en las líneas 166 a la 178 de la figura 46.

```
166  function CambiarHora($tablaHorario="HorarioLaboral"){
167          $this->Tabla = $tablaHorario;
168          $consultarHorario="SELECT * FROM ";
169          $consultarHorario.=$this->Tabla;
170          $consultarHorario.=";";
171          $consultarHorarioT=mysql_query($consultarHorario,$this->conectado);
172          $acumuladorHT=0;
173          while($entregarHT=mysql_fetch_array($consultarHorarioT, MYSQL_ASSOC)){
174                  $arregloHT[$acumuladorHT]=$entregarHT["Edificio"];
175                  $acumuladorHT=$acumuladorHT+1;
176          }
177  return $arregloHT;
178  }
```

Figura 46. Función cambiarHora.

Otro de los Últimos métodos de esta Clase se llama **CrearHora** recibe 6 parámetros los cuales son el nombre de la tabla que contiene los horarios de funcionamiento de las Alarmas para cada edificio, El Edificio al cual se van a configurar el horario de las Alarmas, la hora de Inicio, los minutos en que va a comenzar a funcionar, la hora de finalización de las Alarmas y los minutos en que estas van a dejar de funcionar esto se puede ver en la línea 179 de la **figura 47**. Después de recibir los 6 parámetros se inicializa la variable **Tabla**, con el nombre de la tabla que contiene los horarios para cada uno de los edificios el cual es el primer argumento que recibe el método, luego se inicializa la variable **horaInicial** con la hora en que las alarmas van a comenzar a funcionar, luego se inicializa la variable **minutoInicial** con los minutos en el que la alarmas van a comenzar a funcionar después de la hora especificada como la de inicio, luego se inicializan dos variables más una con la hora de finalización (**horaFinalizada)** y la otra con los minutos de cierre (**minutoFinalizar)** Como se muestra de la línea 180 a la línea 185. Luego se guarda en una variable la consulta de actualización como se muestra en las líneas de la 186 a la 198 de la figura 47 y en la línea 199 se ejecuta la consulta.

```
179  function CrearHora($tablaFH="HorarioLaboral", $edificioHA="Electrica", $horaIA="20", $minutoAI="30", $horaFA="6", $minutoAF="30"){
180          $this->Tabla = $tablaFH;
181          $horaInicial=$horaIA;
182          $minutoInicial=$minutoAI;
183          $horaFinalizar=$horaFA;
184          $minutoFinalizar=$minutoAF;
185          $localHA=$edificioHA;
186          $actualizarHorario="UPDATE ";
187          $actualizarHorario.=$this->Tabla;
188          $actualizarHorario.=" SET HoraI='";
189          $actualizarHorario.=$horaInicial;
190          $actualizarHorario.="', minutoI='";
191          $actualizarHorario.=$minutoInicial;
192          $actualizarHorario.="', HoraF='";
193          $actualizarHorario.=$horaFinalizar;
194          $actualizarHorario.="', minutoF='";
195          $actualizarHorario.=$minutoFinalizar;
196          $actualizarHorario.="' WHERE Edificio='";
197          $actualizarHorario.=$localHA;
198          $actualizarHorario.="';";
199          $salvarconsulta=mysql_query($actualizarHorario,$this->conectado);
200  }
```

Figura 47. Función CrearHora.

Otro de los últimos métodos de la clase **GestionarDB** es el método **AccesoBasico** este recibe como argumentos dos parámetros el primero es el nombre de la tabla que contiene a los usuarios que están autorizados para poder utilizar el sistema de vigilacia y el segundo el Usuario del cual se pretende obtener información tal y como se muestra en la línea 201 de la **figura 48,** este método comienza inicializando dos variable las cuales son **Tabla** y **idUsuario** la cuales toman el primer y segundo Argumento del método respectivamente luego estas dos variables forman parte de una cadena que se guarda en la variable

consultarUsuario y esta cadena en realidad es una consulta de selección y esto lo puede ver entre las líneas 202 a la 208 de la **figura 48,** luego se ejecuta la consulta.

Después que se ejecuta la consulta se obtienen la cantidad de tuplas y si estas son diferentes de cero, se recorre cada una de las tuplas encontradas con un lazo WHILE obteniendo los parámetros del usuario, PASSWORD y cargo. Luego estos son retornados por medio de un arreglo como se ve entre las líneas 209 y 223 de la figura 48.

```
201  function AccesoBasico($tablaUsuario="Usuarios", $loginUsuario="Marenco"){
202          $this->Tabla = $tablaUsuario;
203          $idUsuario=$loginUsuario;
204          $consultarUsuario="SELECT * FROM ";
205          $consultarUsuario.=$this->Tabla;
206          $consultarUsuario.=" WHERE Usuario='";
207          $consultarUsuario.=$idUsuario;
208          $consultarUsuario.="';";
209          $respuestaUsuario=mysql_query($consultarUsuario,$this->conectado);
210          $registrosUsuarios=mysql_num_rows($respuestaUsuario);
211          if($registrosUsuarios != 0){
212                  while($entregarUS=mysql_fetch_array($respuestaUsuario, MYSQL_ASSOC)){
213                          $arregloUS[0]=$entregarUS["Usuario"];
214                          $arregloUS[1]=$entregarUS["Password"];
215                          $arregloUS[2]=$entregarUS["Cargo"];
216                  }
217                  return $arregloUS;
218          } else {
219                  $arregloUS = array();
220                  return $arregloUS;
221          }
222  }
223  }
224  ?>
```

Figura 48. Función Acceso Basico.

CONCLUSIONES

Al final de este trabajo de graduación, se pueden concluir varios puntos detallados a continuación.

- Se logró obtener un prototipo de sistema de seguridad basado en el protocolo de comunicación IEEE 802.11 con software y hardware libre de bajo costo en relación a un sistema de seguridad comercial. Dicho logro fue posible debido a las herramientas ofrecidas por la comunidad de software y hardware libre.

- La placa de desarrollo UDOO, es la responsable directa de tener un servidor web funcional y robusto en donde se montaron todos los servicios de la aplicación.

- Los servicios obtenidos constan de una interfaz web amigable, en donde existe los usuarios administrativos con todos los permisos para manipular la configuración del sistema, y los usuarios comunes, que solo son usuarios del sistema. Dentro de los servicios se tienen: creación de nuevos usuarios, manipulación de horarios fuera de actividades normales de la Universidad, video en tiempo real de las áreas asignadas, mostrar el estado de los sensores, generación de videos como respaldo de evidencia de eventos delictivos, manipulación del movimiento de las cámaras.

- La placa de desarrollo Raspberry PI, fue capaz de proporcionar los puntos de control, en ella, se conectaron los sensores de presencia, magnéticos y vibración, también se conectó una cámara web para proporcionar a la aplicación, la parte de video vigilancia, logrando obtener un video streaming bastante aceptable en las áreas controladas, así como la instalación del servidor de video vigilancia zoneminder, que dentro de sus cualidades está la del almacenamiento de imágenes para la generación de video, la taza de transferencia de imágenes en unidades f.p.s (frame por segundo), manipulación en el color de las imágenes.

- Dentro de las herramientas de software libre utilizadas, cada una de ellas hizo su aporte en funciones específicas del prototipo. Detallando los aportes individuales en software para la parte del sistema, programación y aplicación.

- En el servidor principal El sistema operativo UDOObuntu hizo su aporte para montar el sistema operativo en la placa UDOO, UDOObunto, proporciona una plataforma robusta y estable como todo sistema LINUX al sistema, también se instaló el servidor web Apache, haciendo posible la visualización de la aplicación en

el lado del cliente; como software de programación, la técnica de programación AJAX, hizo posible la realización de la aplicación interactivamente proporcionándole dinamismo a los sitios web, esto porque dadas las características en si misma del sistema de seguridad, se necesita de rapidez en la parte de monitorización de las alarmas, el lenguaje de programación PHP hizo su aporte en la creación de todas las paginas dinámicas que se muestran en lado del cliente. Mencoder, aplicación quehizo posible la generación de videos a partir de imágenes estáticas. el software de aplicación MySQL, como gestor de la base de datos que tiene todas las tablas responsables de los estados en las alarmas, así como la creación de usuarios autorizados y comunes, las tablas de todos los horarios y la tabla de videoteca.

En los puntos de control, al estar utilizando la Raspberry PI, RASPBIAN es la plataforma instalada permitiendo así, tener el lenguaje de programación Python, encargado de sensar los estados de las señales enviadas por todos los sensores. El servidor de video zoneminder, que hizo posible junto con la placa y una camará web, emular una camará IP, permitiendo la comunicación remotamente, apache, mySQL y PHP permiten la instalación de zoneminder, así como paquetes adicionales necesarios para una correcta instalación del servidor. PiBlaster, Aplicación responsable del manejo de señales PWM, etc.

Adicionalmente al desarrollo en software y hardware se pueden mencionar.

- Los sistemas de seguridad comerciales incluyen muchos elementos que hacen de estos, sistemas potentes y muy eficientes, sin embargo, y como desventaja principal, se hacen muchas veces inalcanzables porque presentan costos monetarios elevados, debido a la compra de licencias tanto en software como en hardware.
- Los sistemas de seguridad basados en software y hardware de libre distribución, son una buena alternativa para implementar sistemas de seguridad de bajo costo, dado que se evita el gasto de adquirir licencias sumado a esto que algunos software de libre distribución, presentan características que muchas veces y en relación a software de uso comercial, presentan funcionalidades mucho más potentes, por ejemplo el servidor de video zoneminder, dentro de las cualidades que presenta, tiene la característica de soportar cámaras de todo tipo tanto comerciales como genéricas incluyendo las webcam implementando así un sistema de seguridad de bajo costo.
- Las microcomputadoras embebidas, Rasberry Pi y UDOO, fácilmente pueden ser sustitutos del hardware comercial, debido a las características que presentan, por ejemplo, de los 26 puertos GPIO proporcionados por la Rasberry PI, pueden ser

utilizados para el diseño de sistemas de seguridad, 17 puertos GPIO como posibles entradas y salidas de datos desde y hacia sensores y dispositivos complementarios. Dicha característica está siendo utilizada para controlar los dispositivos de entrada provenientes de los sensores magnéticos, de presencia, de vibración, además de la configuración de 2 puertos GPIO vía software para salidas PWM del control de servomotores.

- En resumen, todas las tecnologías estudiadas en software y hardware libre, bien encaminadas, pueden llegar a la construcción de aplicaciones potentes, siempre y cuando se haga un esfuerzo por parte de la persona que la desarrollé.

RECOMENDACIONES.

Después de concluir las ventajas de utilizar estas tecnologías se puede hacer las siguientes recomendaciones.

- Proponer temas de investigación para la comunidad universitaria en áreas de seguridad, ya que un problema que se tuvo fue, la falta de material bibliográfico propio de la universidad en temas de sistemas de seguridad en cualquier tipo, siendo este una limitante en los antecedentes y punto de partida.

- Al hacer propuesta de temas de este tipo, se está incentivando a la comunidad estudiantil a ser más proactivo en la seguridad del patrimonio universitario. Dichos temas no necesariamente pueden ser trabajos de graduación, sino que también, temas en trabajos de proyecto de ingeniería.

Dar seguimiento a un sistema de seguridad, utilizando siempre tecnologías de libre distribución, pero sobre protocolos más especializados como el estándar IEEE 802.11.15.4.

- Aprovechar siempre al máximo, las placas de desarrollo como Raspberry Pi y UDOO, y también alguna otra desconocida, puesto que este tipo de sistemas embebidos, está desarrollándose continuamente. Por ejemplo, en el momento de comenzar el trabajo de investigación, la placa Raspberry Pi estaba en su versión B, en el transcurso del desarrollo del tema, la fundación Raspberry PI lanzo el modelo B+, incorporando 40 pines GPIO, 4 puertos USB y cambiando la tarjeta SD a una micro SD. Sin duda alguna, esta placa seguirá desarrollándose aumentando la velocidad de procesamiento, memoria RAM, etc. aprovechando en beneficio de la Universidad, este tipo de tecnologías.
- Aumentar la clase de la tarjeta SD mejora el rendimiento de la aplicación.

NOMENCLATURA.

AJAX: Asynchronous JavaScript And XML, JavaScript Asíncrono y XML.

AVI: Audio Video Interleave, Intercalar Audio y Video.

CCD: Charged Coupled Device, Dispositivo de acoplamiento de carga.

CCTV: Closed Circuit Television, Circuito cerrado de televisión.

CGI: Common Gateway Interface, Interfaz de Entrada Común.

CMOS: Complementary Metal Oxide Semiconductor, Semiconductor de óxido metálico complementario.

CSS: Cascading Style Sheets, Hoja de Estilo en Cascada.

DHCP: Dynamic Host Configuration Protocol, Protocolo de configuración dinámica de hosts.

EIE: Escuela de Ingeniería Eléctrica.

F.P.S: Fotogramas Por Segundo.

FIA: Facultad de Ingeniería y Arquitectura.

GNU: GNU's Not Unix, GNU no es Unix.

GPIO: General Purpose Input Output, Entrada Salida de Propósito General.

HTML: Hypertext Markup Language, Lenguaje de Marcado de Hipertexto.

HTTP: Hypertext Transfer Protocol, Protocolo de Transferencia de Hipertexto.

JPEG: Joint Photographic Experts Group, Grupo de Expertos en fotografía Unidos

kbps: kilobits por segundo.

LAN: Local Area Network, Red de área local.

Mbps: megabits por segundo.

MPEG: Moving Picture Experts Group. Grupo de Expertos de Imágenes en movimiento.

NTP: Network Time Protocolo, Red de Protocolo de Tiempo.

PTZ: PAN, TILT, ZOOM. Paneo, Inclinación, Enfoque.

PWM: Pulse Width Modulation, ModulaciónporAncho de Pulso.

SSH: Secure Shell, Intérprete de Órdenes Segura.

TCP: Transmission Control Protocol, Protocolo de control de transmisión.

UES: Universidad de El Salvador.

USB: Universal Serial Bus, Bus Serie Universal.

WEP: Wired Equivalent Privacy, Privacidad Equivalente con Cable.

WLAN: Wireless Local Area Network, Red de Área Local Inalámbrica.

GLOSARIO DE TÉRMINOS.

AD HOC: significa dar una solución elaborada a la medida o para un fin específico o particular.

ADVANCED IP SCANNER: Es una aplicación que explora direcciones IP en router.

APACHE: Servidor web más popular y su objetivo principal es la oferta de servicios WEB a los programas clientes.

CAMBOZOLA: Es una API desarrollada en JAVA que permite el streaming de imágenes. Esta aplicación es utilizada por el servidor Zoneminder.

CLIENTE SSH: Accede a máquinas remotas a través de una red. Permite manejar por completo la computadora mediante un intérprete de comandos,

CLIENTE/SERVIDOR: Describe la relación entre dos programas informáticos en la que un programa, el cliente, realiza una solicitud de servicio a otro programa, el servidor, que satisface esta solicitud. Normalmente, varios programas de cliente comparten los servicios de un programa de servidor común. Un navegador Web es un programa de cliente que solicita servicios (el envío de páginas Web o archivos) a un servidor Web.

DIRECCIÓN IP: Una dirección IP es simplemente una dirección en una red IP que utiliza un ordenador o dispositivo conectado a esa red. Las direcciones IP permiten a todos los ordenadores o dispositivos conectados encontrarse y pasarse datos entre ellos.

FFMEPG: Colección de software libre para la gestión de video y audio, su potencia está en transcodificar y hacer streaming de audio y video.

IEEE 802.11: Familia de estándares para LAN inalámbricas. El estándar 802.11 admite una transmisión de 1 o 2 Mbps en la banda de 2,4 GHz. IEEE 802.11b admite velocidades de datos hasta 11 Mbps en la banda de 2,4 GHz, mientras que 802.11g admite hasta 54 Mbps en la banda de 5 GHz.

JQUERY: Librería de JavaScript que simplifica la manera de interactuar con documentos HTML, árbol DOM , manejo de eventos, y desarrollo de animaciones.

LINUX: Linux es un sistema operativo de código fuente abierto dentro de la familia Unix. Por su solidez y disponibilidad, Linux ha adquirido popularidad en la comunidad de software libre y entre los desarrolladores de aplicaciones comerciales.

Motion JPEG: Motion JPEG es una técnica de compresión/descompresión sencilla para el vídeo en red.

MySQL: Es un Gestor de Base de Datos de libre distribución, este gestiona bases de datos relacionales con un grado muy alto de confiabilidad, seguridad y aplicabilidad.

NTSC: Sistema de codificación de colores analógico que se usa en los sistemas de televisión de Japón, Estados Unidos y otras partes de América. NTSC define la señal de vídeo con 525 líneas de TV por fotograma, con una frecuencia de actualización de 30 fotogramas por segundo.

PAL: Sistema de codificación de colores analógico que se usa en los sistemas de televisión de Europa y muchas otras partes del mundo. PAL define la señal de vídeo con 625 líneas de TV por fotograma, con una frecuencia de actualización de 25 fotogramas por segundo.

PHP: Lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico.

PI BLASTER: Software de libre distribución diseñado para generar señales PWM con un grado muy alto de resolución para los puertos GPIO de la placa Rasberry PI.

PUTTY: Es un clienteSSH, Telnet, rlogin, y TCP raw con licencia libre.

PYTHON: Lenguaje de programación interpretado

RASPBERRY PI: es una computadora de placa reducida

RASPBIAN: Sistema operativo oficial diseñado específicamente para correr en placa de desarrollo Raspberry Pi. Su desarrollo esta basado en el sistema operativo DEBIAN

ROUTER: también conocido como ENRUTADOR, es un dispositivo para conexión de subredes de computadoras a nivel 3 según el modelo OSI o nivel de red.

SDFORMATER: Es una aplicación gratuita para formateo de unidades SD.

SENSORES: Son dispositivos Electrónicos que buscan transformar señales físicas, químicas, térmicas, etc. a señales eléctricas, dentro de la rama de aplicaciones está la de seguridad electrónica, estos sensores son más específicamente de movimiento, de presencia, de vibración, así como de aviso etc.

SERVIDOR WEB: Un servidor Web es un programa que permite a los navegadores Web recuperar archivos de ordenadores conectados a Internet.

SHELL: Se emplea para referirse a aquellos programas que proveen una interfaz de usuario para acceder a los servicios del sistema operativo. Estos pueden ser gráficos o de texto simple, dependiendo del tipo de interfaz que empleen. Los shells están diseñados para facilitar la forma en que se invocan o ejecutan los distintos programas disponibles en el computador.

STREAMING: transmisión, tecnología que permite la descarga y casi al mismo tiempo la reproducción en especial audio y video.

TARJETA SD: Es una pequeña tarjeta que permite guardar información en dispositivos portátiles como teléfonos móviles, cámaras digitales o tablets. Las tarjetas SD se diferencian por sus medidas, su capacidad para almacenar contenidos y la velocidad a la que transmiten y copian los datos.

TUPLA: En programación es un tipo de dato agrupado, con mucha utilización en registro de base de datos.

UDOO: es una minicomputadora de desarrollo la cual es utilizada con sistema operativo Linux posee en la misma placa un micro controlador ARDUINO DUE.

UDOOBUNTU: Es el sistema operativo oficial de la placa de desarrollo UDOO, su diseño persigue dos objetivos fundamentales, velocidad y accesibilidad para el usuario.

WEP: Protocolo de seguridad inalámbrica, especificado en el estándar IEEE 802.11, que está diseñado para proporcionar a una red de área local inalámbrica (WLAN) un nivel de seguridad y privacidad comparable al que se espera normalmente de una LAN con cable. La seguridad se encuentra en dos niveles distintos: cifrado de 40 bits y cifrado de 128 bits. Cuanto mayor sea el número de bits, tanto más seguro será el cifrado.

WIN32DISKIMAGER: Aplicación que puede ser utilizada para escribir y grabar archivos imágenes de disco en dispositivos removibles o desmontables como las tarjetas SD. Es de licencia GNU.

WINDOWS: Sistema operativo o un conjunto de programas que posibilita la administración de los recursos de una computadora. Este tipo de sistemas empieza a trabajar cuando se enciende el equipo para gestionar el hardware a partir desde los niveles más básicos.

ZONEMINDER: software de libre distribución diseñado por muchos componentes entre los cuales se tienen: ficheros ejecutables binarios, scripts de Perl, Script de PHP, bases de datos, etc. con el propósito de tener un servidor de vídeo vigilancia completo y sin ningún costo de licencia.

REFERENCIAS BIBLIOGRÁFICAS.

- **Guía técnica de vídeo IP.**
  Factores y técnicas a considerar para un correcto uso de las aplicaciones de vigilancia y monitorización remota basada en IP.
  AXIS COMMUNICACIONS.

- **Redes Inalámbricas en los países en desarrollo.**
  Una guía práctica para planificar y construir infraestructuras de telecomunicaciones de bajo costo.
  Cuarta edición.

- **El gran libro de HTML5, CSS3, y Javascript.**
  Juan Diego Gauchat.
  marcombo, ediciones técnicas.

- **Diseño y construcción de un data logger multiparámetro con comunicación vía internet.**
  Luis Alfredo Gómez Calidonio.
  Willian Erick Hernández Iraheta.
  UES, marzo de 2013.

- **The Linux Command Line.**
  Second Internet Edition.
  William E. Shotts, Jr.

- **Python 2.1 Bible**
  Dave Brueck and Sthepen Tanner.
  Hungry Minds, Inc.

- **Manual de AJAX.**
  Las entrañas de AJAX
  Escrito por:Juan Mariano Fuentes.
  Dirigido por: Sergio Gálves Rojas, 2da Edición 2009.

SITIOS WEB.

- Fundación Raspberry Pi: http://www.raspberrypi.org/
- Página web oficial de UDOO: http://www.udoo.org/
- Página web oficial de Zoneminder: http://www.zoneminder.com
- http://www.axis.com/es/
- https://github.com/sarfata/pi-blaster/
- http://debianyderivadas.blogspot.com/2010/08/instalacion-y-configuracion-de.html

ANEXOS.

PREPARACIÓN DE TARJETA SD, INSTALACIÓN DE SISTEMA OPERATIVO DESDE WINDOWS Y ACCESO REMOTO.

Para configurar la tarjeta SD que contendrá el sistema operativo de la Raspberry Pi desde el sistema operativo Windows; es necesario hacer una serie de pasos previos empezando con el formateo correcto de la tarjeta SD hasta la instalación y configuración. En la web se puede encontrar suficiente información sobre la instalación del sistema operativo a la Raspberry Pi. A continuación se detalla un posible camino a tomar.

Nota: Una condición importante es que todas las herramientas son de libre distribución.

Programas necesarios.

SDFormater[31]. Es una aplicación gratuita para formateo de unidades SD.

win32diskimager[32]. Aplicación que puede ser utilizada para escribir y grabar archivos imágenes de disco en dispositivos removibles o desmontables como las tarjetas SD. Es de licencia GNU.

Raspbian[33]: Sistema operativo diseñado específicamente para correr en la Raspberry pi. En el link oficial se puede encontrar todo lo relacionado con el proyecto Raspberry.

Como una recomendación, es útil guardar todos los programas en una carpeta para una fácil instalación y acceso.

Paso 1. Formateo de Tarjeta SD.

Después de haber descargado el conjunto de aplicaciones, se formatea la tarjeta SD.

Al acceder a la aplicación SDFormater, este estará a la espera de recibir las acciones a ejecutar. En la figura 49 se muestra la interfaz de la aplicación, la interfaz es sencilla y muestra todos sus controles; este software establece una partición FAT32 por defecto, que es el tipo de partición más utilizada actualmente por elementos desmontables de almacenamiento.

En la figura 50 se muestra la ventana emergente que notifica el formato terminado después de dar clic al botón Format de la figura 49.

[31] https://www.sdcard.org/downloads/formatter_4/eula_windows/

[32] http://sourceforge.net/projects/win32diskimager/

[33] http://www.raspberrypi.org/downloads/

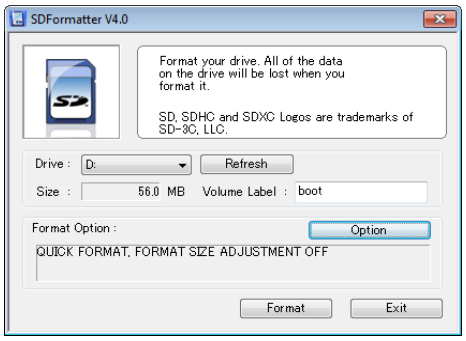

Figura 49. Interfaz de Usuario de SDFormatter.

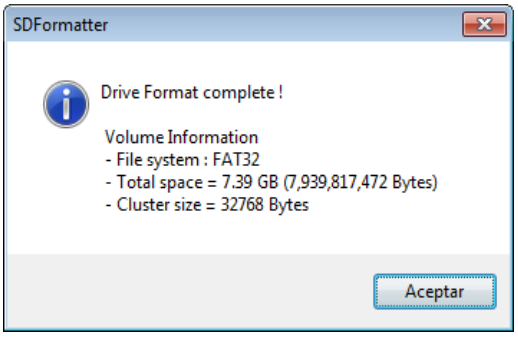

Figura 50. Finalización de formateo de tarjeta SD.

Paso 2. Escribiendo Sistema Operativo a tarjeta SD.

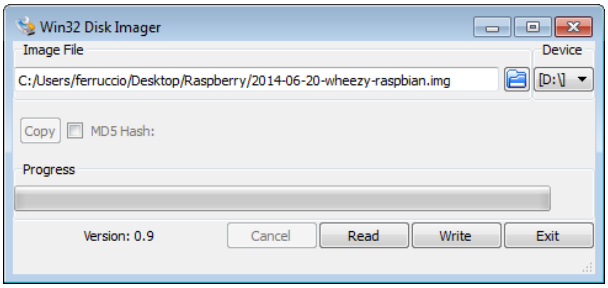

Figura 51. Interfaz de Win32 Disk Imager.

El siguiente paso es seleccionar la ruta de la imagen a grabar en la tarjeta SD, para ello es necesario ejecutar la aplicación Win32DiskImager en donde su interfaz de usuario se muestra en la figura 51. Simplemente se carga la imagen del sistema operativo en cualquier versión de Raspbian.img a la aplicación dando clic en la carpeta origen y escribiendo dicha imagen en la tarjeta SD. En la figura 51 se detalla la ruta completa de la imagen del sistema operativo; c:/Users/ferrucio/Desktop/Raspberry/2014-06-20-wheezy-raspbian.img. El tiempo estimado es de aproximadamente 10 minutos para cargar el sistema operativo en la tarjeta, el proceso se aprecia en la figura 52.

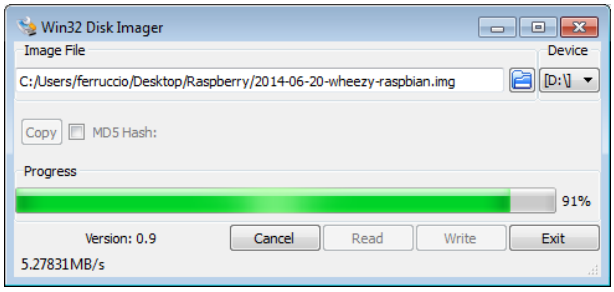

Figura 52. Porcentaje en escritura de archivo.img en tarjeta SD.

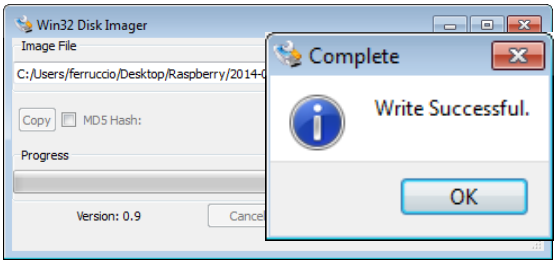

Figura 53.Escritura de archivo.img en tarjeta SD finalizado.

Luego de transcurrido el tiempo de grabado se despliega una ventana emergente que indica la finalización de la escritura del sistema operativo, como se muestra en la figura 53.

Paso 3. Accediendo a la Raspberry Pi y primeras configuraciones.

Para acceder a la Raspberry pi. Existen dos formas de realizarlo. Una de ellas es conectando un cable HDMI al puerto HDMI o por la entrada de video conectada a un televisor como monitor; en cualquier caso es directamente entre la microcomputadora y el monitor.

La segunda forma es vía SSH conectando un cable Ethernet entre el ROUTER y el puerto Ethernet.

Se utilizará el segundo camino para darle las primeras configuraciones a la Raspberry Pi. En este caso se auxiliara de dos nuevas aplicaciones. Advanced IP Scanner y PuTTy.

Advanced IP Scanner[34]. Es una aplicación que explora direcciones IP en router y es de libre distribución.

PuTTy[35]. Es un cliente SSH, Telnet, rlogin, y TCP raw con licencia libre.

Con estas dos nuevas aplicaciones, se accederá al sistema operativo recién instalado en la tarjeta SD de la Raspberry Pi. Después de conectar el cable Ethernet y la fuente de alimentación con la tarjeta SD colocada en la ranura, se utiliza Advanced IP Scanner para conocer la dirección IP asignada por el router al sistema operativo.

[34] http://www.advanced-ip-scanner.com/es/

[35] http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html

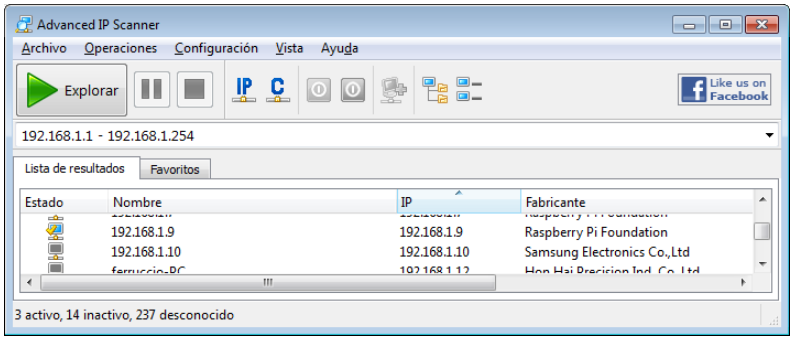

Figura 54. Interfaz de programa Advanced IPScanner.

Advanced IP Scanner muestra una lista de resultados de todas las direcciones IP encontradas que están siendo asignadas por el router. Para el caso particular de la Raspberry Pi, le fue asignada la dirección 192.168.1.9 como lo describe la figura 54. Advanced IP Scanner además de la dirección IP, muestra información adicional como el estado (cheque verde), el nombre del dispositivo y el fabricante. En este caso particular, el nombre y la dirección IP son las mismas.

Al conocer la dirección IP asignada por el router a la Raspberry Pi. Se accede remotamente vía PuTTy como se indica en la figura 55. Además, esta será la manera de acceder al sistema operativo de la placa siempre y cuando se conozca el valor de la dirección IP. Cuando se desconozca ese dato, se usa Advanced IPScanner descrito en el párrafo anterior. Una buena práctica es asignar una dirección IP estática en los archivos de configuración para redes de las distribuciones LINUX.

Se escribe la dirección IP en Host Name (or IP address) y al darle clic al botón Open se accede al sistema operativo de la Raspberry Pi.

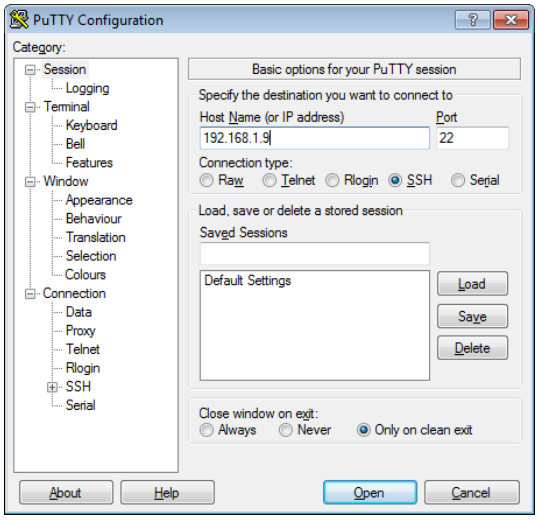

Figura 55. Interfaz de programa PuTTY

En primera instancia se pedirá login y password que por defecto son:

login as: pi

password: raspberry

En la figura 56 se resalta una nota NOTICE para la primera configuración. El comando sugerido por el sistema operativo es: **sudo raspi-config**

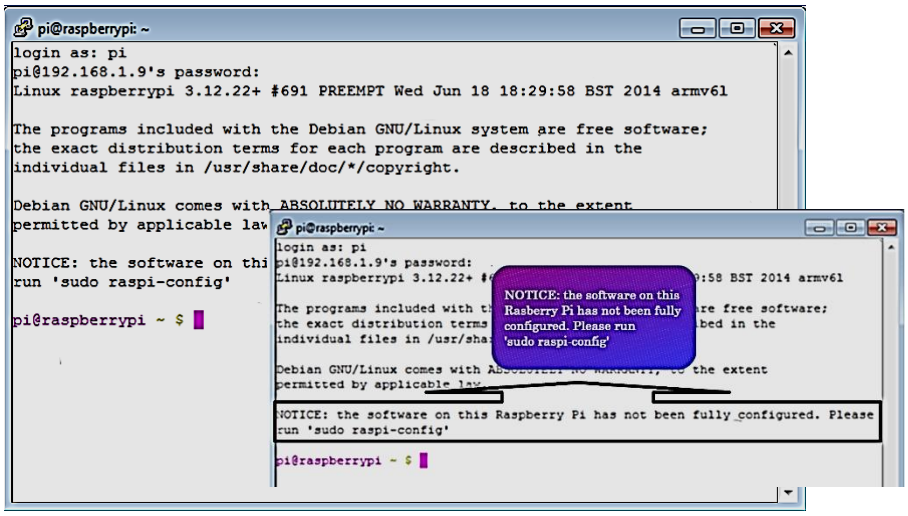

Figura 56. Accediendo por primera vez a Raspbian.

Al ejecutar el comando sudo raspi-config se accede a las primeras configuraciones. La figura 57 muestra un menú en donde solo la primera opción es de interés. Esta opción expande los archivos del sistema y garantiza el uso completo de la tarjeta SD. La figura 58 muestra el mensaje devuelto al ejecutar la opción 1 del menú de la figura 57.

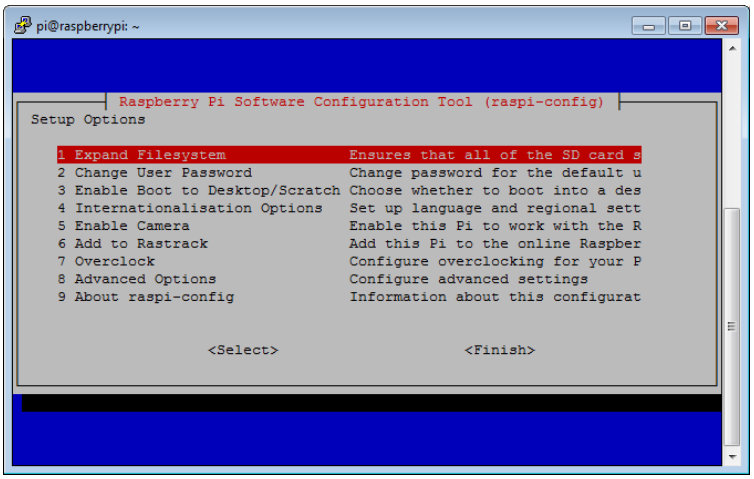

Figura 57 Configuración de herramientas de sistema operativo Raspbian.

.

Figura 58. Finalización de configuración de herramientas de Raspbian.

En la siguiente sesión de inicio al sistema operativo, se ejecutan dos comandos básicos que en todo sistema LINUX es necesario realizar antes de instalar aplicaciones, estos comando son update y upgrade. La figura 59 muestra la forma de ejecutar ambos comandos desde la Shell de LINUX.

```
pi@raspberrypi ~ $ sudo apt-get update
pi@raspberrypi ~ $ sudo apt-get upgrade
```

Figura 59. Comandos básicos necesarios antes de instalar aplicaciones en sistemas LINUX.

Para establecer la fecha y hora correcta del sistema operativo Raspbian, la sentencia de la figura 60 elige la zona horaria en donde se ubique la placa en el mundo, se necesitará configurar para El Salvador.

```
pi@raspberrypi ~ $ sudo dpkg-reconfigure tzdata
```

Figura 60. Sentencia que establece la fecha y hora correcta en sistemas LINUX.

Al ejecutar la sentencia, se despliega una pantalla de configuración en donde se muestra Geographic area (área geográfica) y Time zone (zona horaria). Para el área geográfica se selecciona América y para la zona de horaria El Salvador, la figura 61 despliega el menú en donde será elegida el área geográfica de América y la figura 62 para la zona horaria de El Salvador.

Esta configuración es necesaria para que todas las imágenes.jpg que están siendo generadas, tengan la fecha y hora correcta, esto es posible porque cuando se accede a

internet, la placa Raspberry Pi se conecta a un servidor NTP (Network Time Protocol / red de protocolo de tiempo); este configura el sistema operativo en la placa con la fecha y hora local correctamente en el punto de control, evitando así que al momento de la generación de vídeos, se muestren los monitores con fechas y horas incorrectas ante un evento de intrusión, y esto es indeseado.

En la figura 63 se muestra el mensaje en la Shell cuando se configura la fecha y hora del sistema operativo en la placa.

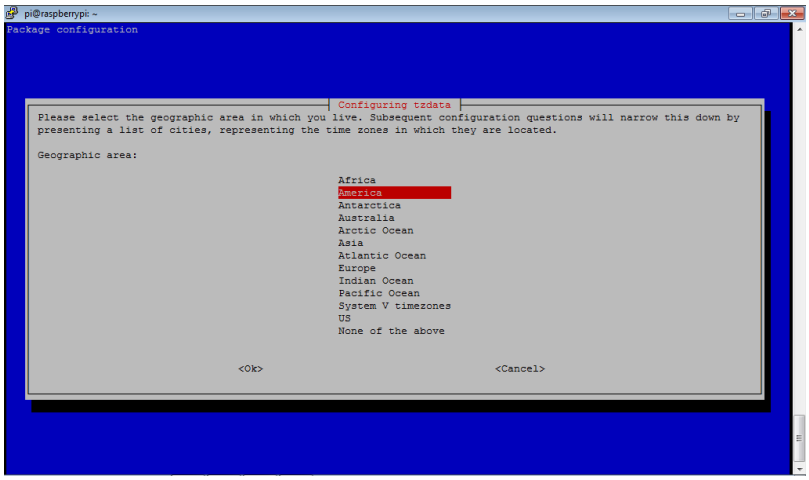

Figura 61. Menú que despliega las áreas geográficas de la tierra.

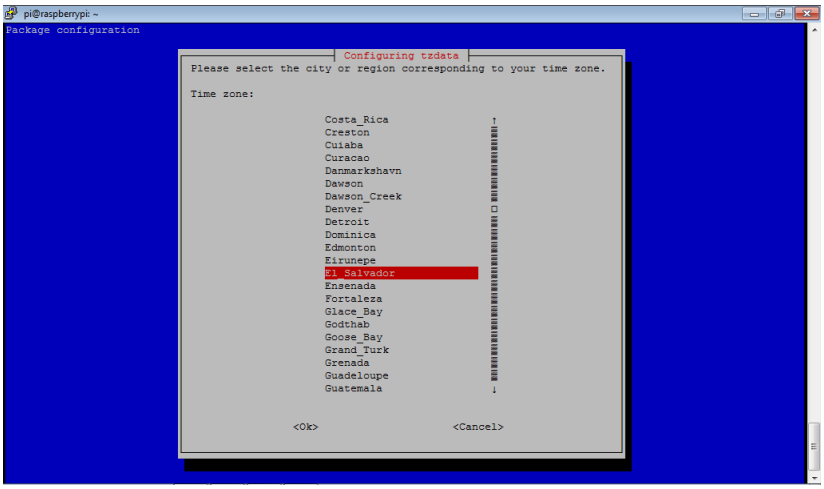

Figura 62. Menú con todas las zonas y lugares de la tierra.

```
Current default time zone: 'America/El_Salvador'
Local time is now:      Tue Aug 12 01:04:29 CST 2014.
Universal Time is now:  Tue Aug 12 07:04:29 UTC 2014.
```

Figura 63. Shell de LINUX que muestra la fecha y hora después de la configuración.

INSTALACIÓN DE SERVIDOR DE VIDEO ZONEMINDER.

Para la correcta instalación del servidor de video zoneminder, existe un buen blog que realiza la instalación paso a paso[36].

Los pasos que se detallan a continuación, es la forma particular para el proyecto de seguridad y está basado en el blog sugerido en el párrafo anterior.

Como primer paso, se conecta una cámara web al puerto USB de la Raspberry Pi. Para ello es necesario ver qué tipos de cámara soporta Raspbian. Existe una página wike[37] que detalla las cámaras reconocidas por la Raspberry Pi. En su defecto, el comando lsusb proporciona si la cámara es reconocida como se aprecia en la figura 64.

```
pi@raspberrypi ~ $ lsusb
Bus 001 Device 002: ID 0424:9512 Standard Microsystems Corp.
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 001 Device 003: ID 0424:ec00 Standard Microsystems Corp.
Bus 001 Device 004: ID 045e:0770 Microsoft Corp.
```

Figura 64. Verificación de reconocimiento de cámara web

En la figura 64, se reconoce el dispositivo 004 como la cámara web.

Luego se procede a la instalación de todos los paquetes necesarios, librerías etc. Se accede como súper usuario figura 65.

```
pi@raspberrypi ~ $ sudo su
root@raspberrypi:/home/pi#
```

Figura 65. Acceso como súper usuario.

Primeramente, se deben instalar los servicios necesarios y algunos otros paquetes para una mejor administración, además se instala el servidor apache con todas sus dependencias, así como el gestor de base MySQL y PHP con sus dependencias. Las dos sentencias que se muestran en la figura 66, cumplen con ese objetivo.

```
root@raspberrypi:/home/pi# aptitude install apt-listbugs apt-file gpm vim less rcconf openssh-server

root@raspberrypi:/home/pi# aptitude install apache2 libapache2-mod-php5 libapache2-mod-auth-mysql php5 php5-mysql mysql-server
 mysql-client
```

Figura 66. Paquetes necesarios antes de instalar el servidor de vídeo.

[36] http://debianyderivadas.blogspot.com/2010/08/instalacion-y-configuracion-de.html
[37] http://elinux.org/RPi_USB_Webcams

Cuando en el proceso de instalación de MySQL, se muestre la ventana de la figura 67 se asigna una contraseña de acceso para el usuario root.

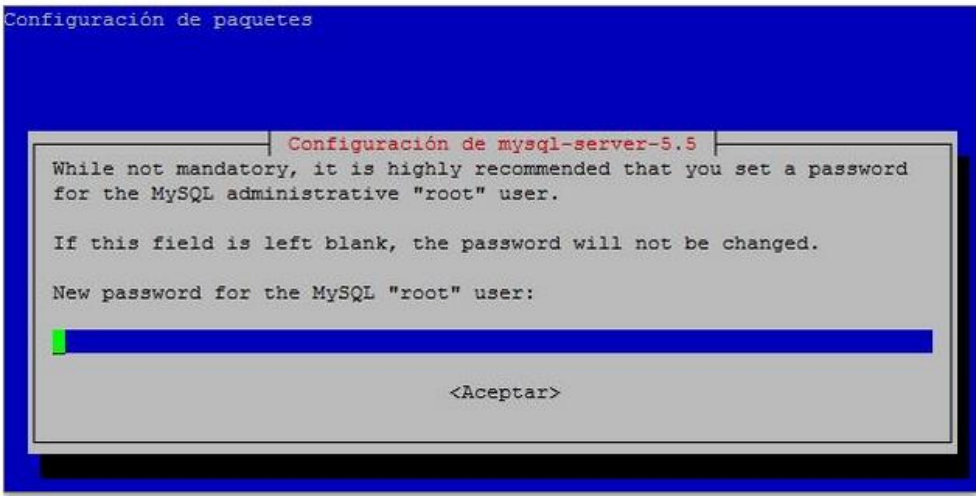

Figura 67. Asignación de contraseña al usuario root.

Luego se instalará el conjunto de aplicaciones de ffmpeg y el servidor de vídeo zoneminder como lo sugiere la figura 68. Estas dos aplicaciones son básicas para el sistema de seguridad. La sentencia de zoneminder instalará el servidor y todas las dependencias, así como la creación de la base de datos ZM.

```
root@raspberrypi:/home/pi#  aptitude install ffmpeg
root@raspberrypi:/home/pi# aptitude install zoneminder
```

Figura 68. Instalación de dependencias ffmpeg y el servidor de vídeo zoneminder.

Se configura mail-transfer-agent nullmailer, para envio de email, con el nombre por defecto en la instalación mostrado en la figura 69.

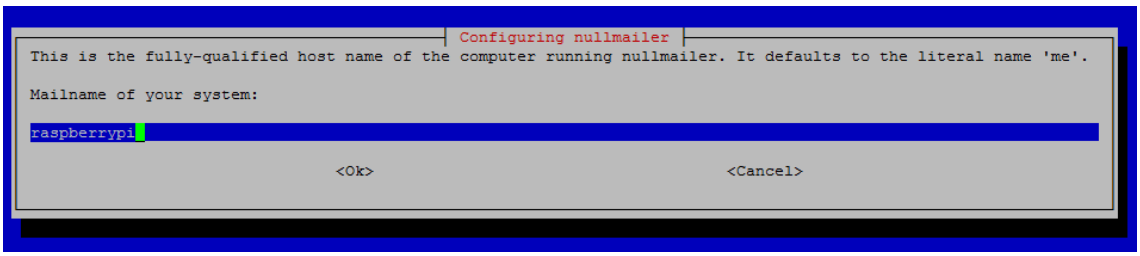

Figura 69. raspberrypi, nombre por defecto durante la instalación.

Las siguientes dos ventanas emergentes durante la instalación, se dejarán tal cual se sugiera en el proceso, estas ventanas son las presentadas en la figura 70 y la figura 71.

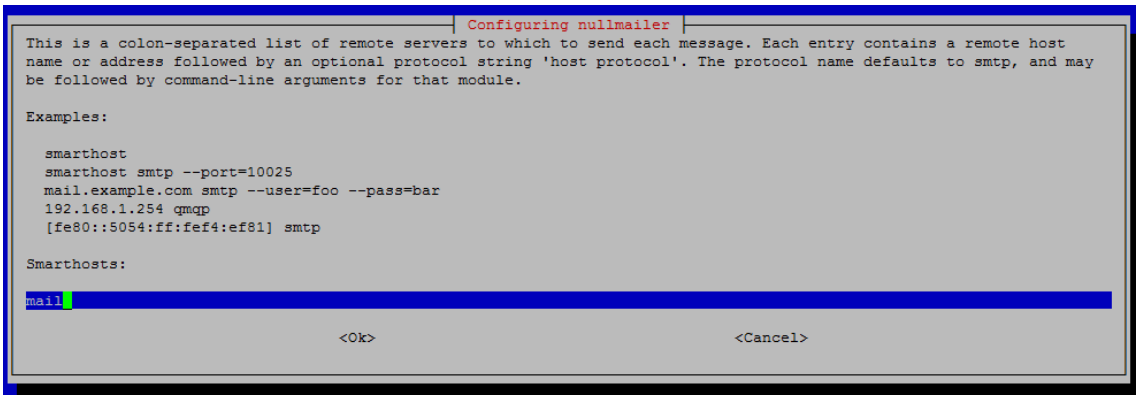

Figura 70. nullmailer configurado con nombre mail.

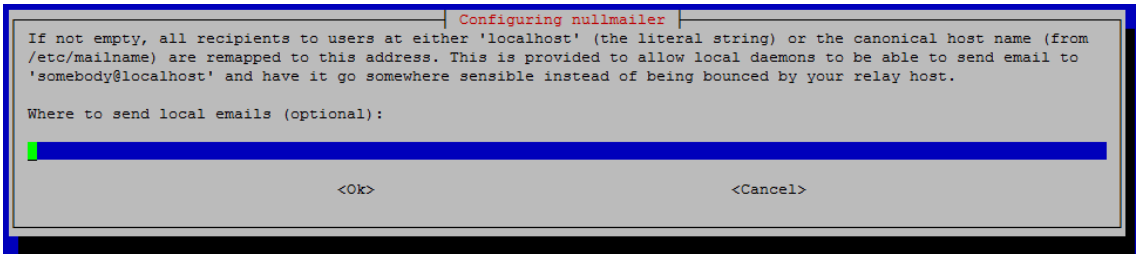

Figura 71. Ventana opcional para la configuración de email con campo vacío.

Para reconfigurar dicho servicio se escribe la siguiente sentencia que se muestra en la figura 72.

```
pi@raspberrypi ~ $ sudo dpkg-reconfigure nullmailer
```

Figura 72. Reconfiguración de nullmailer.

Configuración de zoneminder:

Agregar el usuario www-data con la sentencia de la figura 73 al grupo vídeo.

```
root@raspberrypi:/home/pi# gpasswd -a www-data video
```

Figura 73. agregando el usuario www-data.

Agregar un alias en apache para poder acceder a ZoneMinder desde el servidor web y se reinicia o se recarga la configuración de apache para que los cambios tengan efecto como se muestra en la figura 74.

```
root@raspberrypi:/home/pi# ln -s /etc/zm/apache.conf /etc/apache2/conf.d/zoneminder.conf
root@raspberrypi:/home/pi# /etc/init.d/apache2 restart
```

Figura 74. Agregando alias a ZoneMinder.

Se configuran los permisos adecuados a /usr/bin/zmfix descrito en la figura 75, para que pueda ser leído y ejecutado por todos los usuarios.

```
root@raspberrypi:/home/pi# chmod 4755 /usr/bin/zmfix
root@raspberrypi:/home/pi#  zmfix -a
```

Figura 75. Agregando permiso a zmfix.

Se asigna a www-data como el propietario del directorio temporal de ZoneMinder, la sentencia mostrada en la figura 76 realiza la asignación.

```
root@raspberrypi:/home/pi# chown www-data.www-data /usr/share/zoneminder/temp
```

Figura 76. www-data como propietario temporal.

Además, editar el archivo "/etc/sysctl.conf" y agregar las siguientes líneaskernel.shmall = 134217728 y kernel.shmmax = 134217728 (para 128MB de memoria compartida). La figura 77 muestra la forma de llamar sysctl.conf y la figura 78 muestra el archivo editado.

```
root@raspberrypi:/home/pi# nano /etc/sysctl.conf
```

Figura 77. Accediendo al archivo sysctl.conf desde el editor de texto nano.

```
  GNU nano 2.2.6                          File: /etc/sysctl.conf

# net.ipv4.conf.all.secure_redirects = 1
#
# Do not send ICMP redirects (we are not a router)
#net.ipv4.conf.all.send_redirects = 0
#
# Do not accept IP source route packets (we are not a router)
#net.ipv4.conf.all.accept_source_route = 0
#net.ipv6.conf.all.accept_source_route = 0
#
# Log Martian Packets
#net.ipv4.conf.all.log_martians = 1
#

# rpi tweaks
vm.swappiness=1
vm.min_free_kbytes = 8192

kernel.shmall = 134217728
kernel.shmmax = 134217728
```

Figura 78. Archivo de texto sysctl.conf redimensionado 128 MB editando las líneas kernel.shmall y kernel.shmmax.

Se descarga, descomprime y se copia cambozola al directorio “/usr/share/zoneminder/” para ver el streaming de las cámaras, las sentencias mostradas en la figura 79 al ser ejecutadas, se tiene una api Java cambozola en la carpeta de zoneminder. El streaming cambozola[38] es una Api que permite la un flujo de imágenes estáticas.

```
root@raspberrypi:/home/pi# wget http://www.charliemouse.com:8080/code/cambozola/cambozola-latest.tar.gz
-rw-r--r-- 1 www-data www-data 234516 Oct 22  2013 cambozola-latest.tar.gz
root@raspberrypi:/home/pi# tar -xvzf cambozola-latest.tar.gz
root@raspberrypi:/home/pi# cp cambozola-0.935/dist/cambozola.jar /usr/share/zoneminder/
```

Figura 79. Copiando streaming Image Viewer.

Luego, se habilita el cliente de streamingcambozola desde las opciones de la consolade administración de ZoneMinder (Options / Images / OPT_CAMBOZOLA) de la pestaña Images.

ZoneMinder usa por default un usuario y contraseña débil para las query con la base de datos. Por ello, se recomienda cambiar el usuario y contraseña que usa ZoneMinder. Esto se hace editando el archivo cuya sentencia es la que se describe en la figura 80 "/etc/zm/zm.conf" y modificando el contenido del archivo las directivas ZM_DB_USER y ZM_DB_PASS como se detalla en la figura 81.

```
root@raspberrypi:/home/pi# nano /etc/zm/zm.conf
```

Figura 80. Accediendo al archivo de configuración zm.conf.

```
# ZoneMinder database name
ZM_DB_NAME=zm

# ZoneMinder database user
ZM_DB_USER=usuario_nuevo

# ZoneMinder database password
ZM_DB_PASS=password_nuevo
```

Figura 81. Reasignando nombre de usuario y password de la base de dato zm.

Una vez modificado el archivo zm.conf se crea el usuario con su password en MySQL, la figura 82 muestra la manera de hacerlo y asigna todos los permisos de la DB zm.

[38]http://www.charliemouse.com/code/cambozola/

```
root@raspberrypi:/home/pi# mysql -u root -p

mysql> GRANT all ON zm.* to usuario_nuevo@localhost IDENTIFIED BY 'password_nuevo';

mysql> exit;
Bye
root@raspberrypi:/home/pi#
```

Figura 82. Asignando permisos de la base de datos zm al usuario nuevo.

Otra configuración que se puede modificar es la ruta donde se guardarán los eventos e imágenes (quizá otro disco duro) modificando el enlace simbólico para que apunte a la nueva ruta, en la figura 83 se muestra las carpetas a borrar, mientras que en la figura 84 se tienen las nuevas carpetas en una nueva ruta, la figura 85 asigna un enlace simbólico a las carpetas nuevas desde la carpeta share de LINUX, en la figura 86 se asigna a la ruta nueva como propietario de www-data.

```
root@raspberrypi:/home/pi#  rm /usr/share/zoneminder/events
root@raspberrypi:/home/pi#  rm /usr/share/zoneminder/images
root@raspberrypi:/home/pi#  rm /usr/share/zoneminder/temp
```

Figura 83. Borrando carpetas las carpetas events, images y temp de la ruta indicada.

```
root@raspberrypi:/home/pi#  mkdir /ruta/nueva/images
root@raspberrypi:/home/pi#  mkdir /ruta/nueva/events
root@raspberrypi:/home/pi#  mkdir /ruta/nueva/temp
```

Figura 84. Creando nuevamente las carpetas eliminadas en una nueva ruta, puede ser un disco duro externo.

```
root@raspberrypi:/home/pi# ln -s /ruta/nueva/events /usr/share/zoneminder/events
root@raspberrypi:/home/pi# ln -s /ruta/nueva/images /usr/share/zoneminder/images
root@raspberrypi:/home/pi#  ln -s /ruta/nueva/temp /usr/share/zoneminder/temp
```

Figura 85. Creando los enlaces simbólicos de las carpetas events, images y temp hacia la ruta nueva.

```
root@raspberrypi:/home/pi# chown www-data.www-data -R /ruta/nueva/
```

Figura 86. La /ruta/nueva/ es la nueva propietaria de www-data.www-data.

Si ya se han agregado monitores, antes de eliminar los enlaces simbólicos hay que parar el servicio de ZoneMinder. Una vez creados los enlaces simbólicos se mueve el contenido de los directorios "/var/cache/zoneminder/" a la nueva ruta y se arranca el servicio ZoneMinder.

IMPORTANTE: Cuando se actualiza ZoneMinder mediante el sistema de paquetes, se eliminarán los enlaces que se acaban de crear y se crearán los que vienen por default. Por lo que hay que repetir los pasos, pero ANTES hay que tener mucho CUIDADO en mover los eventos de "/ruta/nueva/events" fuera de dicha ruta porque de lo contrario se eliminarán todos los eventos cuando se inicie ZoneMinder.

Conceptos e Información importante.

- Un monitor es la configuración de una cámara ya sea para monitorizar o grabar imágenes. Una vez configurado, si se hace de manera correcta, se habilita y la cámara es accesible y se mostrará de color verde, de lo contrario se mostrará de color rojo o anaranjado.
- La consola de ZoneMinder de la figura 87 es la interfaz Web.

Figura 87. Consola principal del servidor ZoneMinder desde un cliente web.

- Un evento es el registro generado por una cámara cuando se graban imágenes.
- Por default, cada 10 minutos (puede modificarse) se genera un nuevo evento que contiene las imágenes de la grabación.
- Las imágenes de los eventos se guardan en la carpeta "/events/".
- Los filtros sirven para mostrar determinados eventos en base a parámetros configurables (fecha, causa, porcentaje en disco, id del monitor, etc). Viene incluido un filtro por default llamado PurgeWhenFull que elimina de manera automática los eventos antiguos cuando el disco duro llega al porcentaje configurado.

Configuración de las Opciones (Options) desde la consola de ZoneMinder

Se accede a estas opciones desde el link "options" de la consola principal de ZoneMinder, figura 85. Si se realizan varias modificaciones se recomienda reiniciar ZoneMinder para evitar inconsistencia o pérdida de datos. Se recomienda modificar al menos las siguientes opciones que se muestran desde la figura 86 a la figura 88 y las pestañas que se vayan indicando en los siguientes párrafos.

Figura 88. Imagen amplificada de la zona opción de la consola principal.

En la pestaña "System" figura 89:

- La opción OPT_USE_AUTH(opción de autentificación de usuario) debe marcarse para permitir autenticación de usuarios. Al guardar pedirá usuario y contraseña que por defecto son admin/admin. También aparecerá una pestaña llamada "Users" desde la cual se puede modificar la clave del usuario admin y crear nuevos usuarios con los permisos deseados.

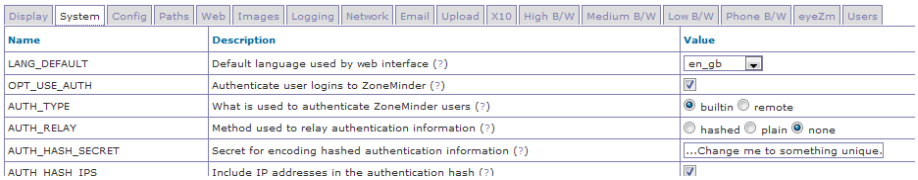

Figura 89. Pestaña System con las configuraciones respectivas.

- La opción LANG_DEFAULT (Lenguaje por defecto), permite modificar el idioma. Se cambia al idioma al español (es_ar). Gran parte no está traducido al español, para continuar traduciendo hay que editar el archivo "/usr/share/zoneminder/lang/es_ar.php".

En la pestaña Paths (Enlaces) figura 90, se pueden modificar las rutas donde se almacenaran imágenes, sonidos etc.

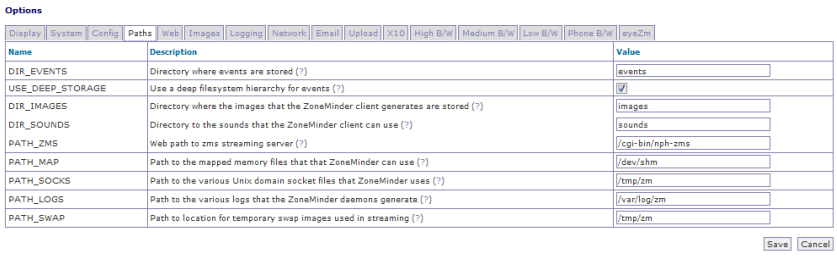

Figura 90. Pestaña paths con todas las configuraciones necesarias.

- En la pestaña "Web" figura 91

WEB_RESIZE_CONSOLE Desmarcarlo.

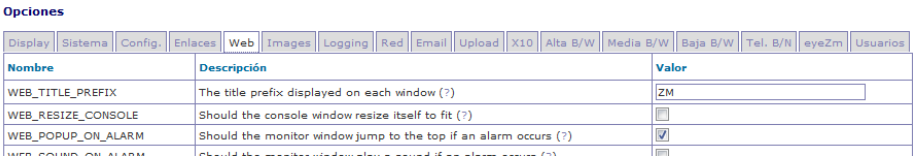

Figura 91. *Pestaña Web desmarcar WEB_RESIZE_CONSOLE.*

- En la pestaña "Images"

  OPT_CAMBOZOLA marcarlo.

  PATH_CAMBOZOLA cambozola.jar (donde fue guardado anteriormente).

  STREAM_METHOD Dejar jpeg.

  PATH_FFMPEG Dejar /usr/bin/ffmpeg

Creación de usuarios.

En la pestaña "Users" (Usuarios). Para que aparezca esta pestaña debe marcarse la opción "OPT_USE_AUTH" desde la pestaña "System".

- Se modifica el password de admin dando clic en el nombre del usuario. Se recomienda no eliminar la cuenta admin al menos hasta que sea creada otra cuenta con todos los permisos.
- Se crean más cuentas de usuario.
- Pueden crearse usuarios con los permisos siguientes:
  - Lenguaje para esa cuenta de usuario en particular.
  - Stream (permite ver video en vivo de las cámaras.
  - Events (permite ver o modificar o eliminar eventos).
  - Control (permite controlar cámaras).
  - Monitors (permite ver y editar monitores).
  - System(determina si el usuario puede ver o modificar las configuraciones del sistema. Como la de usuarios o del sistema completo).
  - Ancho de banda.
  - Restringir a determinados monitores.

Configuración de Monitores.

La configuración de monitores, será siempre desde la consola de zoneminder. En el botón Agregar Nuevo Monitor. Si se desea modificar las configuraciones, se sigue el enlace desde

la opción Origen. En la figura 92 se ve el aspecto de los monitores agregados, su función y origen.

Figura 92. Consola que muestra la habilitación de un monitor Electrica_planta_1.

Se acceden a las configuraciones de monitor. En ella se encuentran 5 pestañas que serán configurables para cada monitor.

La figura 93 muestra todos los campos configurables para el nuevo monitor entre los que se tienen el nombre, el Tipo origen, Función, Habilitado, etc. básicamente los parámetros importantes son el Nombre, Tipo Origen, función y f.p.s

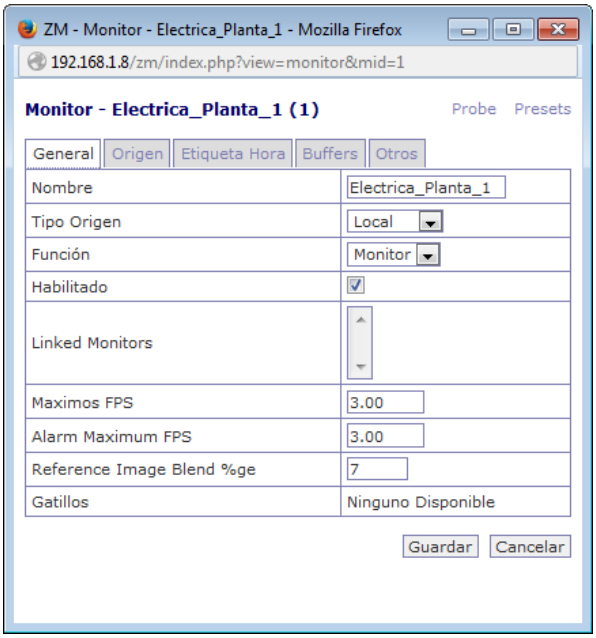

Figura 93. Configuración de la pestaña General.

De la figura 95, pestaña Origen, las configuraciones más importante son: la Device Path (ruta del dispositivo), el tipo de señal y la resolución en pixeles. La pestaña Etiqueta hora mostrada en la figura 94, configurará la etiqueta que está siendo mostrada sobre el flujo de imágenes.

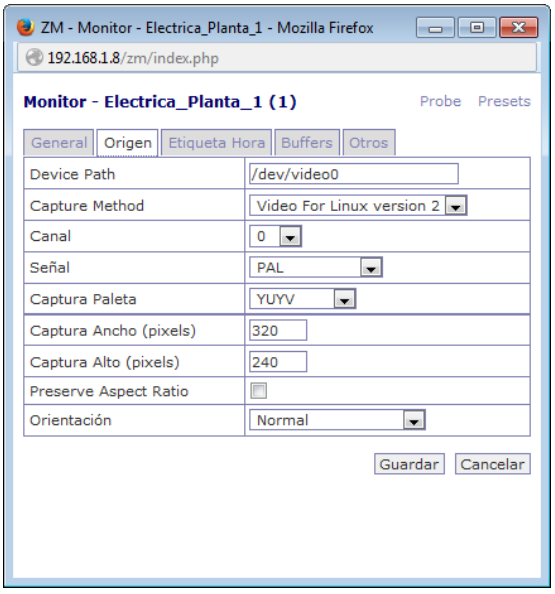

Figura 95. Configuración de la pestaña Origen.

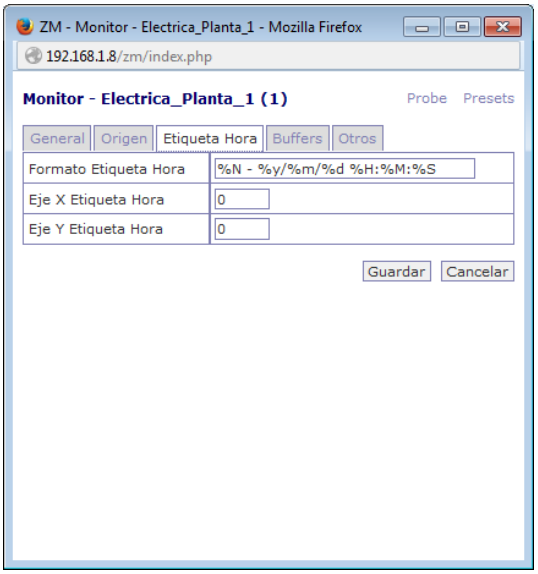

Figura 94. Configuración de la pestaña Etiqueta.

La pestaña Buffers que se detalla en la figura 97, simplemente configura aspectos relacionados al flujo de imágenes y la pestaña Otros mostrada en la figura 96, dentro de la característica principal es darle la longitud de Sección a las imágenes almacenadas, así como el salto de cuadros para que la base de datos no crezca demasiado.

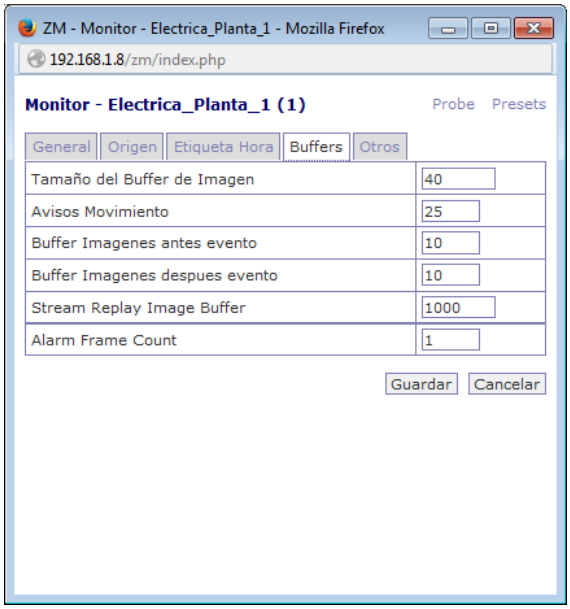

Figura 97. Configuración de la pestaña Buffers.

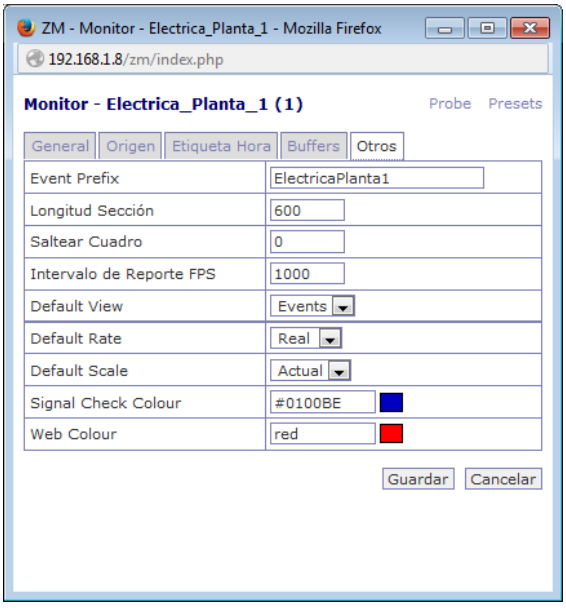

Figura 96. Configuración de la pestaña otros.

Asignación IP-Estática y configuración MiniDongle USB

La configuración de la Mini Dongle USB está sujeta a las especificaciones del fabricante ya que el dispositivo debe especificar la compatibilidad con el kernel de LINUX con versión 2.6 o versiones de Kernel LINUX actuales, en términos de sistema operativo que se cuenta con las versiones posibles a instalar en una Raspberry Pi, para el caso se inserta la mini dongle al puerto USB y para comprobar la compatibilidad se utiliza el comando lsusb como se muestra en la figura 98.

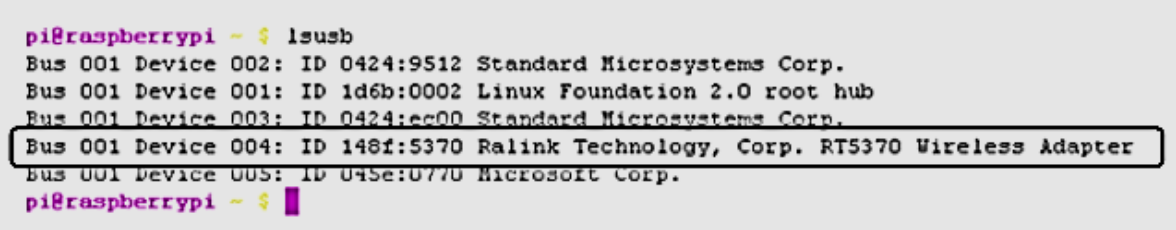

Figura 98. . *Comprobación de reconocimiento de mini Dongle USB.*

De figura 98 se observa que en el Bus 001 Device004, se identifica ID y se muestra una breve descripción del fabricante, modelo. En la figura 99 muestra que al digitar el comando ifconfig se detallan los adaptadores de red que poseen en la placa Raspberry Pi, además, se observa un eth0, propio de la placa Raspberry PI, lo que se conoce comúnmente como localhost, y por último se muestra la wlan0 que es el adaptador inalámbrico.

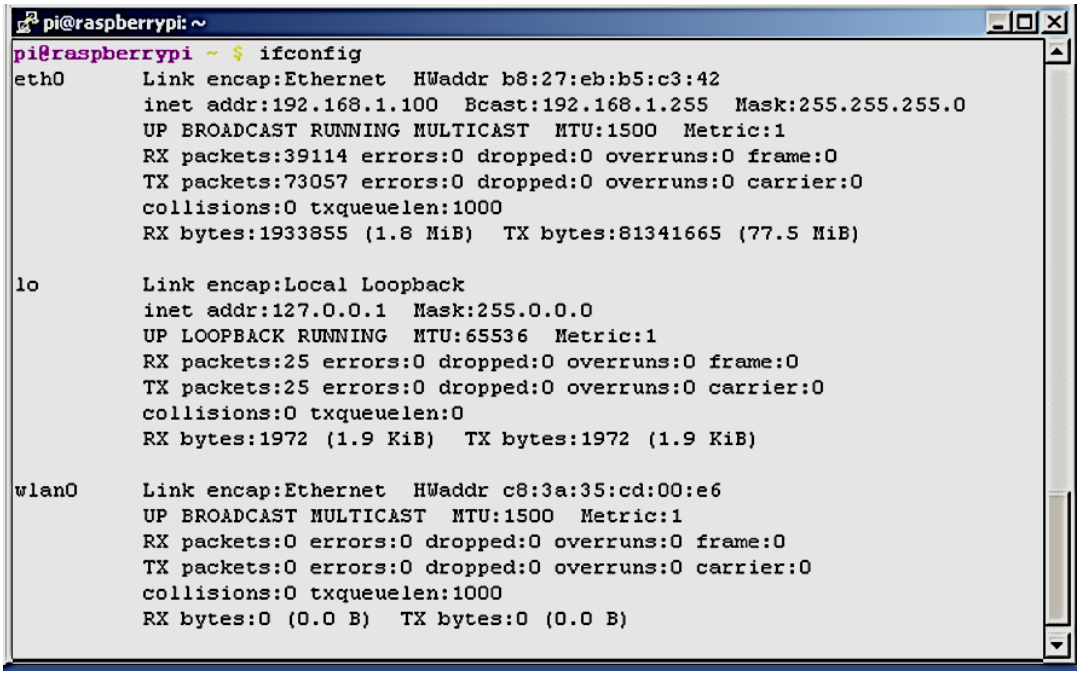

Figura 99. Comando ifconfig que muestra los adaptadores de red para la placa Raspberry Pi

Aún falta asignar una dirección IP estática, para el adaptador de red inalámbrica como puede observarse en la figura 100, se está utilizando el eth0, es necesario editar el archivo interfaces, para ello se utiliza el editor de texto nano con permiso de súperusuario y su la ruta se muestra en la figura 101.

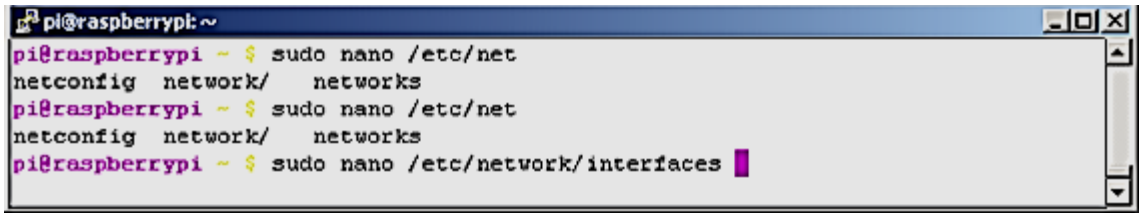

Figura 100. Abriendo archivo interfaces para su edición.

Al ejecutar la línea sudo nano /etc/network/interfaces de la figura 100, se muestra lo que contiene el archivo interface. Utilizando el editor nano en la figura 101 se aprecia primeramente que fue comentada con numeral la línea: iface default inet dhcp.Que ya ha sido digitada cambiando el dhcp por static, luego la dirección IP, mascara de red, puerta de enlace deben ser conforme a la red inalámbrica a trabajar.

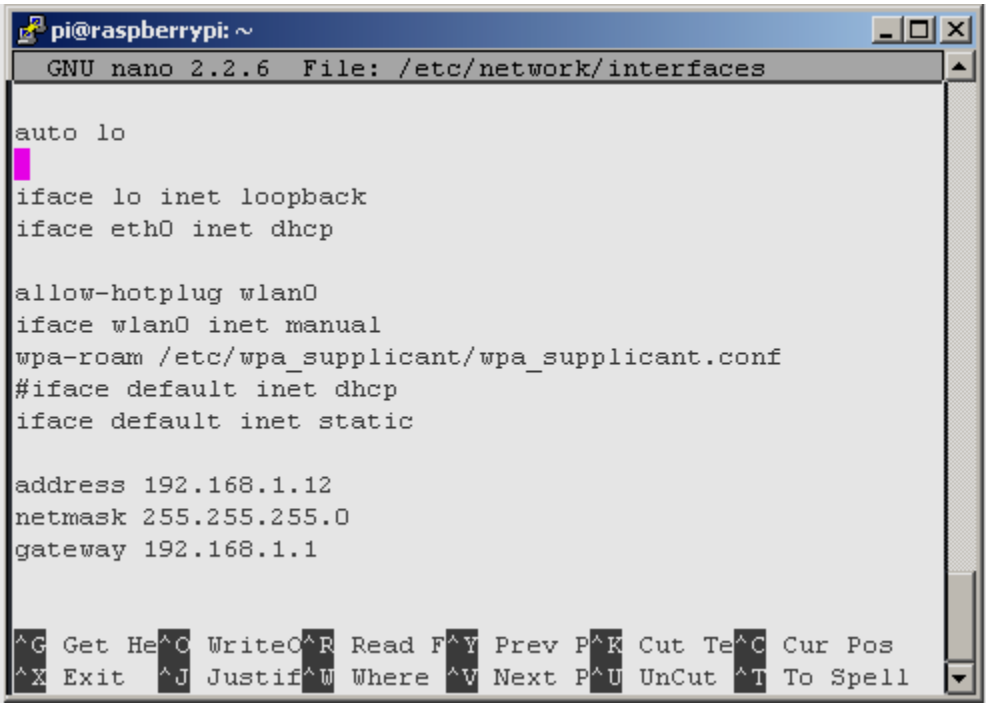

Figura 101. Configuración de archivo interfaces para la asignación de IP estática.

Se guarda los cambios realizados al archivo ya editado con Ctrl + O; y luego con Ctrl + Xse cierra el editor de texto

Hasta este punto solo falta descifrar la clave WPA de la red inalámbrica utilizando el comando que se muestra en la figura 102 con permiso de súperusuario; realizando una exploración con las redes que se encuentren.

```
pi@raspberrypi ~ $ sudo iwlist wlan0 scan
```

Figura 102. *Exploración de a través del comando scan.*

Luego de haber obtenido la lista de las redes inalámbrica, seidentifica la red propia para la placa Raspberry PI.Mostrada en la figura 103.

```
Cell 05 - Address: C8:3A:35:01:D2:F0
          Channel:9
          Frequency:2.452 GHz (Channel 9)
          Quality=69/70  Signal level=-41 dBm
          Encryption key:on
          ESSID:"TESIS"
          Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s; 18 Mb/s
                    24 Mb/s; 36 Mb/s; 54 Mb/s
          Bit Rates:6 Mb/s; 9 Mb/s; 12 Mb/s; 48 Mb/s
          Mode:Master
          Extra:tsf=00000000ac77b560
          Extra: Last beacon: 670ms ago
          IE: Unknown: 00055445534953
          IE: Unknown: 010882840B162430486C
          IE: Unknown: 030109
          IE: Unknown: 2A0100
          IE: Unknown: 2F0100
          IE: Unknown: 32040C121860
          IE: Unknown: 2D1A7E181BFF000000000000000000000000000000000000000000000000
          IE: Unknown: 3D16090F040000000000000000000000000000000000000000
          IE: Unknown: DD0900101802003F02C0000
          IE: WPA Version 1
              Group Cipher : CCMP
              Pairwise Ciphers (1) : CCMP
              Authentication Suites (1) : PSK
          IE: Unknown: DD180050F2020101000003A4000027A4000042435E0062322F00
```

Figura 103. *Identificación propia de red a utilizar.*

Después de tener la red identificada, se utiliza el siguiente comando: wpa_passphrase.En esa línea digitar el ESSID de la Red y la contraseña de la red, este comando devolverá en la terminal el dato descifrado de la red inalámbrica como lo describe la figura 104, se copia desde network hasta el cierre de las llaves hacia el archivo wpa_supplicant.conf.

```
pi@raspberrypi ~ $ wpa_passphrase TESIS udoo2014
network={
        ssid="TESIS"
        #psk="udoo2014"
        psk=340e17e029badc0195bc0a824371f07cf147ebcb69c6619797e36e00116c8cbf
}
```

Figura 104. *Descifrando la red inalámbrica con el comando wpa_passphrase.*

Se digita en la SHELL la sentencia de la figura 105 para abrir y editar el archivo wpa_supplicant.conf.

```
pi@raspberrypi ~ $ sudo nano /etc/wpa_supplicant/wpa_supplicant.conf
```

Figura 105. *Abriendo archivo wpa_suplicant para la edición de la clave psk.*

En la figura 106 se agregan los datos mostrados en la figura 104 quedando el archivo wpa_supplicant.conf con las líneas devueltas por wpa_passphrase.

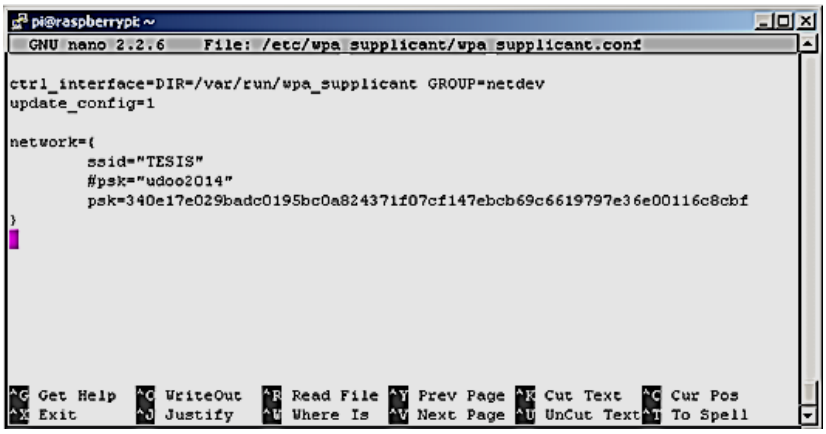

Figura 106. *Contenido de wpa_supplicant.conf con el ssid, #psk y psk añadidos*

Para finalizar solo falta revisar a que adaptador tiene la ruta o dirección de IP de enlace, como se muestra en la figura 107 se muestra que la ruta está asignada al adaptador eth0 y se debe cambiar al adaptador wlan0.

```
pi@raspberrypi ~ $ route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         192.168.1.1     0.0.0.0         UG    0      0        0 eth0
192.168.1.0     *               255.255.255.0   U     0      0        0 eth0
pi@raspberrypi ~ $
```

Figura 107. *Resultado de la ejecución del comando route .*

Luego se digita en la Shell el comando que agrega la ruta hacia wlan0, en este caso dice red irreconocible como se detalla en la figura 108, esto se debe a que estásiendo utilizando el eth0, para corregir el error, es necesario reiniciar o apagar la placa Raspberry Pi, desconectar el cable de red Ethernet y en la siguiente sesión asegurarse que la mini dongle esté conectada al puerto USB, se enciende la Raspberry Pi y se deberá conectar de manera inalámbrica.

```
pi@raspberrypi ~ $ sudo route add default gw 192.168.1.1 dev wlan0
SIOCADDRT: Network is unreachable
```

Figura 108. Agregando la ruta wlan0.

DETALLE DE LOS ELEMENTOS QUE FORMARAN PARTE DEL SISTEMA DE SEGURIDAD PARA LOS PUNTOS DE CONTROL.

Partiendo de la sección del diseño metodológico del sistema, la figura 11 muestra todos los componentes que estarán conectados por medio de un cable ribbon a la placa Raspberry Pi. En esta sección se detalla los elementos que envían señales y reciben señales desde y hacia la Raspberry Pi.

El primer elemento de importancia es la interfaz electrónica entre la placa y los sensores, Teclado, sirena y servomotores. Esta interfaz electrónica es necesaria porque se tienen valores de voltaje diferente que los sensores y elementos proporcionan; estos niveles son de 12 V para el sensor de vibración, 5 voltios para el teclado, sirena y servomotores, 3.3 voltios para el sensor PIR y 0V (gnd) para el sensor magnético. Básicamente la función de dicha interfaz el acondicionar las señales para que la Placa Raspberry PI solo reciba señales de voltajes de 3.3V o proporciones señales con el mismo valor de voltaje. Para manejar los valores de voltaje de 12 y 5 voltios, su usa un opto acoplador PC817 que aísla los 3.3 voltios de E/S en el header GPIO de la placa.

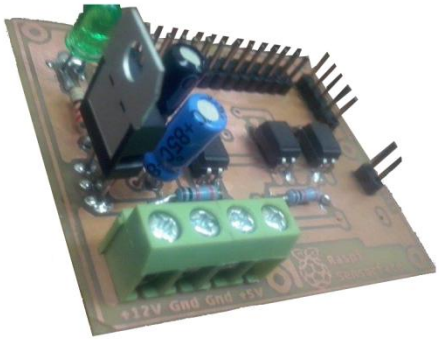

Figura 109. Tarjeta PCB realizada para interfaz electrónica.

En la figura 110 se muestra la interfaz electrónica ya terminada y se conoce como punto de control. Compuestopor caja plástica, acrílico, tarjeta electrónica PCB ypuertosRJ11, además se ve debidamente rotulada.

Figura 110. Punto de control compuesto por caja plástica, acrílico, tarjeta electrónica PCB y puertos RJ11.

En la figura 111 se muestra como quedan los servomotores montados sobre los soportes, además de la cámara web, adicionalmente; la cámara web, los servomotores y los soportes brackets en su conjunto emulan una cámara IP

Figura 111. Servomotores sobre brackets, cámara web, el conjunto montado sobre caja plástica para proyectos electrónicos.

La figura 112 muestra el sensor PIR montado sobre una caja modular para ser instalado en el lugar conveniente.

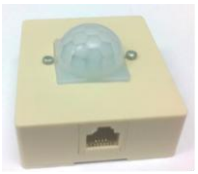

Figura 112. Sensor PIR montado sobre caja modular telefónica.

El sensor magnético con el conector rj11 hembra respectivo se muestra en la figura 113.

Figura 113. Sensor magnético con conector hembra RJ11.

El sensor de vibración con cable rj11 adaptado se muestra en la figura 114.

Figura 114. Sensor de vibración con cable para ser adaptado al punto de control.

Sirena y el teclado se muestra en la figura 115, resaltando que internamente tiene una modificación en donde el teclado y la sirena son componentes mutuamente excluyentes entre si.

Figura 115. Sirena y teclado en una sola unidad. Pero internamente separada su circuitería.

SITIO DEL SISTEMA

El sistema puede accederse desde cualquier navegador pero principalmente desde Firefox, Google-Crome e Internet Explorer, ya que estadísticamente estos navegadores abarcan alrededor del 89% de los clientes web.

El sitio del proyecto lo forman un conjunto de páginas, las cuales se discutirán a lo largo de esta sección.

La página de inicio del sitio es index.php. En esta página se da la bienvenida a los visitantes y se indica básicamente la necesidad de iniciar sesión y registrarse, para poder acceder al sistema; ya que de lo contrario no podrá ingresar. Esta característica es necesaria por la naturaleza misma del sitio; porque solamente el personal autorizado y registrado en la base de datos tendrá acceso a las páginas interiores del sitio, dependiendo del status que dicho usuario posea.

En la figura 116 se muestra la página de inicio llamada index.php

Figura 116. Página de inicio del sitio[39].

Esta es la puerta de acceso al sistema de seguridad y video de vigilancia. En la parte derecha de la ventana se tiene el formulario para iniciar sesión. El usuario deberá ingresar su nombre y password, tal y como se registró a su ingreso a la base de datos. Al enviar los registros indicados; estos son recibidos por el archivo control.php ubicado en el servidor. Este programa se encarga de hacer la búsqueda respectiva de los datos enviados y si el

[39] 192.168.1.96/tesis2/index.php. sitio en construcción.

usuario y clave existen en la base de datos; habrá iniciado sesión y será redireccionado a una página predefinida. Si por el contrario, el usuario no se encontrará en la base de datos o su password no coincidiera con su usuario, no podrá registrarse y se le redireccionará a otra página indicándole dicha condición.

Una vez el usuario inicie la sesión, pasará por defecto a la página Monitoreo1.php; siendo que esta página visualiza el video y las alarmas de la primera planta de escuela de ingeniería eléctrica, la cual es considerada como la página inicial para el caso del monitoreo de las zonas cubiertas por el sistema.

En la figura 117 se muestra ésta página en donde se ven algunas secciones importantes de describir.

Figura 117. Página Monitoreo.

En esta página muestra el corazón del sistema de seguridad, en la parte superior de la página, se tiene el centro de control de video.

Esta sección accede al video de vigilancia del punto específico, que para el caso de monitoreo1.php, se dijo anteriormente, se trata de la primera planta de la escuela de ingeniería eléctrica. Además a la derecha del video se ven tres opciones de visualización, ya que es posible que en un edificio existan varios puntos de vigilancia y se encuentren varias cámaras conectadas. En este proyecto, se ha implementado el sistema con tres

cámaras de seguridad; cada una ubicada en diferentes puntos, pero pueden ser vistas desde estas páginas seleccionando uno, dos o cuatro monitores.

En la parte inferior de la página, se tienen tres campos importantes. El primero es el reporte de alarmas activas, registradas en la base de datos; información que está permanentemente siendo actualizada desde el servidor. Gracias a la aplicación de la técnica Ajax, en un proceso es transparente al cliente. Estas alarmas aparecen en forma de tabla en el recuadro de la izquierda; aquí se muestran todos los edificios pertenecientes a la Universidad de El Salvador que estarían vigilados por el sistema; en la columna derecha de esta tabla, aparece la cantidad de alarmas activadas en dicho edificio y en caso de no existir alarmas se indica claramente con la palabra ***Ninguna*** como se aprecia en la figura 118.

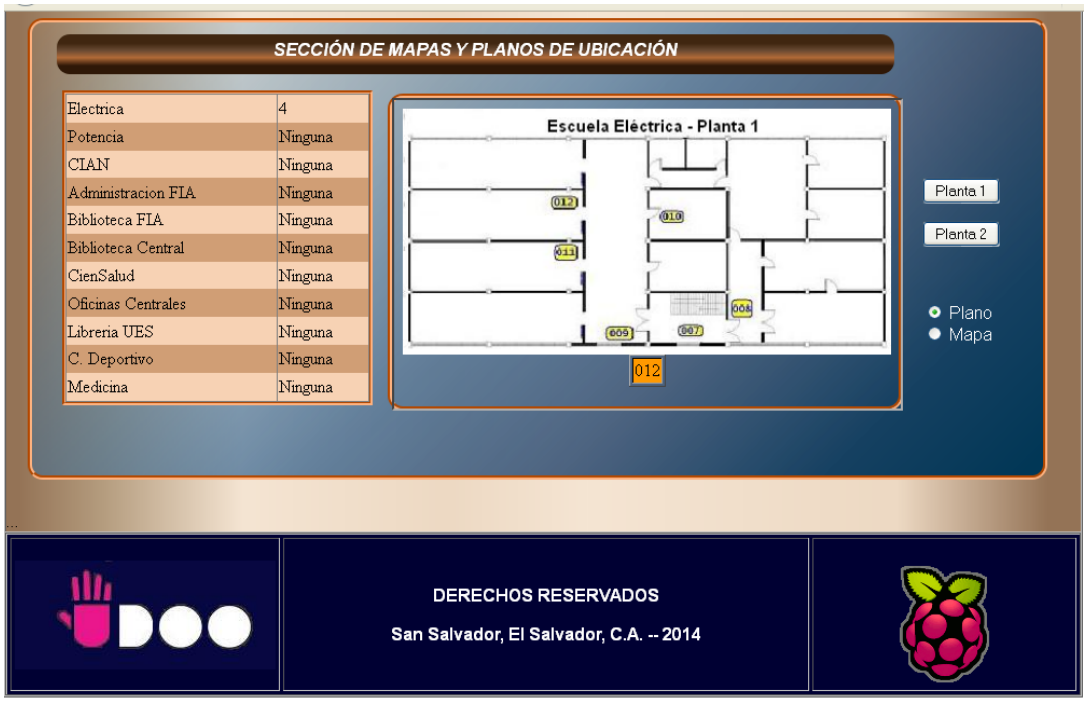

Figura 118. Sección de control y ubicación de las alarma.

En el cuadro de la derecha aparece el plano de ubicación de los diferentes sensores existentes en la zona protegida por el sistema para cada uno de los edificios, según la página en la que se esté ubicado. Aquí en este plano que muestra la infraestructura y distribución interna de los espacios en el edificio, se muestran en su parte inferior los sensores que han sido activados por medio del número mostrado en el pequeño cuadro de color. Este color con el que se muestra el sensor activado, indica el color de la bandera actual de dicho sensor; como se sabe puede ser verde, naranja o rojo. Las banderas verdes no se muestran en el plano y debe entenderse que dicha zona se encuentra sin alarmas.

A la derecha del cuadro de planos, se ubican dos botones por medio de los cuales se puede alternar el plano a visualizar. En el caso de edificios que como el de la escuela de

ingeniería eléctrica, solo poseen dos plantas, estos planos pueden cambiarse presionando el botón respectivo. Luego se tienen dos selectores tipo "*radio buttoms*" (radio botón), estos selectores cambian entre la vista del plano antes mencionado y el mapa georreferenciado que ubica las coordenadas del edificio, de forma que el operador del sistema pueda dirigir rápidamente al cuerpo de custodios que cumplan con la función de reacción ante el disparo de una alerta naranja o una alarma roja. Esto se ilustra en la figura 119.

Figura 119. Ubicación por medio de Georreferenciación.

En caso de no registrarse en el sistema o al no encontrase el usuario en la base de datos, no podrá ingresar al sitio del sistema de seguridad y se mostrará la siguiente página de acceso prohibido al sitio, como muestra la figura 120.

Figura 120. Página de acceso denegado.

La página 121 del sitio, muestra el formulario para crear usuarios y registrarlos en la base de datos. En esta página el administrador del sistema deberá introducir la información requerida en cada uno de los campos, los cuales de muestran en la figura 121.

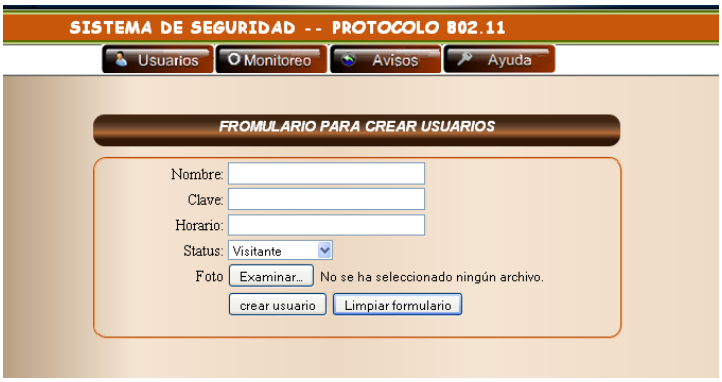

Figura 121. Formulario de registro de usuarios.

Los campos de datos para el registro de usuarios se muestran en la tabla 6:

1	**Nombre**	**En este campo se debe ingresar el nombre del empleado o usuario que tendrá acceso al sistema.**
2	**Clave**	**Es la clave o password de acceso para ingresar al sistema**
3	**Horario**	**Es el horario asignado al empleado como jornada laboral**
4	**Status**	**Se refiere al nivel de privilegios que tendrá el usuario que se está creando**
5	**Foto**	**Debe ser un archivo .jpg con la foto del usuario, con un tamaño aproximado de 300 x 300 píxeles como recomendación.**

Tabla 6. Campos de registro para la página formularios de registro de usuarios de la figura 121.

Una vez introducida toda la información, se oprime el botón crear usuario para enviar los registros a la base de datos, de esta forma el empleado o usuario podrá tener acceso al sistema abriendo una sesión.

La próxima página se abre desde el menú de borrar usuario y nos muestra una lista con todos los usuarios existentes en la base de datos, como muestra la figura 122. En este listado podremos seleccionar al usuario que se eliminará de la base de datos, haciendo click en el Id del usuario, que aparece al izquierda de la tabla.

Figura 122. Lista de usuarios registrados

Como se ve, en este momento aparecen 5 usuarios en total registrados en la base de datos; este dato aparece al final de la tabla. Para borrar un usuario de esta lista, posicionamos el cursor sobre el numero Id del usuario como se muestra a continuación.

Figura 123. Selección para borrar usuarios.

Al dar clic sobre el Id mostrado en la figura 123, se redireccionará a la página Borrar.php, en donde aparece la ficha del usuario con toda la información de éste. Aquí se verifica si efectivamente es el usuario que se desea borrar o no; así se puede regresar para hacer una nueva selección, como muestra la figura 124.

Figura 124. Ficha con los datos del usuario a borrar.