



# **UNIVERSIDAD TÉCNICA ESTATAL DE QUEVEDO**

## **UNIDAD DE POSGRADO**

### **MAESTRÍA EN CONECTIVIDAD Y REDES DE ORDENADORES**

Proyecto de Tesis previa la  
obtención del Grado Académico  
de Magíster en Conectividad y  
Redes de Ordenadores

#### **TEMA:**

**“EL CONTROL Y LAS TELECOMUNICACIONES COMO INFLUENCIA EN EL  
DESARROLLO TECNOLÓGICO DEL HOGAR. IMPLEMENTACIÓN DE UN SISTEMA  
DOMÓTICO MEDIANTE EL CONTROL DE UN TELÉFONO CELULAR CON  
PLATAFORMA ANDROID USANDO TECNOLOGÍA ARDUINO”.**

#### **AUTOR:**

**ING. JOSÉ LUIS VILLARROEL INTRIAGO**

#### **ASESOR:**

**ING. EFRAÍN DÍAZ, MSc.**

**QUEVEDO– LOS RÍOS- ECUADOR**

**2016**

## **CERTIFICACIÓN**

**Ing. Efraín Díaz, MSc.**, Docente Tutor del Proyecto de Tesis, previo a la obtención del Título Académico de Magíster en Conectividad y Redes de Ordenadores

### **C E R T I F I C A**

Que el Ing. José Luis Villarroel Intriago, ha cumplido con la elaboración del Proyecto de Tesis titulado: **“EL CONTROL Y LAS TELECOMUNICACIONES COMO INFLUENCIA EN EL DESARROLLO TECNOLÓGICO DEL HOGAR. IMPLEMENTACIÓN DE UN SISTEMA DOMÓTICO MEDIANTE EL CONTROL DE UN TELÉFONO CELULAR CON PLATAFORMA ANDROID USANDO TECNOLOGÍA ARDUINO.**

El mismo que está apto para la presentación y sustentación respectiva.

A handwritten signature in blue ink, consisting of stylized, overlapping loops and strokes, positioned above a horizontal line.

**Ing. Efraín Díaz, MSc.**

**DOCENTE- ASESOR**

## AUTORÍA

Yo, José Luis Villarroel Intriago, certifico que el proyecto de tesis titulado: **“EL CONTROL Y LAS TELECOMUNICACIONES COMO INFLUENCIA EN EL DESARROLLO TECNOLÓGICO DEL HOGAR. IMPLEMENTACIÓN DE UN SISTEMA DOMÓTICO MEDIANTE EL CONTROL DE UN TELÉFONO CELULAR CON PLATAFORMA ANDROID USANDO TECNOLOGÍA ARDUINO”**, el cual presento como requisito para optar mi **“MAESTRÍA EN CONECTIVIDAD Y REDES DE ORDENADORES”** de la Universidad Técnica Estatal de Quevedo, es el producto de mi labor investigativa. Así mismo doy fe que este trabajo es original e inédito.



AUTOR

## **DEDICATORIA**

Dedico este proyecto de titulación principalmente a nuestro creador y padre de todos nosotros Dios, por haberme dado salud y permitirme haber llegado hasta este momento tan importante de mi formación profesional.

A mi esposa Paola, ya que cada noche luego de una dura jornada de trabajo, junto a mi pequeña Joyce me acompañaban en el duro caminar de la elaboración del proyecto y la documentación respectiva.

A mi madre y padre por ser unos de los pilares más importantes en mi vida y por demostrarme siempre su amor y apoyo incondicional como también porque gracias a ellos sé que la responsabilidad se la debe vivir como un compromiso de dedicación y esfuerzo día a día.

A mis familiares, y muy en especial a mi cuñado David porque sus valiosos conocimientos adquiridos durante su formación educativa profesional, los cuales fueron vitales para haber llegado hasta la meta propuesta.

Y finalmente muchas gracias a todos los que directamente o indirectamente han recorrido este largo caminar conmigo y sienten la satisfacción inmensa de haber logrado esta meta.

## **AGRADECIMIENTO**

Quiero agradecer primero al Espíritu Santo de Dios por haberme dado el don de la inteligencia y guiado a lo largo de este nuevo recorrido hacia el éxito profesional.

A mis padres, por los valores y las enseñanzas que me dieron desde niño para ser lo que ahora soy, una persona sencilla, sincera y honesta.

De igual manera a la Universidad Técnica Estatal de Quevedo, autoridades y profesores por darme la oportunidad de crecer nuevamente profesionalmente y aprender cosas nuevas.

Le agradezco la confianza, apoyo y dedicación de tiempo al ingeniero Efraín Díaz por haberme compartido sus conocimientos y sobre todo por guiarme durante el desarrollo del trabajo de investigación.

Finalmente, agradezco infinitamente a las personas que directa o indirectamente han sido parte del desarrollo del proyecto de titulación e investigación, por su apoyo y colaboración en cada momento solicitado.

**EL AUTOR**

## PRÓLOGO

En estos tiempos existe un gran despilfarro a nivel energético al dejar encendido artefactos, luces en nuestro hogar y de esta manera también nos genera problemas en la economía familiar, como también imaginémonos llegar a nuestro hogar después de un día muy caluroso, agobiante y encontrarnos con un ambiente fresco, agradable y relajado ya que encendimos de manera remota nuestro aire acondicionado.

Por las diferentes razones antes mencionadas se plateo realizar este proyecto el cual radica en la necesidad de mejorar el nivel de vida del ser humano, integrando tecnologías a nivel de hardware y software, de esta manera controlar mediante un dispositivo móvil cada uno de dispositivos en el hogar.

La importancia de este estudio de investigación radica en que aprovechamos herramientas para diseñar, crear y generar productos que ayudan al ser humano mejorando la calidad de vida en el hogar, gracias a la innovación en el acceso a la Internet donde integramos equipos móviles conectados a la red para administrarlos de forma remota.

Con esta propuesta se beneficiara al usuario final en el control de manera remota de dispositivos en el hogar.



---

Ing. José Mauricio Villarroel I.

## RESUMEN

Domótica como concepto se base en la gestión automática del control de dispositivos de manera local o remota en el hogar de esta manera obtener beneficios como lo es el confort, seguridad, ahorro energético y reducción de trabajo manual.

El proyecto de investigación y desarrollo nos muestra un prototipo el cual gestiona el control de dispositivos de manera local o remota en el hogar.

Se elaboró un circuito integrado que está conformado por un módulo Arduino Uno para el procesamiento y accionamiento de señales digitales, un módulo Wifi Tinsy para la comunicación y envío de datos o trama de paquetes hacia el servidor y una tarjeta de control de relays para el accionamiento del encendido o apagado de los diferentes dispositivos a controlarse en el hogar.

Existe una aplicación desarrollada en Visual Basic.Net, la cual es la que lee cada cierto periodo (200 ms) el campo existente en la base de datos el valor de cada dispositivo, es decir el estado actual, encendido o apagado final y lo envía en forma de trama hacia el dispositivo de control Arduino.

Para el almacenamiento de datos y gestión de información se usó Postgresql, en la creación del acceso web se utilizó PHP de esta manera mediante un computador poder ingresar de manera remota, como también podemos hacerlo desde un dispositivo móvil ya que desarrollamos una aplicación en Android Studio.

## **ABSTRACT**

Domotics as a concept is based on the automatic management of the control of devices locally or remotely in the home in this way to obtain benefits such as comfort, security, energy saving and reduction of manual work.

The research and development project shows us a prototype which manages the control of devices locally or remotely in the home.

An integrated circuit consisting of an Arduino Uno module was developed for the processing and activation of digital signals, a TinySine Wifi module for the communication and sending of packet data or frame to the server and a relay control card for the drive on or off the various devices to be controlled in the home.

There is an application developed in Visual Basic.Net, which is the one that reads every time (200 ms) the existing field in the database the value of each device, that is to say the current state, on or final shutdown and sends it in the form of a frame to the Arduino control device.

For data storage and information management was used PostgreSQL, in the creation of web access PHP was used in this way by a computer to be able to enter remotely, as we can do it from a mobile device as we developed an application in Android Studio .

.



## ÍNDICE

PORTADA.....	i
CERTIFICACIÓN .....	ii
AUTORÍA .....	iii
DEDICATORIA.....	iv
AGRADECIMIENTO.....	v
PRÓLOGO .....	vi
RESUMEN .....	vii
ABSTRACT .....	viii
ÍNDICE .....	ix
ÍNDICE DE TABLAS .....	xii
ÍNDICE DE FIGURAS .....	xiii
INTRODUCCIÓN .....	xv
<b>CAPÍTULO I: MARCO CONTEXTUAL DE LA INVESTIGACIÓN .....</b>	<b>16</b>
1.1 Formulación de la Problemática .....	17
1.2 Situación actual del Problema .....	17
1.3 Problema de investigación .....	17
1.3.1 Problema general .....	17
1.3.2 Problema derivado .....	18
1.4 Objetivos .....	18
1.4.1 Objetivos generales .....	18
1.4.2 Objetivos específicos .....	18
1.5 Justificación .....	18
<b>CAPÍTULO II: MARCO TEÓRICO .....</b>	<b>20</b>
2.1 Fundamentación conceptual .....	21
2.1.1 Domótica .....	21
2.1.1.1 Sistemas inteligentes centralizados y descentralizados .....	21
2.1.1.2 Descripción de los sistemas domóticos .....	22
2.2 Fundamentación teórica .....	23
2.2.1 Características básicas de Arduino .....	23

2.2.1.1 Hardware .....	23
2.2.1.1.1 Arduino Uno .....	24
2.2.1.1.2 Modulo Wifi Tinsine .....	25
2.2.1.1.3 Aplicaciones para dispositivos móviles .....	26
2.2.3 Tablet PC o tableta electrónica .....	27
2.2.4 Sistema operativo Android .....	27
2.2.4.1 Historia .....	27
2.2.4.2 Introducción .....	29
2.2.4.3 Definición de Android .....	29
2.2.4.4 Características de Android .....	29
2.2.4.5 Versiones de Android .....	30
2.2.5 Lenguajes de programación .....	31
2.2.5.1 Introducción .....	31
2.2.5.2 Android Studio .....	31
2.2.5.2.1 Principales características de Android Studio .....	32
2.2.5.2.2 Requerimientos para la instalación de Android Studio .....	33
2.2.5.2.3 Ventajas del uso del Android Studio .....	33
2.2.5.3 Php .....	33
2.2.5.4 Visual Basic.Net .....	34
2.2.5.5 SDK de Android .....	34
2.2.6 Base de datos .....	34
2.2.6.1 Introducción .....	34
2.2.6.2 PostgreSQL .....	35
2.2.6.2.1 Características PostgreSQL .....	35
2.2.6.2.2 Ventajas de PostgreSQL .....	35
2.2.7.1 Historia .....	36
2.2.7.2 Apache .....	36
2.2.7.3 Servidor web .....	36
2.2.7.4 Ventajas .....	36

2.2.7.5 Inconvenientes .....	37
<b>CAPÍTULO III: METODOLOGÍA DE LA INVESTIGACIÓN .....</b>	<b>38</b>
3.1 Tipo de investigación .....	39
3.2 Instrumentos de investigación .....	39
3.2.1 Método de observación.....	39
3.3 Costos de materiales propuestos en el desarrollo de la investigación .....	39
3.3.1 Materiales para el desarrollo del sistema domótico .....	39
3.3.1 Presupuesto del nivel de programación .....	40
<b>CAPÍTULO IV: ANÁLISIS E INTERPRETACIÓN DE LOS RESULTADOS .....</b>	<b>41</b>
4.1 Análisis de la tecnología Arduino .....	42
4.1.1 Análisis para la conectividad en la transmisión de datos .....	43
4.2.- Diseño del prototipo electrónico del sistema de control domotico .....	44
4.2.1 Descripción general del sistema .....	44
4.2.2 Esquema y diseño del sistema electrónico .....	45
4.2.2.1 Tarjeta de control o CPU .....	45
4.2.2.2 Módulo Wifi .....	46
4.2.2.3 Módulo de Relays .....	47
4.2.2.4 Fuente de poder .....	47
4.2.3 Acoplamientos de componentes electrónicos .....	48
4.3.- Desarrollo de las herramientas informáticas para la gestión de la información .....	49
4.3.1.- Análisis para el desarrollo de las aplicaciones informáticas y la gestión de la base de datos .....	50
4.3.2.- Diseño del análisis relacional de la base de datos .....	51
4.3.3.- Desarrollo de la aplicación de gestión de la información con el servidor y el esquema electrónico .....	52
4.3.3.- Desarrollo de la aplicación web la para gestión de la información con el servidor y el esquema electrónico .....	53
4.3.3.- Desarrollo de la aplicación móvil la para gestión de la información con el servidor y el esquema electrónico .....	54
<b>CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES .....</b>	<b>58</b>

5.1 Conclusiones .....	59
5.2 Recomendaciones .....	60
Anexo A: Certificación plagio académico sistema Urkund .....	62
Anexo B: Funcionamiento del sistema domótico .....	63
Anexo C: Manual de usuario.....	67
Anexo D: Programación.....	79
Bibliografía .....	105

## ÍNDICE DE TABLAS

Tabla 1: Cuadro del presupuesto de materiales electrónicos.....	39
Tabla 2: Cuadro de presupuesto de las herramientas informáticas usadas .....	40
Tabla 3: Cuadro comparativo de la tecnología modular Arduino.....	42
Tabla 4: Cuadro comparativo de las diferentes tipos de comunicaciones inalámbricas .....	43
Tabla 5: Cuadro comparativo de los módulos Wifi .....	44
Tabla 6: Cuadro comparativo de las herramientas para el desarrollo de software.....	50
Tabla 7: Cuadro comparativo de las herramientas para el desarrollo de base de datos.....	50
Tabla 8: Caso de uso: Inicio de sesión.....	55
Tabla 9: Caso de uso: Gestión de control de dispositivos .....	56
Tabla 10: Caso de uso: Control de dispositivos en modo automático .....	56
Tabla 11: Caso de uso: Ver reporte de dispositivos .....	57
Tabla 12: Caso de uso: Cerrar sesión .....	57

## ÍNDICE DE FIGURAS

Figura 1: Sector de la Domótica.....	21
Figura 2: Elementos de la Domótica.....	22
Figura 3: Placa Arduino Uno .....	24
Figura 4: Módulo Wifi Tinsine .....	25
Figura 5: Tarjeta Relay .....	26
Figura 6: SmarhPhone.....	26
Figura 7: Tablet Pc o tabletas electrónicas .....	27
Figura 8: Sistema de capas de Android .....	28
Figura 9: Entorno de programación de Android Studio .....	32
Figura 10: Diagrama en bloques del sistema electrónico .....	45
Figura 11: Módulo Arduino Uno .....	46
Figura 12: Módulo Wifi TinySine .....	46
Figura 13: Tarjeta de relay.....	47
Figura 14: Fuente de poder de alimentación .....	47
Figura 15: Esquema del circuito electrónico integrado.....	48
Figura 16: Diagrama en bloques para le gestión de la información.....	49
Figura 17: Diagrama del modelo relacional de la base de datos.....	50
Figura 18: Pantalla de conexión de la aplicación de Visual Basic.Net .....	52
Figura 19: Pantalla de inicio de sesión y reporte.....	54
Figura 20: Pantalla de control de dispositivos en modo automático.....	55

## INTRODUCCIÓN.

Antiguamente se utilizaban sistemas con arquitectura centralizada muy pocos flexibles ya que todos los sensores se conectaban a un solo autómatas o controlador y no existía el control de manera remota.

La domótica es la gestión técnica del uso simultáneo de la electricidad, electrónica e informática y así de esta manera permitir realizar funciones como regulación de temperatura, control de consumos de cada electrodoméstico, programaciones diarias, riego automático del jardín, control de escapes de gases, custodia y vigilancia frente a personas extrañas y el monitoreo como también control de manera local o remota.

Por todo lo mencionado la domótica usa un sinnúmero de dispositivos como sensores y actuadores, los mismos que pueden ser distribuidos en todo el hogar dependiendo las necesidades del usuario final o propietario.

El desarrollo de este trabajo de investigación se basa en la construcción de un sistema electrónico, que permita mejorar la calidad de vida en el hogar mediante el control de dispositivos remotamente desde un teléfono con plataforma Android.

En los tres primeros capítulos se describieron las problemáticas, objetivos generales y justificación, se hizo un breve estudio de la domótica y las tecnologías a nivel electrónico y de programación. Además se analizaron las herramientas de la investigación.

En el capítulo cuarto se analizó las diferentes tecnologías de hardware en lo que respecta a sistemas modulares Arduino para el control de los distintos dispositivos en el hogar, se diseñó un circuito electrónico domótico, se desarrolló la programación de las diferentes aplicaciones informáticas para un control remoto.

Al final de este trabajo de investigación se realizaron las conclusiones y sugerencias respectivas.

# **CAPITULO I**

## **MARCO CONTEXTUAL DE LA INVESTIGACIÓN**



## **1.1. Formulación de la Problemática.**

Imaginémonos que por error dejamos encendido nuestro aire acondicionado o luces en nuestro domicilio los mismos que tienen consumo significativo y que por consiguiente va a afectar a la economía familiar, eh allí plantear dicho sistema domótico para que el usuario verifique si algún artefacto se encuentra encendido y si fuese el caso, remotamente desde un teléfono celular o acceso web proceder a apagarlo como también si estuvimos en un día sumamente caluroso podemos llegar a nuestro hogar después de una jornada de trabajo agobiante para encontrarnos con un ambiente fresco y relajante.

## **1.2 Situación actual del problema**

Actualmente se están difundiendo tecnologías de gran alcance a nivel de hardware como lo es Arduino y también a nivel de software en telefonía móvil como lo es Android, donde integrando estas tecnologías y con el acceso a internet, podemos hacer uso extendido en un sistema completo e integral donde vamos a simplificar tareas diarias al usuario y controlar de manera independiente en cualquier momento o lugar a diferentes dispositivos en el hogar.

## **1.3 Problema de investigación**

### **1.3.1 Problema general**

La falta de recursos económicos para el acceso a sistemas tecnológicos domoticos en los hogares ha generado una dificultad enorme en las condiciones de vida y peor aun cuando existe un despilfarro de energía innecesario, por tal motivo se sugiere analizar lo siguiente:

¿Cómo diseñar y elaborar un sistema domótico de bajo costo para el monitoreo y control de dispositivos mediante un teléfono celular con plataforma Android usando tecnología Arduino en el hogar?

### **1.3.2 Problemas derivados**

**P1:** ¿Cuál de los módulos dentro la tecnología Arduino existentes en el mercado nos garantiza ser el más adecuado al momento de realizar el diseño para un sistema domótico?

**P2:** ¿Cómo diseñar una integración de circuitos que controle el encendido o apagado de dispositivos?

**P3:** ¿Cómo administrar los datos mediante la aplicación de un sistema informático de manera remota?

## **1.4 Objetivos**

### **1.4.1 Objetivo General**

- Implementar un sistema integral de monitoreo y control de dispositivos mediante un teléfono celular con plataforma Android usando tecnología Arduino en el hogar.

### **1.4.2 Objetivos Específicos.**

- ✓ Identificar el módulo a utilizar dentro de la tecnología Arduino que sea el más adecuado para el diseño del sistema domótico.
- ✓ Elaborar un sistema electrónico integral que se ejecute acciones de encendido y apagado de dispositivos eléctricos o electrónicos.
- ✓ Desarrollar una base de datos, una aplicación web y móvil que gestione la información para el control de dispositivos de manera remota.

## **1.5 Justificación**

En la actualidad desarrollar un sistema domotico integral, nos permite ahorrar tiempo, evitando desplazamientos hacia el hogar, brindando confort al usuario final, como también contribuye con el ahorro energético al controlar el apagado remoto de dispositivos.

Muchos de los hogares no disponen de un sistema domótico integral de control de dispositivos y por lo cual no contribuyen a la economía familiar por el desperdicio de la energía eléctrica.

Dadas estas razones nació la propuesta del control remoto de dispositivos en el hogar mediante un dispositivo móvil o computador de escritorio.

El desarrollo e implementación del sistema domotico integral, solventó las necesidades del control total en el hogar y de esta manera tener más confianza y tranquilidad en la seguridad del apagado de los dispositivos en el hogar.

Como conclusión podemos decir que los conocimientos adquiridos fomentan al desarrollo de nuevas aplicaciones que puedan ser realizadas en niveles académicos o profesionales dedicados al ámbito investigativo.

# **CAPÍTULO II**

## **MARCO TEÓRICO DE LA INVESTIGACIÓN**

## 2.1 Fundamentación conceptual

### 2.1.1 Domótica

Domótica es una comunidad del campo tecnológico el cual integra el control y supervisión de dispositivos en el hogar e instituciones usando de manera simultánea el uso de la electricidad, electrónica e informática y así de esta manera permitir realizar funciones como regulación de temperatura, manejo del consumo de los electrodomésticos, programaciones diarias de cada dispositivo, riego del jardín, control de gases y vigilancia.



**Figura 1:** Sector de la Domótica

**Fuente:** <http://rayuelaradio.com/domus-automatica-domotica/>

#### 2.1.1.1 Sistemas inteligentes centralizados y descentralizados.

En domótica existen dos tipos de sistemas los centralizados y los descentralizados.

##### Sistemas Centralizados

Los sistemas centralizados poseen una unidad central de proceso o CPU la cual se encarga de la administración de la información que se envía a los distintos dispositivos del sistema como por ejemplo los sensores, actuadores, detectores entre otros. (obrero, 2013)

Esta unidad central de proceso gestiona la información de los datos del entorno y actúa sobre un determinado circuito electrónico integral que cumple funciones específicas la cuales pueden generar beneficios como el ahorro de energía eléctrica, seguridad, confort, etc.

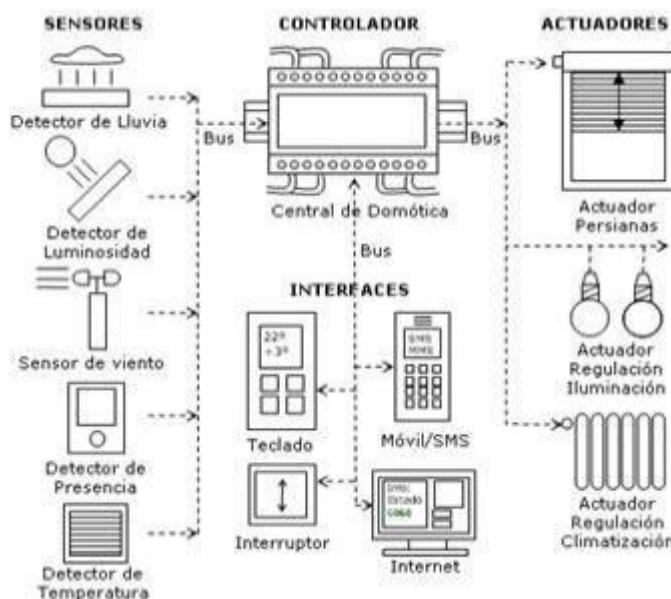
### Sistemas Descentralizados

Estos sistemas no tienen la necesidad de poseer un punto o nodo central inteligente para administrar algunas tareas específicas. El ordenador genera una programación interna en cada una de las unidades y por defecto se automatizan. (obrero, 2013)

Si se requiere un monitoreo remoto se debe realizar una interfaz cliente-máquina la cual debe poseer instrucciones de alta complejidad.

#### **2.1.1.2 Descripción de los sistemas domóticos**

Un sistema domótico está conformado por diferentes dispositivos, el principal de ellos es el procesador de información o unidad central de procesamiento el mismo que puede ser un microcontrolador o microprocesador. Estos son elementos electrónicos son configurables mediante una interfaz por medio de un computador. (Andrade, s.f.)



**Figura 2:** Elementos de la domótica

**Fuente:** <http://www.monografias.com/trabajos93/domotica-vivienda/domotica-vivienda.shtml>

Posee también sensores los cuales envían señales digitales al procesador de acuerdo a un evento generado y estos proceden a generar la activación de un actuador los cuales pueden ser una luminaria, ventilador etc. y de esta manera permitir la gestión tradicional de encendido y apagado de los elementos del hogar como las luminarias o electrodomésticos. (Reyes, s.f.)

Estos sistemas nos puedan dar muchos beneficios como el ahorro de energía eléctrica, incremento de confort y bienestar en el domicilio, como por ejemplo las luces pueden ser controladas en función a cuanta cantidad de luz necesitamos en el ambiente, evitando gastos energéticos innecesarios.

## **2.2 Fundamentación teórica**

### **2.2.1 Características básicas de arduino**

Los elementos principales que posee una placa arduino es el hardware que se refiere a la integración de los componentes físicos y el software que es el desarrollo del programa para realizar una tarea específica. (Lledó, 2012)

#### **2.2.1.1 Hardware**

El hardware que utiliza Arduino es una plataforma libre, su diseño y la distribución puede usarse libremente para la elaboración de un proyecto sin la necesidad de obtener una licencia. (SomosLibres, s.f.)

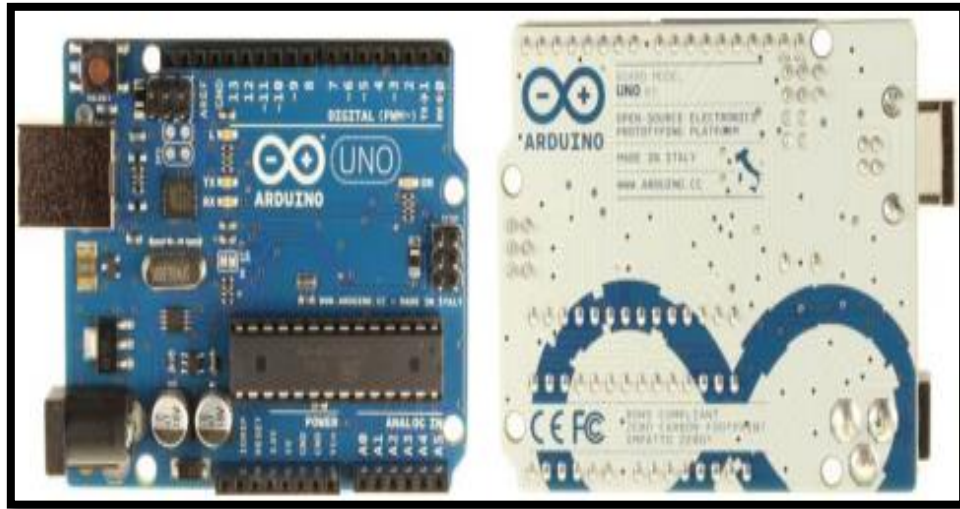
Al ser una plataforma de electrónica abierta existen desarrolladores aficionados que elaboran prototipos no oficiales pero respetando la similitud de sus características propias de la comunidad tecnológica Arduino.

El módulo Arduino viene desarrollado para que se conecten dispositivos como sensores, actuadores entre otros elementos para realizar un sistema integral mediante su interacción.

En el proyecto se han utilizado arduino uno, módulo Wifi Tinsine y tarjetas de relé que describiremos a continuación:

### 2.2.1.1.1 Arduino Uno

Esta placa posee una dimensión de 75x53 milímetros. El microcontrolador Atmega328 es la unidad central de procesamiento.



**Figura 3:** Placa Arduino Uno

**Fuente:** <https://riunet.upv.es/bitstream/handle/10251/18228/Memoria.pdf?sequence=1>

La alimentación se la puede realizar mediante un cable USB o externamente por otra fuente, como también posee pines digitales y analógicos distribuidos en su electrónica. (crm-ulatina, s.f.)

Sus elementos más importantes se detallan a continuación:

- Referencia para pines analógicos (AREF)
- Pines de tierra (GND)
- Pines digitales de entrada y salida
- Pines analógicos
- Conector USB
- Botón Reset
- ICSP (In Circuit Serial Programming)
- Microcontrolador ATmega328
- Fuente de alimentación externa
- Pin de Reset



- Pin de 3.3v
- Pin de 5v (crm-ultima, s.f.)

## 2.2. Módulo Wifi Tinsine

Este proporciona un puente de comunicación TTL hasta la comunicación inalámbrica IEEE802.11b. Se divide en dos partes escudo Tinsine Abeja y módulos WifiBee, por tal razón estos dispositivos pueden conectarse mediante un módulo WIFI, que se controla a través de una red inalámbrica. (TinySine, 2013)



**Figura 4:** Modulo Wifi Timysine

**Fuente:** [http://www.ebay.com/itm/WiFi-Shield-for-Arduino-/321277179515?pt=LH\\_DefaultDomain\\_15&hash=item4acd9cb67b](http://www.ebay.com/itm/WiFi-Shield-for-Arduino-/321277179515?pt=LH_DefaultDomain_15&hash=item4acd9cb67b)

Se puede integrar a la tecnología Arduino, se conecta a un router y se une a la red local después de ser configurado en modo AP. (TinySine, 2013)

Está formado por una tarjeta de relay y los dispositivos que se pueden acoplar son sensores, actuadores, comunicadores, dispositivos móviles como el Smartphone. Como todo sistema requieren una interfaz para conectarse entre sus módulos.



**Figura 5:** Tarjeta relay

**Fuente:** <http://www.ebay.com/itm/New-12V-4-Channel-Relay-Module-With-Optocoupler-For-Arduino>

### 2.2.2.3 Aplicaciones para dispositivos móviles

Las aplicaciones que poseen los dispositivos móviles pueden configurarse de distinta manera. El desarrollo de los programas informáticos puede ser elaborado por el fabricante del dispositivo o por el usuario final. (FotoNostra, s.f.)



**Figura 6:** Smartphone.

**Fuente:** [https://www.google.com.ec/search?q=smartphone&biw=1280&bih=699&source=lnms&tbn=isch&sa=X&ei=tpyRVMXcLsmcNuKKgcgB&ved=0CAYQ\\_AUoAQ#imgdii=](https://www.google.com.ec/search?q=smartphone&biw=1280&bih=699&source=lnms&tbn=isch&sa=X&ei=tpyRVMXcLsmcNuKKgcgB&ved=0CAYQ_AUoAQ#imgdii=).

Existen una infinidad de aplicaciones gratuitas o de pago que se descargan desde la red internet directamente al dispositivo. Suelen ser de fácil instalación o configuración. Por citar algunas referencias de aplicaciones existen sistemas de conversaciones en línea, administrador de e-mails, los juegos, etc. (FotoNostra, s.f.)

### 2.2.3 Tablet de PC o tableta electrónica.

Una tablet es un tipo de mini computadora móvil, posee un gran tamaño con respecto a los dispositivos móviles, la cual posee una pantalla táctil que a su vez puede ser gestionada por un sensor electrónico, por lo consiguiente no existen la necesidad de usar dispositivos periféricos de escritura y movimiento, como también es una magnífica herramienta para la navegación por la red internet y disfrutar de los contenidos multimedia existentes (Todo Tabletas Electrónicas, s.f.)



**Figura 7:** Tablet de PC o Tableta Electrónica.

**Fuente:**<http://todotabletaselectronicas.blogspot.com/2012/08/que-es-una-tableta-electronica.html>.

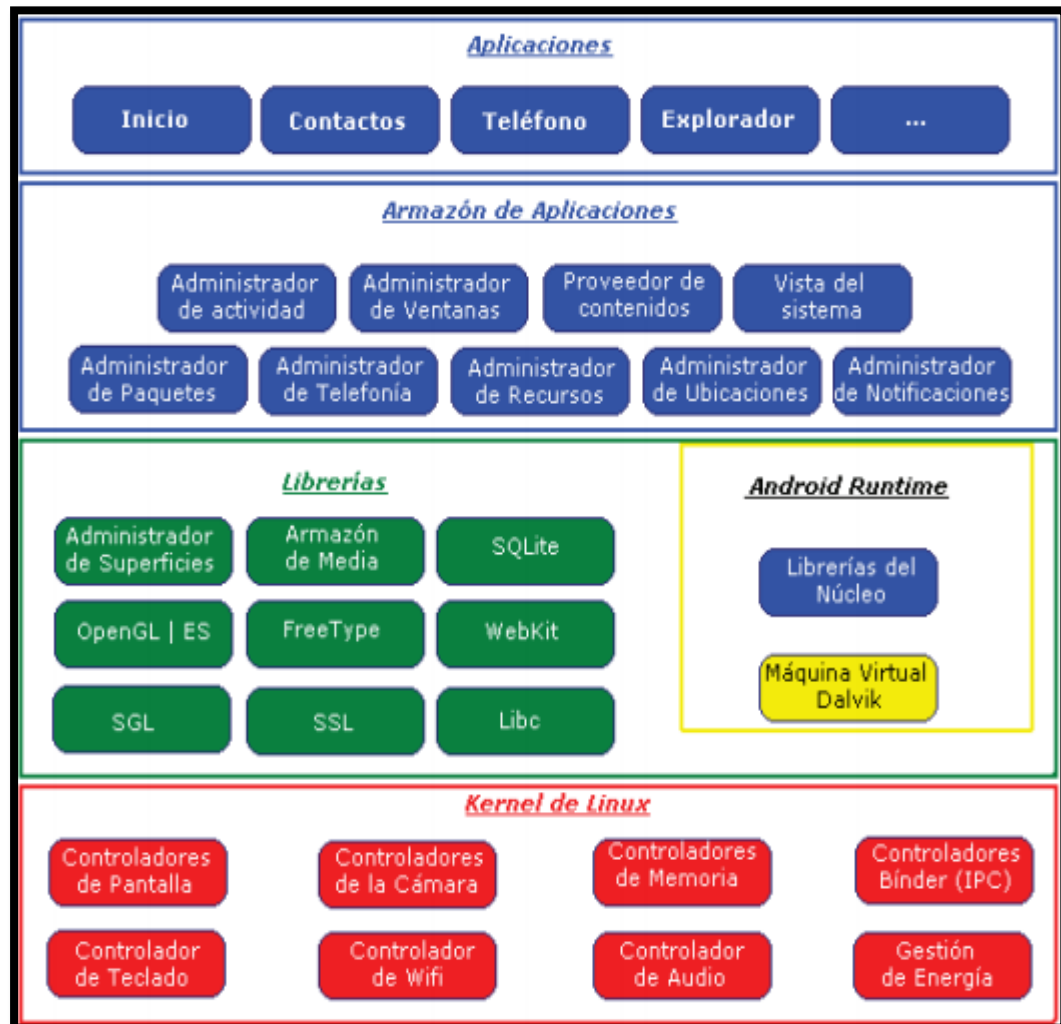
Actualmente, todos los fabricantes de equipos o dispositivos ha optado por incrementar el nivel de producción con referente a las Tablet, por tal motivo existen un centenar de modelos de Tablet Pc comercializándose con distintas características, marcas, capacidades, sistemas operativos, tamaños, costos, etc. (Todo Tabletas Electrónicas, s.f.)

### 2.2.4. Sistema operativo Android

#### 2.2.4.1. Historia

Inicialmente el sistema fue desarrollado por Android Inc., empresa que actualmente pertenece a Google desde el 2005, tres años después tuvo una acogida en el mercado de manera considerable ya que realizaron otra fusión con Open Handset Alliance, la cual es un gran conglomerado de empresas destinadas al desarrollo de tecnologías, quienes fueron impulsores en la

programación de código abierto, pero Google gracias a la incursión de Apache que es el mayor propulsor de software libre divulgo su código fuente. (Android OS, 2012)



**Figura 8:** Sistema de Capas de Android.

**Fuente:** <http://www.it-docs.net/ddata/18.pdf>

En la figura anterior, da una breve descripción de cada una de las fases:

- Como base principal tenemos al Kernel de Linux, que permite controlar a los diferentes periféricos o dispositivos
- Posteriormente tenemos a las librerías que son el núcleo para el desarrollo de software
- En la siguiente fase se gestiona la organización de las aplicaciones en base a los recursos requeridos

- Finalmente, están las aplicaciones a las que se puede acceder de manera autónoma. (Sanz, Saucedo, & Torralbo, s.f.)

Dado que Android está creado en el núcleo de Linux, se tiene acceso a todos sus recursos como sus controladores generales y dispositivos o periféricos. (Wikipedia, 2016)

#### **2.2.4.2. Introducción**

Ha existido en estos últimos años una gran evolución o cambio de manera radical en lo que respecta a la telefonía celular, ya que desde sus inicios tenían un gran tamaño y por consiguiente muy pesados, como también solo diseñados para mantener conversaciones desde cualquier lugar, por lo cual es bastante reducido su uso y de muy pocos beneficios, por todo aquello es que acorde avanza la tecnología en hardware también evoluciona en software mediante la creación de sistemas como Android para el control de dispositivos móviles. (Wikipedia, 2016)

#### **2.2.4.3. Definición de android.**

Es un sistema operativo basado en Linux para para teléfonos celulares, como también es utilizado muy poco en elementos de bajo recursos o consumo como lo es tabletas electrónicas, reproductores de audio. Este permite crear programas o aplicaciones en un framework o entorno de trabajo basado en Java lo que permite que cualquier persona con conocimientos en programación pueda comenzar a elaborar sus propias aplicaciones ya que Android es de código libre o software abierto lo que lo diferencia a otros sistemas operativos. (Wikipedia, 2016)

#### **2.2.4.4. Características de Android.**

A continuación, mencionaremos características que disfrutamos en tablets y smartphones, como es el fácil manejo e intuitivo como también:

- Se adapta con todo tipo mensajes como por ejemplo SMS, MMS, PUSH
- Permite la conectividad multipunto como WiFi, BlueTooth, LTE

- De fácil adaptación a pantallas de mayor resolución, como también VGA o gráficos 2D y 3D
- Soporte nativo para pantallas con varios puntos de contacto
- Soporte de multiproceso
- Permite el almacenamiento de información
- Tiene navegador web por defecto
- Permite reproducción de archivos multimedia más usados y streaming
- Tiene comando de búsqueda por voz mediante herramienta Google Now
- Permite conectividad diferentes dispositivos como gamepad, etc
- Opción de multiusuarios, en la versión Android 4.4, aparecida en el año 2013. (Arcia, s.f.)

#### **2.2.4.5. Versiones de Android**

Google ha intentado preparar su sistema operativo a los adelantos tecnológicos mediante las actualizaciones frecuentes, de esta manera compite con uno de sus más fuertes competidores, Apple con al iOs, quien es su rival más fuerte en el mercado. (ÁLVAREZ, 2014)

A continuación presentaremos las versiones más relevantes de Android y su fecha de lanzamiento:

- Versión 1.0 Apple Pie (23 de septiembre del 2008)
- Versión 1.1 Banana Bread (9 de febrero del 2009)
- Versión 1.5 Cupcake (27 de abril de 2009)
- Versión 2.0 y 2.1 Eclair (26 de octubre del 2009)
- Versión 2.2 Froyo (20 de mayo del 2010)
- Versión 2.3 GingerBread (6 de diciembre del 2010)
- Versión 3.0 y 3.2 HoneyComb (22 de febrero del 2011)
- Versión 4.0 Ice Cream Sandwich (18 de octubre del 2011)
- Versión 4.1 4.2 y 4.3 Jelly Bean (9 de julio del 2012)
- Versión 4.4 KitKat. (31 de octubre del 2013). (Francisco, 2013)

## **2.2.5. Lenguajes de programación**

### **2.2.5.1. Introducción**

Son herramientas elementales, muy básicas para elaborar sentencias de código o programas y desarrollar aplicaciones de manera integral, entre algunas conocidas tenemos a C, C++, Visual Basic Studio, entre otros. (Lenguajes de Programación, s.f.)

El computador interpreta señales de control de un programa desarrollado previamente e instalado en el disco duro del mismo. (moraeduardo2003, s.f.)

Tienen entornos de programación de fácil acceso al usuario y así facilitan la tarea de ejecutar los proyectos a realizarse muy independiente del modelo del computador de escritorio. (Lenguajes de Programación, s.f.)

Se llama código fuente o lenguaje de máquina a las instrucciones de programación y utilizan un compilador para su ejecución. Para la ejecución de un programa que posee un compilador solo es necesario traducir el código simbólico en lenguaje máquina. En cambio otros sistemas de desarrollo poseen un traductor interno que realiza toda la gestión (Lenguajes de Programación, s.f.)

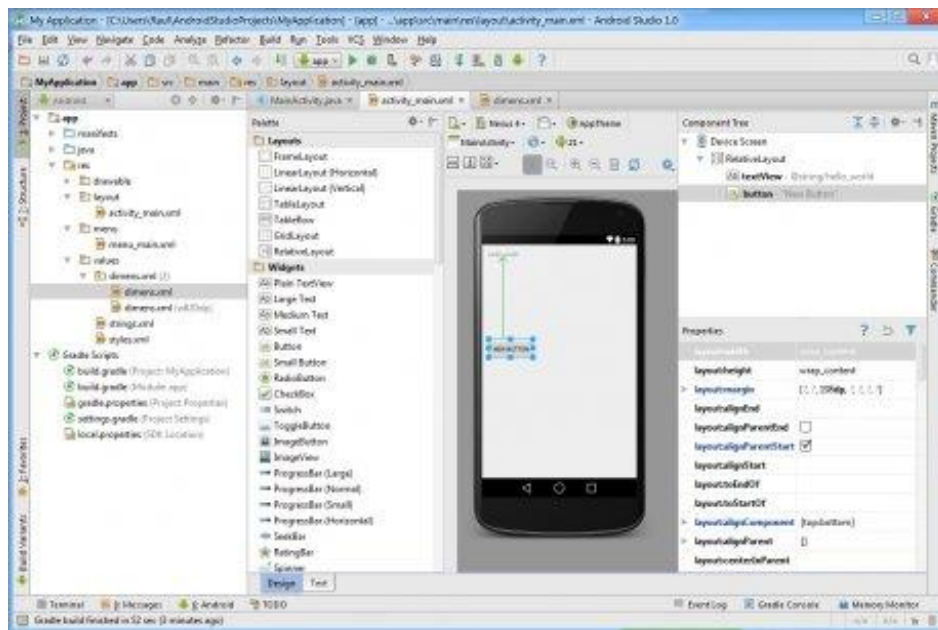
El proceso intérprete lo realiza de manera independiente debido a que no requiere ejecutar ninguna fase de programación, pero carece de rapidez al momento de ejecutarse y su análisis del código fuente de manera simultánea. (Lenguajes de Programación, s.f.)

### **2.2.5.2. Android Studio**

Posee un entorno de desarrollo integrado, el cual supera en algunos aspectos al plugin ADT para Eclipse. Android Studio usa licencia Apache de código libre, fue ejecutado bajo lenguaje Java y su principal característica es que es compatible con distintos sistemas operativos. (academia android, 2014)

Hace muy poco tiempo Google realizó su presentación en un prestigioso congreso de desarrolladores, para dar a conocer un entorno exclusivamente y generar

aplicaciones para la gama de dispositivos Android, de esta manera brindar un mayor control de calidad del proceso. (Actividad Integradora, 2015)



**Figura 9:** Entorno de programación Android Studio.

**Fuente:** <http://android-studio.uptodown.com/windows>

Para el 2014 Android Studio cambió su versión beta a su primera versión estable, y de esta manera fue recomendado como entorno de desarrollo de aplicaciones móviles y así dejando en estado obsoleto al plugin ADT de Eclipse. (academia android, 2014)

#### **2.2.5.2.1. Características principales de Android Studio.**

- Posee un sistema de emulación rápida
- Contiene una infraestructura alta para el desarrollo de pruebas
- Su compilador es bastante flexible ya que usa Gradle
- Sistema de entorno de desarrollo unificado para todos los dispositivos móviles
- Es compatible con NDK, C++
- Soporte integrado para herramientas de Google (App Engine, Google Cloud Messaging)



- Soporte para problemas de compatibilidad y rendimiento
- Herramienta para la gestión de cambios de la aplicación móvil mientras está en ejecución. (Android Studio, s.f.)

#### **2.2.5.2.2. Requerimientos para la instalación del sistema Android Studio.**

- Sistema Operativo Windows 8 y versiones inferiores
- Espacio requerido en el disco duro de 1GB para emulación de imágenes y Android SDK
- Requerimiento de memoria RAM superior a 2GB
- Necesita una mínima resolución de 1280x800 en la pantalla
- Instalar la versión el kit de herramientas de Java superior a 7. (Actividad Integradora, 2015)

#### **2.2.5.2.3. Ventajas del uso del Android Studio**

A continuación presentaremos algunas de sus ventajas:

- Tiene un sin número de plantillas para diseñar los proyectos e interfaces
- Contiene internamente un poderoso compilador Gradle
- Por ser intuitivo y fácil de usar es mucho más rápido y poderoso que Eclipse
- Posee actualizaciones de manera automática y periódica
- Permite la exportación de manera fácil y sencilla de archivo extensión apk
- Permite programar de manera estructurada y ordenada
- No permite dependencias
- Permite crear cualquier emulador sin consumir muchos recursos. (Ardións, 2016)

#### **2.2.5.3 PHP**

Es un codificación de programación web muy poderosa y conocida por los programadores ya permite crear páginas web dinámicas, este funciona incluyendo código php en código html que usa un navegador web el cual es ejecutado por un

servidor Apache y el usuario final recibe un resultado, de esta manera es una herramienta muy amigable a nivel profesional. (¿Qué significa php?, 2012)

#### **2.2.5.4 Visual Basic.NET**

Este lenguaje de programación orientado a objetos, es la unión de Visual Basic con Framework.Net, ya que se integra herramientas actuales y así desarrollar aplicaciones más avanzadas sin perder la similitud en el manejo de instrucciones. (David, 2010)

#### **2.2.5.5 SDK de Android**

SDK es un grupo de herramientas para desarrollo de software mediante el cual podemos desarrollar fácilmente aplicaciones o crear paquetes de instrucciones para un sistema en especial. Por lo general estas nos ayudan mediante ejemplos o alguna nota técnica para la elaboración de nuestro proyecto. (Roldayan, 2013)

Un ejemplo de un SDK tenemos al Android Developer Tools (ADT) el cual es una inserción como herramienta para el entorno de desarrollo integral Eclipse, de esta manera se consigue una mayor capacidad y rapidez para una mejor construcción de aplicaciones Android. (Roldayan, 2013)

### **2.2.6 Base de datos**

#### **2.2.6.1 Introducción**

Una base de datos es un repositorio el cual guarda grandes cantidades de paquetes de información de forma organizada para procesarla y usarla fácilmente por alguna empresa. (Pérez Váldez, 2007)

En el campo de la informática, la base de datos es un sistema integrado por conjunto de datos almacenados en memorias los cuales permiten acceder a ellos para permitir su manipulación, pues en esta existen tablas que guardan información necesaria para generar un registro cuando el usuario lo requiera. (Pérez Váldez, 2007)

### 2.2.6.2 PostgreSQL

Es un gestor de base de datos objetos-relacionales (ORDBMS) que ha sido desarrollado en distintas formas anteriormente. (Robelo, 2011)

#### 2.2.6.2.1 Características de PostgreSQL

- Soporta estándares: SQL92 y SQL99
- Transacciones: Se protege la integridad de datos
- Integridad referencial: Es utilizada para garantizar la validez de la información de la base de datos del gestor PostgreSQL
- Bloqueo de tablas y filas: El gestor PostgreSQL ofrece distintos modos de bloqueo para controlar el acceso de datos dentro de una tabla, algunos de estos modos se genera automáticamente dentro del programa
- Constraints y triggers: Su principal función es mantener la integridad y consistencia en la base de datos
- Múltiple tipos de datos predefinidos: El gestor PostgreSQL realiza la implementación de tipos de datos definidos para el estándar SQL3
- Soporte de tipos funciones de los usuarios: El gestor de base de datos PostgreSQL soporta operaciones, y métodos de acceso y datos definidos por el usuario
- Entre otras características podemos decir que posee conectividad TCP/IP, JDBC y ODBC. Además su interfaz soporta lenguajes como C, C++, Java, Delphi, PHP, Bash. (Robelo, 2011)

#### 2.2.6.2.2 Ventajas de PostgreSQL

- Instalación ilimitada: Debido a que un software de acceso libre no requiere licencia por tal motivo el usuario no puede ser demandado por violar acuerdos de licencia
- Soporte: Posee varios beneficios debido a que la compañía tiene una comunidad de profesionales que contribuyen a resolver las necesidades del usuario

- Ahorro en costos de operación: Este gestor de base de datos ha sido diseñado y creado para poseer un mantenimiento y ajuste reducido con relación a otros productos, conservando sus características, rendimiento y estabilidad
- Estabilidad y confiabilidad: En ninguna ocasión PostgreSQL ha presentado una caída en sus años de operación, dándole un alto grado de confiabilidad
- Extensible: Sus extensiones son de carácter gratuito debido a que si requerimos personalizarlo lo podemos realizar sin ningún inconveniente
- Multiplataforma: Está disponible en cualquier plataforma de Unix en su última versión estable y en la actualidad en versión nativa para Windows
- Diseñado para ambientes de alto volumen: Este gestor de base de datos utiliza una estrategia e almacenamiento de filas conocida MVCC para conseguir mejor respuestas en grandes ambientes
- Herramientas gráficas de diseño y administración de base de datos: Posee varias herramientas gráficas de alta calidad como la de administración de datos (pgAdmin, pgAccess) y de diseño de datos. (Robelo, 2011)

## **2.2.7 Servidor HTTP Apache**

### **2.2.7.1 Historia**

En febrero de 1995 nace como concepto Apache, el cual es un servidor de formato http, pero luego el proyecto fue truncado por un tiempo, más tarde mediante una coordinación de trabajo vía correo electrónico Brian Behlendorf y Cliff Skolnick lograron establecer un espacio libre para desarrolladores y de alguna manera obtener un apoyo organizativo, financiero y legal como servidor y es así como se formó fundación de software Apache en el año 1999. (Fumás, 2014)

### **2.2.7.2 Apache**

Es un servidor web HTTP de software libre, compatible con muchos sistemas operativos, múltiples herramientas, de alta operatividad, muy fiable y sin costo alguno. (Mariael, 2015)

### **2.2.7.3 Servidor Web**

Programa para la transporte de datos de hipertexto mediante el protocolo http. El servidor web está instalado en un computador de escritorio y esta debe tener acceso o conexión Internet y así de esta se puede realizar alguna petición en un navegador y poder acceder a una página web. (Ibrugor, 2015)

### **2.2.7.4 Ventajas**

- Instalación y configuración: software de código abierto
- Costo: es totalmente gratuito
- Funcionalidad y soporte.: alta popularidad y gran aceptación en la red y siempre se están generando mejoras
- Multiplataforma: puede abarcar muchos sistemas operativos y tiene compatibilidad con MacOS, Windows y Linux.
- Rendimiento: puede generar más de un millón de visitas diarias
- Seguridad: Soporte SSL y TLS. (Ibrugor, 2015)

### **2.2.7.5 Inconvenientes**

- Posee formatos de configuración que no han sido estandarizados.
- Falta de integración
- No contiene un buen panel de configuración. (Fumás, 2014)

## **CAPÍTULO III: METODOLOGÍA DE INVESTIGACIÓN**

### 3.1. Tipo de investigación

El presente trabajo de investigación para el desarrollo de un sistema de control domótico integral es del tipo diagnóstico.

El difícil acceso al control de manera remota a dispositivos en el hogar ha generado dificultades en las actividades diarias como también pérdidas económicas al usuario final y no nos brinda confort.

### 3.2. Instrumentos de investigación

#### 3.2.1. Método de observación

El método ejecutado para el desarrollo del proyecto es del tipo OBSERVACIÓN. Durante el proceso de la actividad se determinó el tiempo transcurrido y los recursos para el desarrollo, donde de esta manera nos permitió diseñar el circuito electrónico integral para dar soluciones al buen manejo de los dispositivos en el hogar.

### 3.3 Costos de materiales propuestos en el desarrollo de la investigación.

#### 3.3.1 Materiales para el desarrollo del sistema electrónico.

Descripción	Costo
Módulo Arduino Uno	35,00 usd
Módulo Wifi Tinsine	45,00 usd
Módulo Relays 8 salidas	30,00
Teléfono Celular Samsung táctil	150,00 usd
Router Dlink (Internet)	45,00 usd
Cables varios y conectores	15,00 usd
Computador básico CPU (servidor)	150,00 usd
Reguladores de voltaje 12 voltios	20,00 usd
<b>Total</b>	<b>490,00 usd</b>

**Tabla 1** Cuadro del presupuesto de materiales electrónicos

**Fuente:** Investigador

### 3.3.2. Presupuesto a nivel de programación.

Descripción	Costo
Arduino IDE	Free
Visual Basic.Net (aplicación en servidor)	Free
PostgreSql (base de datos en servidor)	Free
Plataforma Android Studio	Free
Php	Free
<b>Total</b>	<b>Free</b>

**Tabla 2** Cuadro del presupuesto de las herramientas informáticas usadas.

**Fuente:** Investigador



# **CAPÍTULO IV:**

## **ANÁLISIS E INTERPRETACIÓN DE LOS RESULTADOS**

## 4 ANÁLISIS E INTERPRETACIÓN DE LOS RESULTADOS

### 4.1 Análisis de la tecnología Arduino

Arduino es una sociedad en el área tecnológica que posee electrónica modular libre o de hardware abierto, en la cual se diseñan y elaboran placas compuestas por circuitos impresos que se integran a un chip o microcontrolador dentro de un ambiente de programación o desarrollo (IDE).

Como microcontrolador posee un ATMEL AVR el cual dispone de arquitectura RISC (computador con conjunto de instrucciones reducidas) por lo cual es un procesador muy potente y de bajo consumo de energía.




CARACTERÍSTICAS	ARDUINO UNO 	ARDUINO LEONARDO 	ARDUINO NANO 
MICROCONTROLADOR	ATMEGA 328 - 8 BITS	ATMEGA 324 - 8 BITS	ATMEGA 168 o 328 8 BITS
FRECUENCIA	16MHZ	16MHZ	16MHZ
MEMORIA RAM	2KB	2KB	8KB
MEMORIA EEPROM	1KB	1KB	1KB
MEMORIA FLASH	32KB	32KB	16KB
PINES DIGITALES ENTRADA/SALIDA	14/14	20/20	14/14
TENSIÓN /CORRIENTE PINES DIGITALES	5V 40MA	5V 40MA	5V 40MA
PINES ANALÓGICOS ENTRADA/SALIDA	6/0	12/0	8/0
PINES PWM	6	7	6
CONEXIÓN SERIAL/UART	1	1	1
CONEXIÓN USB	SI	SI	SI
COSTO	BAJO	MEDIO	BAJO

Tabla 3 Cuadro comparativo de la tecnología modular Arduino.

Fuente: Investigador

Para el prototipo domotico se usara como interfaz electrónica el Arduino Uno por su amplia memoria flash como también en el bajo requerimiento de los pines o puertos para las salidas digitales hacia los dispositivos a controlar.

También esta tarjeta electrónica es necesaria conectarla mediante un módulo de comunicación a un servidor y de esta manera almacenar los datos por ello se analizara a continuación la tecnología más acorde.

#### 4.1.1 Análisis para la conectividad en la transmisión de datos

TIPO DE RED	WIFI 802.11b	BLUETOOTH	ZIGBEE
MOVILIDAD	ALTA	CORTA	MEDIA
COSTO ASOCIADO	MEDIO – ALTO	BAJA	MEDIA-BAJA
AREA COBERTURA	NACIONAL	LOCAL	LOCAL
INTERFERENCIAS ELECTRONICA	ALTO	ALTO	MEDIA
SEGURIDAD	BAJO	BAJO	MEDIO
APLICACIÓN	INTERNET	SUSTITUIR CABLES	RADIO DIFUSION DIGITAL DE BAJO CONSUMO
VELOCIDAD	11-54Mbps	700Kbps	250Kbps
TIPO DE RED	LAN	PAN	WPAN
ALCANCE	4 a 100 metros	Menor a 30 metros	10 a 300 metros
TAMAÑO DE RED	32 nodos	7	255-65000
TOPOLOGIA	ARBOL	P2P	MESH ESTRELLA ARBOL

**Tabla 4** Cuadro comparativo de los diferentes tipos de comunicaciones inalámbricas

**Fuente:** Investigador

Todas las redes de comunicaciones tienen sus ventajas y desventajas pero para nuestro requerimiento utilizaremos el Wifi debido a que estaremos conectados al internet.

CARACTERÍSTICAS	Wifi Serial ESP8266 	Wifi Tinsine Shield 	Arduino Yun Shield 
PROTOCOLO	802.11b/g/n	802.11b/g/n	LAN - 802.11b/g/n
ESTADOS	P2P, Access Point	P2P, Access Point	P2P, Access Point
POTENCIA DE SALIDA	19.5dBm	0 hasta 12dBm	6 dBm
CONSUMO DE ENERGIA	<10uA	<4ua	BAJO
ANTENA EXTERNA	NO	SI 3dbi	NO
BOTON DE RESET	NO	NO	SI
CONECTOR USB	NO	NO	SI
COSTO	BAJO	BAJO	ALTO

**Tabla 5** Cuadro comparativo de los módulos Wifi.

**Fuente:** Investigador

Para la comunicación con el servidor y el transporte de datos desde el Arduino Uno, se eligió el módulo Wifi Tinsine Shield por tener todos los protocolos de conectividad, un bajo consumo de energía y una antena externa de 3dbi la misma que nos da una seguridad en la conexión hacia el servidor de datos y el Router a usarse dentro de nuestra red Wifi.

## 4.2.- DISEÑO DEL PROTOTIPO ELECTRONICO DEL SISTEMA DE CONTROL DOMOTICO.

### 4.2.1.- Descripción general del sistema.

Para la propuesta a desarrollar del sistema integral domotico se utilizaran los siguientes elementos de hardware y de esta manera tener un control de todos los dispositivos en el hogar.

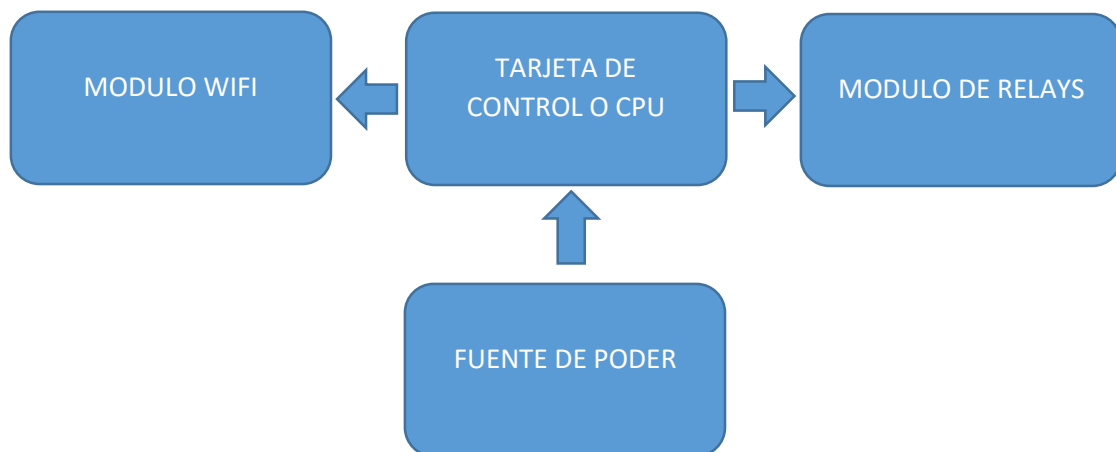
#### Hardware:

- Tarjeta Arduino Uno.
- Módulo Tinsine Shield

- Módulo de Relays para control

#### 4.2.2 Esquema y diseño del sistema electrónico de control integral.

Con la ayuda de un análisis de los diferentes elementos electrónicos tenemos como conclusión el siguiente esquema empleado para la implementación del sistema integral domotico:



**Figura 10** Diagrama en bloques del sistema electrónico

**Fuente:** Investigador

##### 4.2.2.1 Tarjeta de control o CPU

La tarjeta que controla todo el sistema es un Arduino Uno el mismo que dispone de memoria flash con gran capacidad de almacenaje de información y un lenguaje de programación de fácil comprensión como también gran cantidad de puertos para encender o apagar nuestros diferentes dispositivos a ser controlados.

Usaremos los puertos b0 (Tx) y b1 (Rx) para comunicación con el módulo Wifi Tinsine y del b2 hasta el b8 para el accionamiento de los relays del módulo de control. Tiene un puerto USB para su programación mediante una interfaz propia de Arduino.



**Figura 11** Módulo Arduino Uno

**Fuente:** Investigador

#### 4.2.2.2 Módulo Wifi

El modulo que usaremos es el Tinsine y se usara para la comunicación al servidor y envió de una trama de datos a la base de datos de todos los dispositivos a ser controlados.

Tiene una antena de 3dbi la cual nos garantiza la conectividad con gran eficiencia hacia nuestro servidor. Los datos serán leídos por una aplicación realizada en Visual Basic.net desde la base de datos cada 200ms y enviados hacia el arduino por el puerto de comunicación 2000 propio del módulo.

Previamente en la programación del arduino se guardara los datos de la red wifi a conectarnos como es nombre de la red (SSID) y contraseña de la misma para de esta manera el Wifi Tinsine automáticamente se conectara al router inalámbrico y también a la aplicación de envío de trama de paquetes



**Figura 12** Modulo Wifi Tinsine

**Fuente:** Investigador

#### 4.2.2.3 Módulo de Relays

Este módulo realizara el accionamiento o apagado de los diferentes dispositivos conectados en cada una de sus salidas de control.

Cada relay está controlado por un optoacoplador de esta manera garantizamos que no exista interferencias en nuestro sistema de control con la parte de potencia o actuadores.



**Figura 13** Tarjeta de Relays

**Fuente:** Investigador

#### 4.2.2.4 Fuente de poder

Con este elemento suministramos energía regulada de 5 y 12 voltios para alimentar al módulo Arduino como también al módulo de relays de control.

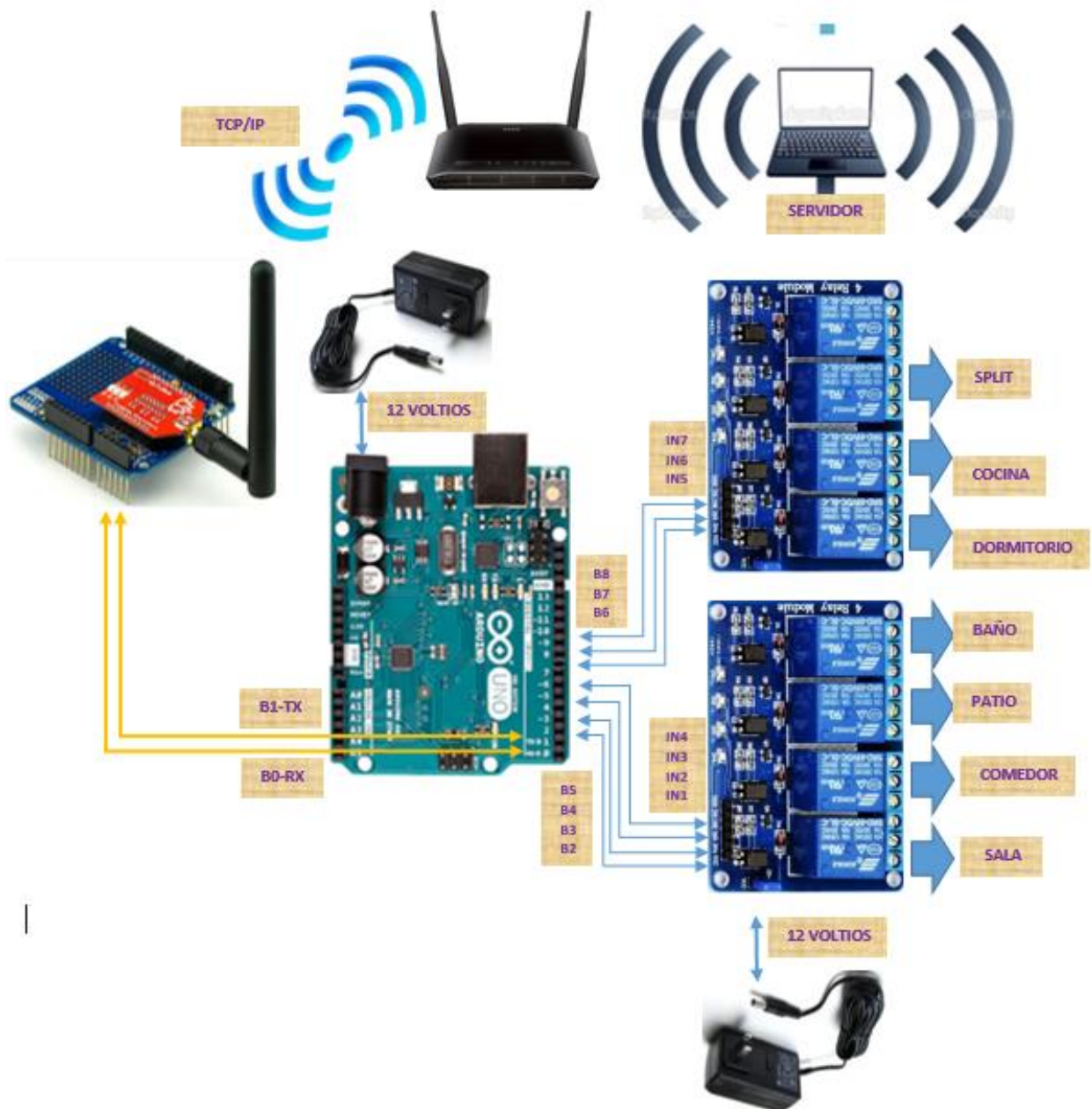


**Figura 14** Fuente de poder para alimentación

**Fuente:** Investigador

### 4.2.3 Acoplamientos de componentes electrónicos

A continuación se detalla en un diagrama esquemático de los elementos que conforman el circuito electrónico integrado del sistema domotico el cual recibirá las peticiones del usuario para encender o apagar dispositivos de manera local o remotamente.



**Figura 15** Esquema del circuito electrónico integrado

**Fuente:** Investigador



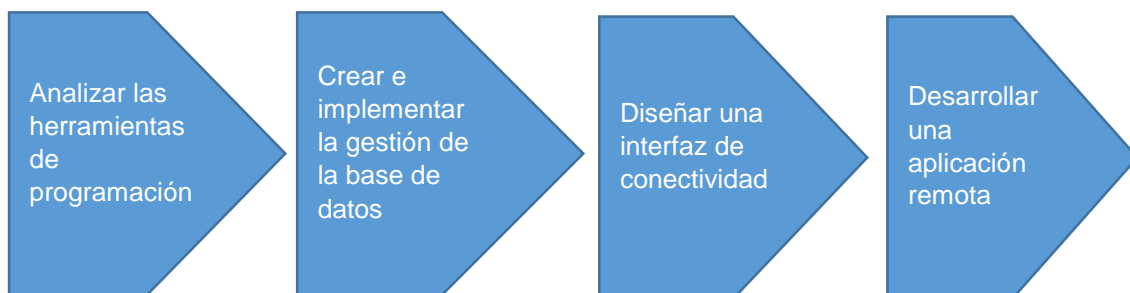
En el presente circuito electrónico integrado se realiza el control de cada uno de los relays que comandan a los dispositivos conectados en la sala, comedor, patio, baño, dormitorio, cocina, Split. Arduino uno comanda las señales desde b2 hasta b8 las cuales están conectadas a las entradas del módulo relay desde In1 hasta In7, es decir, b2 con In1, b3 con In2, b4 con In3, b5 con In4, b6 con In5, b7 con In6, b8 con In7.

El Arduino Uno se comunica vía serial al módulo Wifi Tinsy por los pines b0 y b1 para la transmisión y recepción de datos y de esta manera tenemos comunicación de manera inalámbrica con el servidor de datos. En la programación del arduino se guardan parámetros de configuración de la red wifi a conectarse dentro del hogar como lo es el nombre, contraseña, etc. para que de esta forma una vez haya una autenticación entre los 2 dispositivos módulo wifi Tinsy y el Router del hogar exista el enlace establecido.

El módulo Arduino Uno se alimenta con una batería de 12 voltios de voltaje constante y como en la tarjeta de los relays tenemos opto acopladores para aislar la parte de control o señales digitales con la de potencia por esta razón se debe alimentar con otra fuente externa de 12 voltios regulados de total independencia.

#### **4.3.- DESARROLLO DE LAS HERRAMIENTAS INFORMATICAS PARA LA GESTION DE LA INFORMACION.**

Primeramente se realizó un análisis de herramientas de desarrollo para de esta manera elegir la más adecuada para el control a cada uno de los dispositivos de manera remota mediante un computador de escritorio o móvil.



**Figura 16** Diagrama en bloques para la gestión de la información

**Fuente:** Investigador

#### 4.3.1.- Análisis para el desarrollo de las aplicaciones informáticas y la gestión de la base de datos.

En esta sección se analizara previamente las herramientas informáticas a utilizar como también los gestores de base de datos para el almacenaje de la información.

<b>Características</b>	<b>Java</b>	<b>Visual Basic Net</b>	<b>Php</b>	<b>Android Studio</b>	<b>Eclipse</b>
<b>Consistencia</b>	Poca	Superior	Poca	Superior	Media
<b>Procesamiento de memoria</b>	Superior	Poca	Poca	Poca	Bajo
<b>Amplitud</b>	Bajo C++	-----	-----	-----	-----
<b>Capacidad</b>	Poca	Superior	Poca	Superior	Medio
<b>Plataforma</b>	Todas	Todas	Todas	Android	Android
<b>Entorno visual</b>	Medio	Medio	Media	Superior	Superior
<b>Desarrollo de sitios web</b>	Si	No	Si	No	No
<b>Diseño de aplicaciones móviles</b>	Si	No	No	SI	SI

**Tabla 6** Cuadro comparativo de las herramientas para desarrollo de software

**Fuente:** Investigador

Entre las herramientas de desarrollo de software se eligió Android studio para elaborar la aplicación móvil por su bajo consumo de memoria y fue creado para hacer app para dispositivos bajo plataforma Android, para crear la página web se usará PHP y Visual Basic.Net para la aplicación de comunicación con el sistema electrónico por su alto rendimiento.

Para la proyecto a desarrollarse se utilizaran los siguientes elementos de software para la gestión de la información enviada desde el sistema electrónico hacia el servidor de datos.

<b>Características</b>	<b>Oracle</b>	<b>MySql</b>	<b>Sql Server</b>	<b>PostgreSql</b>
<b>Licencia</b>	Privada	Libre	Privada	Libre
<b>Conectividad</b>	Lenta	Segura	Segura	Estable
<b>Configuración</b>	Difícil	Fácil	Fácil	Fácil
<b>Consumo</b>	Medio	Bajo	Alto	Bajo

<b>Soporte</b>	Multiplataforma	No posee	Multiplataforma	Multiplataforma
<b>Costo</b>	Alto	Medio	Medio	Gratuito

**Tabla 7** Cuadro comparativo de las herramientas de administración de base de datos

**Fuente:** Investigador

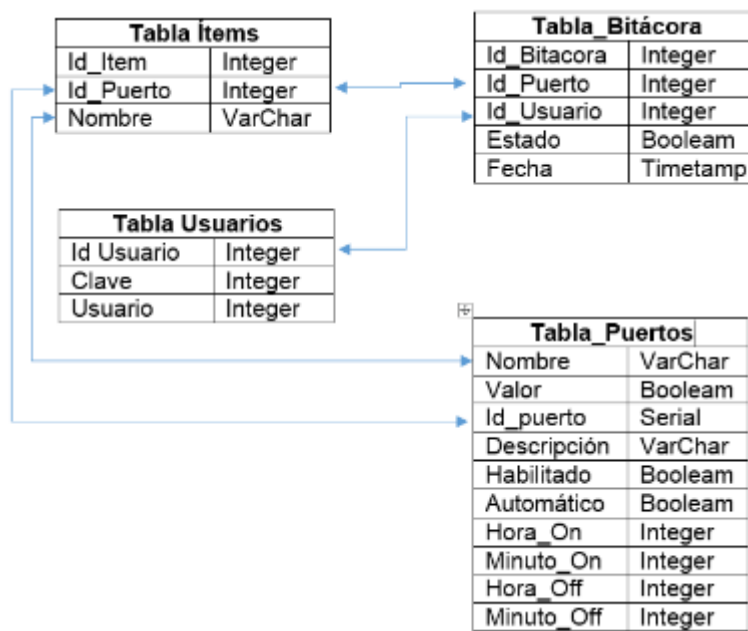
Se eligió PostgreSQL al ser un gestor de base de datos libre, estable, de fácil configuración, de bajo consumo y gratuito.

#### Software:

- Android Studio
- Php
- PostgreSQL
- Visual Basic.Net
- Apache

#### 4.3.2.- Diseño del análisis relacional de la base de datos

Como previamente se eligió ya utilizar PostgreSQL para la gestión de la información ahora presentaremos ya el diseño de la base de datos como tal mediante el análisis relacional de cada uno de sus campos y tablas.



**Figura 17** Diagrama del modelo relacional de la base de datos

**Fuente:** Investigador

En la tabla puertos manejamos las asignaciones de cada uno de los dispositivos a ser controlados como es el encendido automático, almacenar la hora y minuto, también el habilitado de cada uno de las variables a controlar.

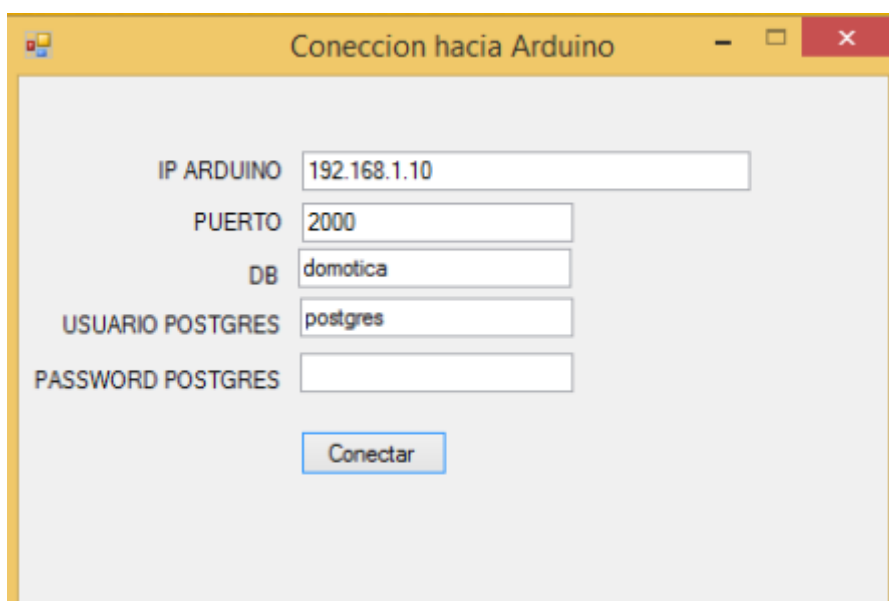
En la tabla ítems le asignamos un nombre o descripción a cada uno de los puertos o dispositivos.

En la tabla bitácora nos permite guardar información de cada uno de los puertos con su respectivo registro de hora, minuto y fecha cuando ocurrió el evento del encendido o apagado de cada dispositivo.

En la tabla usuario permite el acceso al sistema informático mediante la validación con su respectiva clave de administrador.

#### **4.3.3.- Desarrollo de la aplicación de gestión de la información con el servidor y el esquema electrónico.**

Para tener comunicación desde el esquema electrónico Arduino y el servidor de datos desarrollamos una aplicación en Visual Basic.Net que nos permitirá conectarnos a nuestra base de datos y transportar trama o paquetes hasta el sistema de control.

The image shows a screenshot of a Windows application window titled "Coneccion hacia Arduino". The window has a yellow title bar with standard Windows window controls (minimize, maximize, close). The main area is light gray and contains five text input fields arranged vertically. The labels for the fields are "IP ARDUINO", "PUERTO", "DB", "USUARIO POSTGRES", and "PASSWORD POSTGRES". The values entered in the fields are "192.168.1.10", "2000", "domotica", "postgres", and an empty field respectively. Below the fields is a blue button labeled "Conectar".

**Figura 18** Pantalla de conexión de la aplicación en Visual Basic.Net  
**Fuente:** Investigador

Con esta aplicación obtenemos la conectividad desde la base de datos al sistema electrónico utilizando el módulo Wifi Tinsine y consiste en mediante un puerto TCP/IP transportar una cadena de paquetes vía inalámbrica por el puerto de comunicación 2000 cada 200m seg, por eso se debe colocar el nombre, usuario y password para su conexión.

La dirección IP, máscara, puerta de enlace, nombre de la red wifi, contraseña se guardan previamente en el Arduino para que de esta manera exista la conectividad entre aplicación y módulo Wifi Tinsine.

#### **4.3.4.- Desarrollo de la aplicación web para la gestión de la información con el servidor y el esquema electrónico.**

Para el desarrollo de la aplicación web se usó código Php mediante la herramienta de diseño de páginas Dreamweaver por ser una herramienta poderosa.

El programa que permite el acceso web contiene 10 módulos los cuales se describirán su funcionalidad a continuación:

**LOGOUT.-** Módulo para salir y regresar al módulo de inicio.

**INDEX.-** Módulo para ingresar el usuario y contraseña

**MÓDULO.-** Módulo de conexión a la base de datos

**MAIN.-** Módulo o pantalla principal

**ACTUALIZAR CONFIGURACION.-** Módulo que actualiza en la base de datos los cambios al seleccionar la casilla automática.

**ACTUALIZAR ESTADO DE PUERTOS.-** Módulo lee de la base de datos y actualiza los estados de los botones de control en la pantalla

**ESTADO DE PUERTOS.-** Módulo lee de base de datos y dibuja la pantalla de control.

**CONFIGURAR PUERTO.-** Módulo dibuja en la pantalla para configurar los botones de control

**ESTADO DE LUCES.-** Módulo que realiza el cambio de estado (encendido o apagado) cada luminaria en el plano de casa dibujándolo

**REPORTE.-** Módulo para generar los reportes de los eventos ejecutados durante un rango determinado.

#### 4.3.5.- Desarrollo de la aplicación móvil la para gestión de la información con el servidor y el esquema electrónico.

Para el desarrollo de la aplicación móvil se usó Plataforma Android Studio por ser una herramienta poderosa.

La aplicación móvil contiene 3 pantallas las cuales permiten realizar instrucciones como iniciar sesión, cerrar sesión, ver reportes, configuración de los dispositivos en modo automático, activación o desactivación, las cuales presentaremos de manera breve a continuación y para cada una está elaborada su respectivo caso de uso:



**Figura 19** Pantalla de inicio de sesión y reporte de control de dispositivos con la aplicación móvil.  
Fuente: Investigador



**Figura 20** Pantalla de inicio de sesión y reporte de control de dispositivos de modo automatico.

**Fuente:** Investigador

Los casos de uso explican el funcionamiento de la pantalla de la aplicación.

<b>CASO DE USO 01</b>	Iniciar Sesión.	
<b>PRE-CONDICIÓN</b>	Tener acceso al sistema	
<b>DESCRIPCIÓN</b>	Se solicita al ingresar el Nombre del Usuario y la Clave de acceso.	
<b>SECUENCIA</b>	<b>PASO</b>	<b>ACCIONES</b>
	<b>1</b>	Empieza cuando el usuario inicia la aplicación Domótica Mobile ejecutándola en su teléfono móvil.
	<b>2</b>	El sistema muestra el menú "Control de Dispositivos".
	<b>3</b>	Luego digita su Usuario y Clave.
	<b>4</b>	Y presiona sobre el botón "Iniciar Sesión".
	<b>5</b>	El sistema verifica que el Usuario y clave estén correctos e inicia la sesión.
<b>POST-CONDICIÓN</b>	El usuario ingresa al sistema	
<b>EXECPCIONES</b>	<b>PASO</b>	<b>ACCIÓN</b>
	<b>5</b>	Si el usuario no está registrado en el sistema de la base de datos, se emite un mensaje indicando que no tiene permiso para acceder, como también si no hay conectividad al servidor presenta un mensaje de fallo en el servicio.
<b>COMENTARIOS</b>	El administrador habilitar para que el sistema posee más usuarios	

**Tabla 8** Caso de uso: Inicio de sesión

**Fuente:** Investigador

En los siguientes dos casos de uso se explica como gestionar los distintos dispositivos de manera manual y autónoma.

<b>CASO DE USO 02</b>	Gestionar el control de los dispositivos.	
<b>PRE-CONDICIÓN</b>	Realizar la gestión de activación de dispositivos.	
<b>DESCRIPCIÓN</b>	El usuario solicita la activación o desactivación de los diferentes dispositivos de control.	
<b>SECUENCIA</b>	<b>PASO</b>	<b>ACCIONES</b>
	<b>1</b>	Inicia cuando el usuario presiona sobre el botón “ACT.” o “DESACTIVADO” en la pantalla “Control de Dispositivos”.
	<b>2</b>	En la pantalla del móvil se cambia de estado en los botones de control y activamos los dispositivos.
	<b>3</b>	El sistema de manera remota envía una orden hacia el sistema domótico.
<b>POST-CONDICIÓN</b>	El usuario puede gestionar el apagado o encendido de dispositivos	
<b>EXECPCIONES</b>	<b>PASO</b>	<b>ACCIÓN</b>
<b>COMENTARIOS</b>	El usuario puede realizar la activación y desactivación de un total de siete dispositivos.	

**Tabla 9** Caso de uso: Gestión de control de dispositivos

**Fuente:** Autores

<b>CASO DE USO 03</b>	Programar modo automático a los dispositivos.	
<b>PRE-CONDICIÓN</b>	Realizar una configuración de modo automático	
<b>DESCRIPCIÓN</b>	El usuario configura la hora y minuto de encendido como también la hora y minuto de apagado de cada uno de los dispositivos para su control de manera automática.	
<b>SECUENCIA</b>	<b>PASO</b>	<b>ACCIONES</b>
	<b>1</b>	Inicia cuando el usuario presiona sobre cualquiera de los 7 botones como “Patio”, etc. en su pantalla del móvil.
	<b>2</b>	Muestra el Formulario “Control de Dispositivos Área/Dispositivo”.
	<b>3</b>	Se retorna a la pantalla de “Control de Dispositivos” y coloca una letra A que indica que esta en modo Automático.
<b>POST-CONDICIÓN</b>	El usuario puede realizar un encendido o apagado autónomo de los dispositivos	
<b>EXECPCIONES</b>	<b>PASO</b>	<b>ACCIÓN</b>
<b>COMENTARIOS</b>	Para realizar el modo automático es necesario activar su casilla de verificación.	

**Tabla 10** Caso de uso: Control de dispositivos de modo automático

**Fuente:** El Investigador



<b>CASO DE USO 04</b>	Ver Reporte de dispositivos.	
<b>PRE-CONDICIÓN</b>	Ver el estado de los dispositivos mediante la búsqueda por un rango de fecha establecidas.	
<b>DESCRIPCIÓN</b>	El usuario elige los parámetros (fecha y hora inicial, fecha y hora final) para visualizar el reporte de los dispositivos.	
<b>SECUENCIA</b>	<b>PASO</b>	<b>ACCIONES</b>
	<b>1</b>	Inicia cuando el usuario desea saber el estado anterior de los dispositivos en el hogar.
	<b>2</b>	El usuario en la pantalla “Control de Dispositivos” debe presionar sobre el botón “Ver Reportes”.
	<b>3</b>	El sistema muestra el Formulario “Reporte del Control de Luces”.
	<b>4</b>	El usuario escribe los parámetros (fecha y hora inicial, fecha y hora final) en el formato que está presente y al finalizar de colocar todos automáticamente presenta los datos del rango establecido.
	<b>5</b>	El sistema verifica que exista al menos un parámetro de búsqueda seleccionado.
	<b>6</b>	El sistema muestra el Reporte de los Dispositivos (Id. Puerto, Descripción, Fecha, Estado) en el registro.
<b>POST-CONDICIÓN</b>	El usuario puede ver los reportes de control	
<b>EXECPCIONES</b>	<b>PASO</b>	<b>ACCIÓN</b>
	<b>5</b>	Si no existe ningún rango asignado para el reporte no se presenta nada.
<b>COMENTARIOS</b>	El registro de control permite al usuario o administrador realizar la correcta gestión de los dispositivos.	

**Tabla 11** Caso de uso: Ver Reporte de Dispositivos

**Fuente:** El investigador

<b>CASO DE USO 05</b>	Cerrar Sesión	
<b>PRE-CONDICIÓN</b>	Finalizar el acceso al sistema	
<b>DESCRIPCIÓN</b>	El usuario cierra la sesión del sistema.	
<b>SECUENCIA</b>	<b>PASO</b>	<b>ACCIONES</b>
	<b>1</b>	Inicia cuando el usuario presiona sobre el botón “Cerrar” en la pantalla “Control de Dispositivos”.
	<b>2</b>	El sistema cierra la sesión del usuario y regresa a la pantalla principal “Domótica Mobile V1.0.0”.
<b>POST-CONDICIÓN</b>	El usuario ha salido del sistema	
<b>EXECPCIONES</b>	<b>PASO</b>	<b>ACCIÓN</b>
<b>COMENTARIOS</b>		

**Tabla 12** Caso de uso: Cierre de Sesión

**Fuente:** El investigador

# **CAPÍTULO V**

## **CONCLUSIONES Y RECOMENDACIONES**

## 5 CONCLUSIONES Y RECOMENDACIONES

### 5.1 Conclusiones

Al finalizar el presente trabajo de investigación y desarrollo se ha llegado a las siguientes conclusiones:

- Con el análisis e identificación de la tecnología Arduino se logró escoger el sistema modular más adecuado para así de esta manera diseñar nuestro circuito electrónico integral, se eligió como interfaz electrónica el Arduino Uno por su amplia memoria flash como también en el bajo requerimiento de los pines o puertos para las salidas digitales hacia los dispositivos a controlar y su bajo costo. Para la comunicación con el servidor y transporte de datos se eligió el módulo Wifi Tinsine Shield por poseer todos los protocolos de conectividad, un bajo consumo de energía y una antena externa de 3dbi la misma que nos da una seguridad en la conexión hacia el servidor de datos y el Router a usarse dentro de nuestra red Wifi.
- Los procesos de encendido o apagado de luminarias o artefactos electrónicos, fueron automatizados mediante la implementación de la aplicación web, móvil y el circuito electrónico integral domótico, el cual consta de una tarjeta control Arduino Uno la cual ejecuta las distintas operaciones de los dispositivos finales del sistema mediante un módulo de relay, para la transmisión inalámbrica se utilizó el módulo Tinsine el cual posee elementos que corrigen interferencias.
- La correcta gestión de la información obtenida del sistema domótico integral fue administrada de manera precisa en el servidor de datos como también mediante sus aplicaciones las cuáles se desarrollaron mediante VisualBasic.NET para generar la pantalla principal y su respectiva base de datos realizado mediante PostgreSQL para obtener reportes de control. La aplicación móvil se la realizó mediante la plataforma Android Studio para controlar los dispositivos de manera remota.

## 5.2 Recomendaciones

- Establecer privilegios a nivel de software para usuario básico o administrador, de esta manera gestionar la base de datos y aplicaciones de forma independiente.
- Implementar sensores de carga eléctrica en cada punto para obtener un control y registro real del consumo eléctrico de cada uno de los dispositivos.
- Producir un modelo real a partir del prototipo domótico integral que pueda ser instalado en diferentes áreas de hogares o instituciones según las necesidades requeridas.
- Realizar un método de comprobación o sistema redundante que nos garantice la conectividad de manera fiable en caso de fallo eléctrico con el servidor.
- Utilizar la documentación de ayuda con el fin que el usuario pueda tener mejor interacción.

# **ANEXOS**

## ANEXO A: Certificación plagio académico sistema Urkund



Universidad Técnica Estatal de Quevedo  
Facultad de Ciencias de la Ingeniería



**Fecha:** 16 de Diciembre del 2016

**De:** Ing. Efraín Díaz M.

**Para:** Ing. Roque Vivas Moreira, Director de la Unidad de Posgrado

**Asunto:** Informe de Urkund

Mediante la presente cumpla en presentar a usted el informe de tesis del Ing. José Luis Villarroel Intriago cuyo tema: **“EL CONTROL Y LAS TELECOMUNICACIONES COMO INFLUENCIA EN EL DESARROLLO TECNOLÓGICO DEL HOGAR. IMPLEMENTACIÓN DE UN SISTEMA DOMÓTICO MEDIANTE EL CONTROL DE UN TELÉFONO CELULAR CON PLATAFORMA ANDROID USANDO TECNOLOGÍA ARDUINO”**, revisado bajo mi dirección, toda vez que se ha desarrollado de acuerdo al reglamento general de graduación de posgrado de la Universidad Técnica Estatal de Quevedo y cumple con el requerimiento de análisis URKUND el cual avala los niveles de originalidad en un 97%, y de similitud de 3%.



### Urkund Analysis Result

**Analysed Document:** Proyecto Domotica - Maestria en redes y ordenadores.docx (D24396048)  
**Submitted:** 2016-12-16 06:49:00  
**Submitted By:** josephlouis007@hotmail.com  
**Significance:** 3 %

#### Sources included in the report:

Prototipo para monitoreo y control de personas adultas mayores utilizando tecnologia ZigBee.docx (D24351626)  
<http://www.applesfera.com/apple-1/ios-vs-android-la-eterna-batalla-por-combatir-la-fragmentacion>  
<http://tabletas.about.com/od/Conoce-las-tabletas/fl/Que-es-Android.htm>  
<http://androidzone.org/2013/05/historia-de-android-la-evolucion-a-lo-largo-de-sus-versiones/>  
<http://www.ibrugor.com/blog/apache-http-server-que-es-como-funciona-y-para-que-sirve/>

#### Instances where selected sources appear:

8

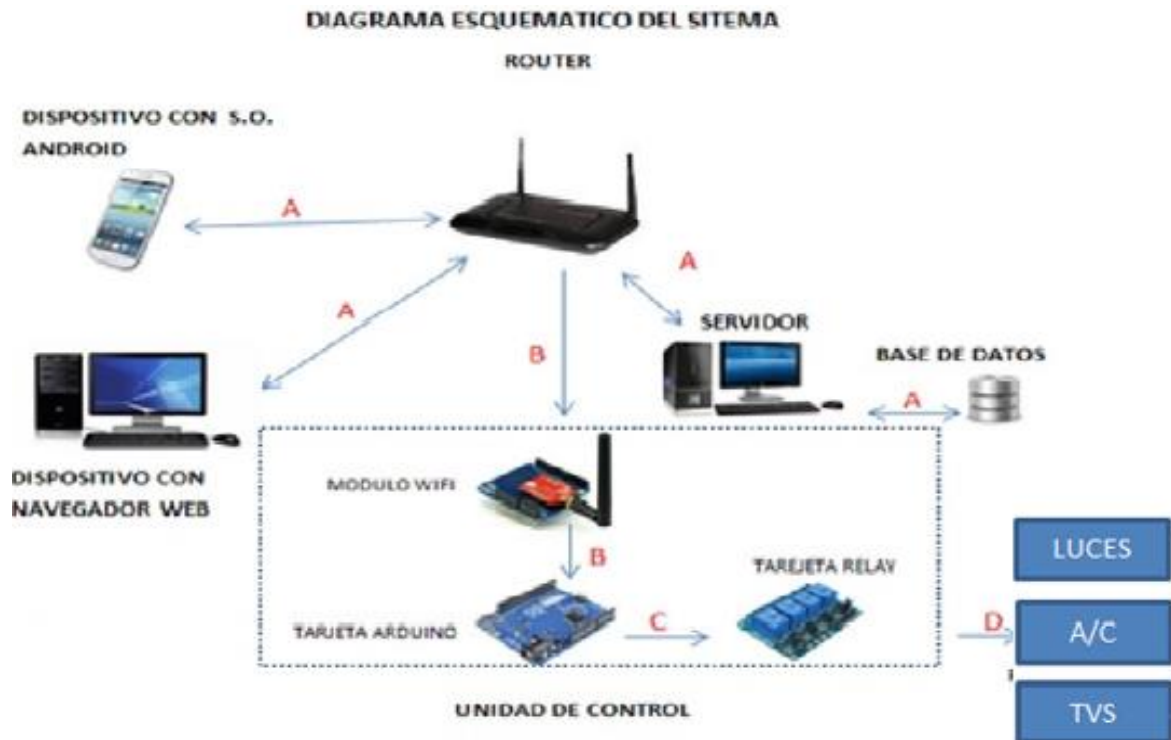
Particular que informo para trámites pertinentes, de acuerdo a lo que establece el reglamento de grados y títulos de la Universidad Técnica Estatal de Quevedo.

Por la atención a la presente reitero mis agradecimientos.

Atentamente

  
Ing. Efraín Díaz Macías, MSc.  
DIRECTOR DE TESIS

## ANEXO B: FUNCIONAMIENTO DEL SISTEMA



- A) Datos de los puertos (encendido, apagado, nombre, automático, hora apagado, hora encendido).
- B) Datos de encendido y apagado.
- C) Señal lógica digital de encendido o apagado para las luces.
- D) Corriente eléctrica para el encendido de las luces.

## IMPLEMENTACIÓN DEL HARDWARE



## DESCRIPCION DE FUNCIONAMIENTO:

Los dispositivos se encuentran conectados en una red LAN a través de un punto de acceso (ROUTER con salida a Internet), en la LAN se encuentra un servidor que contiene la base de datos (Apache/PostgreSQL) y los programas de necesarios (Php, VisualBasic.Net, AndroidStudio) para brindar respuesta a las solicitudes realizadas por los dispositivos de la red como pueden ser cambio de estado de los puertos o consultas de estado de los mismos.

## DISPOSITIVO CON S.O. ANDROID

Tiene instalada una apk. o aplicación para android la misma que fue desarrollada en Android Studio, la cual tiene una interface gráfica para poder realizar solicitudes al servidor tales como consultar el estado actual de los puertos o realizar cambio de estados de los mismos y de esta manera encender o apagar luminarias o artefactos electrónicos.

## SERVIDOR

En este servidor corre un programa llamado **“Conexión a Arduino”** realizado en VisualBasic.Net, que permite establecer un enlace por medio de una conexión TCP cliente a través del puerto 2000, que será el puerto que estará escuchando arduino para recibir los datos de encendido o apagado de los puertos ya que Arduino utiliza este puerto de comunicación para transporte de paquetes de datos o tramas.

Para el caso de la aplicación android realizada en Android Studio utiliza un servicio WebApi (WebApi) para la conexión directa al sistema administrador de la base de datos.

El sistema recibe las solicitudes de los dispositivos de la red, a través del puerto 8080 si es desde un navegador web, las solicitudes son ejecutadas por el servidor en el caso del puerto 8080 el servidor utiliza php para comunicarse con la base de datos.



La trama enviada hacia arduino es recibida por el módulo wWifi Tinsine, una vez recibido por el módulo Wifi este pasa la información vía una comunicación serial por medio de los pines P0(B0) y P1(B1) de Arduino. La trama está compuesta por 6 bytes que representan el puerto su estado y la verificación de errores de la trama para prevenir encendido o apagados no deseados por motivos de pérdida, debilidad o ruido de la señal.

### UNIDAD DE DATO

10001

**Azul:** valor del puerto (0=encendido; 1=apagado)

**Verde:** Id del puerto.

**Rojo:** valor del checksum (si el valor del puerto es 1 se suma 1 al Id del puerto y este será el valor de checksum)

### TRAMA

La trama está compuesta por 12 unidades de datos y tiene como inicio de trama el carácter!

### DISPOSITIVO CON NAVEGADOR WEB

Cuenta con un navegador WEB que permite entrar al servidor a través del puerto 8080 y solicitar la interfaz gráfica que permitirá realizar solicitudes al servidor tales como consultar el estado actual de los puertos o realizar cambio de estados de los mismos además de poder realizar un reporte del historial de los estados que han tenido los puertos al transcurrir el tiempo.

### UNIDAD DE CONTROL

Compuesta por 3 dispositivos: Módulo Arduino, Módulo Wifi Tinsine para Arduino, y 2 la Tarjeta de Relay de 12 voltios.

El módulo wifi recibe la trama que es enviada cada 300ms desde el servidor, una vez recibida la trama es enviada por medio de una comunicación serial a la tarjeta arduino, arduino procesa los datos y pone en sus puertos los valores recibidos; la tarjeta relay recibe los estado TTL de los puertos de arduino y conecta o desconecta los contactos de los relay dependiendo del valor del puerto.

Si un puerto arduino tiene el valor de 0 entonces el relay conectado a ese puerto cerrara sus contactos y encenderá una luminaria o controlara algún artefacto electrónico.

## ANEXO C: MANUAL DE USUARIO

### PANTALLA DE ACCESO A LA APLICACIÓN:

Para ingresar a la aplicación seleccionar el icono Aplicación Domótica Mobile que se encuentra en la pantalla del dispositivo móvil y hacer doble clic.



### PANTALLA DE INGRESO DE USUARIO:

Luego para ingresar a la pantalla de control debemos ingresar el usuario final (admin) y el password o contraseña (123) que se encuentra por default en la base de datos.



## PANTALLA DE CONTROL DE LA APLICACIÓN:

En esta pantalla se muestra las diferentes partes de esta aplicación.



Esta pantalla permite el encendido o apagado de las luminarias o artefactos conectados en su borneras de control en el Arduino, visualización del estado (Encendido, Apagado y Automático) de las mismas y acceso a la configuración de cada una de ellas para modo automático como también al acceso de los reportes diarios.

**Botones de control.-** Sirven para encender y apagar las lámparas e indican el estado de las mismas.

**Nombres actuales.-** Indica el nombre actual de la lámpara como también al hacer clic en estos podemos hacer uso del modo de configuración automático de cada uno de estos.

## ENCENDIDO DE LUMINARIAS O ARTEFACTOS ELECTRONICOS:

Para realizar el encendido de alguna luminaria o artefacto electrónico se debe dar clic en el botón de control al que se quiere activar y de cambia a modo ACT.

## APAGADO DE LUMINARIAS O ARTEFACTOS ELECTRONICOS:

Para realizar el apagado de alguna luminaria o artefacto electrónico se debe dar clic en el botón de control al que se quiere desactivar y de cambia a modo DESACTIVADO.

## PANTALLA DE ENCENDIDO/APAGADO DE LUMINARIAS O ARTEFACTOS ELECTRONICOS EN MODO AUTOMÁTICO:

En esta pantalla se puede programar un encendido/apagado de cada luminaria o artefacto electrónico conectado en el arduino a una determinada hora específica, ir a la parte de la “HORA DE ENCENDIDO” y “MINUTO DE ENCENDIDO” para programar la hora del encendido y después ir a la parte de la “HORA DE APAGADO” y “MINUTO DE APAGADO” para programar la hora del apagado de luminarias o artefacto electrónico, luego finalmente dar clic en el botón PROGRAMAR y luego se regresa al menú principal de control y se activa el modo automático indicando con la letra “A”.

Si no se desea realizar ningún cambio se debe presionar el botón CANCELAR

**Domótica Mobile**  
Control de Dispositivos  
Area/Dispositivo: Patio

Hora de encendido: 12  
Minuto de encendido: 05  
Hora de apagado: 12  
Minuto de apagado: 07

BOTON PARA PROGRAMAR → PROGRAMAR  
← CANCELAR BOTON PARA CANCELAR CONFIGURACION

**Domótica Mobile**  
Control de Dispositivos

PATIO	A	DESACTIVADO
BAÑO		DESACTIVADO
DORMITORIO		ACT.
COCINA		DESACTIVADO
COMEDOR		ACT.
SALA		DESACTIVADO
SPLIT		ACT.
VER REPORTE		
CERRAR		

**Configuración de encendido.-** Permite asignar la hora y minuto de encendido automático de la luminaria o artefacto en modo automático.

**Configuración de apagado.-** Asigna la hora y minuto de apagado automático de la luminaria o artefacto en modo automático.

**Botón programar-** Guarda las configuraciones realizadas.

A continuación daremos a conocer cada una de las pantallas de configuración en modo automático



**Domótica Mobile**  
Control de Dispositivos  
Area/Dispositivo: Dormitorio

Hora de encendido:

Minuto de encendido:

Hora de apagado:

Minuto de apagado:

BOTON PARA PROGRAMAR  MODO AUTOMATICO

CANCELAR

**Domótica Mobile**  
Control de Dispositivos  
Area/Dispositivo: Baño

Hora de encendido:

Minuto de encendido:

Hora de apagado:

Minuto de apagado:

BOTON PARA PROGRAMAR  MODO AUTOMATICO

CANCELAR

**Domótica Mobile**  
Control de Dispositivos  
Area/Dispositivo: Sala

Hora de encendido:

Minuto de encendido:

Hora de apagado:

Minuto de apagado:

BOTON PARA PROGRAMAR  MODO AUTOMATICO

CANCELAR

**Domótica Mobile**  
Control de Dispositivos  
Area/Dispositivo: Split

Hora de encendido:

Minuto de encendido:

Hora de apagado:

Minuto de apagado:

BOTON PARA PROGRAMAR  MODO AUTOMATICO

CANCELAR

## PANTALLA DE CONTROL DE REPORTES:

Aquí podemos nosotros realizar consultas del encendido y apagado de luminarias y artefactos electrónicos conectados mediante el cambio del rango de la fecha y hora en los cuadros establecidos en el formato establecido:

Desde: dia/mes/año espacio hora/minuto/segundo

Hasta: dia/mes/año espacio hora/minuto/segundo



ID	PUERTO	DESCRIPCION	FECHA	ESTADO
2		Sala	06/06/2016 06:23:50	Apagado
5		Dormitorio	06/06/2016 06:23:46	Encendido
7		Patio	06/06/2016 06:23:43	Encendido
4		Cocina	06/06/2016 06:23:41	Apagado
7		Patio	06/06/2016 06:23:39	Apagado
5		Dormitorio	06/06/2016 06:23:35	Apagado
6		Baño	06/06/2016 06:23:35	Apagado
7		Patio	06/06/2016 06:23:34	Encendido
7		Patio	06/06/2016 06:23:33	Apagado
7		Patio	06/06/2016 06:23:30	Encendido
7		Patio	06/06/2016 06:22:53	Apagado
8		Split	21/05/2016 16:20:02	Encendido
2		Sala	21/05/2016 16:20:01	Encendido
3		Comedor	21/05/2016 16:20:00	Encendido
4		Cocina	21/05/2016 16:19:59	Encendido
5		Dormitorio	21/05/2016 16:19:59	Encendido
6		Baño	21/05/2016 16:19:58	Encendido
7		Patio	21/05/2016 16:19:57	Encendido
8		Split	21/05/2016 16:19:56	Apagado
2		Sala	21/05/2016 16:19:56	Apagado
3		Comedor	21/05/2016 16:19:55	Apagado
4		Cocina	21/05/2016 16:19:54	Apagado
5		Dormitorio	21/05/2016 16:19:53	Apagado
6		Baño	21/05/2016 16:19:52	Apagado
7		Patio	21/05/2016 16:19:49	Apagado
8		Split	16/05/2016 18:20:03	Encendido
7		Patio	16/05/2016 18:19:59	Encendido
6		Baño	16/05/2016 18:19:58	Encendido
5		Dormitorio	16/05/2016 18:19:56	Encendido
4		Cocina	16/05/2016 18:19:57	Encendido
3		Comedor	16/05/2016 18:19:56	Encendido
2		Sala	16/05/2016 18:19:55	Encendido

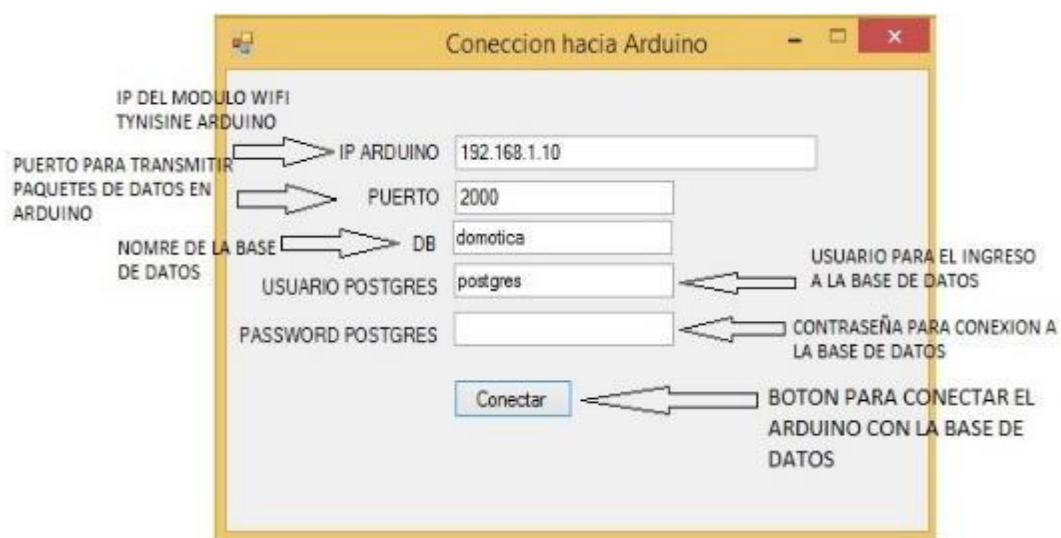


## APLICACIÓN INTERMEDIARIA ENTRE ARDUINO Y LA BASE DE DATOS

Dar doble clic en el icono ejecutable de la aplicación.



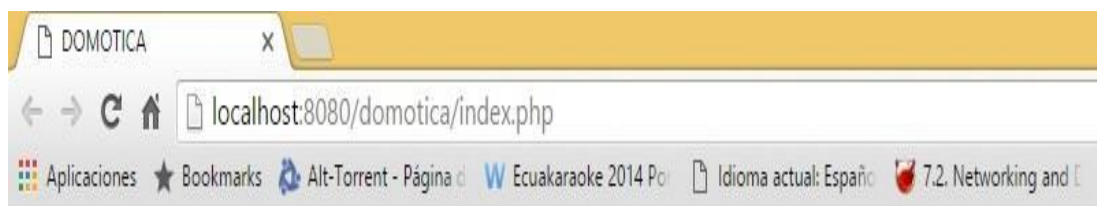
En la siguiente pantalla se realiza la conexión entre arduino y la base de datos, se debe ingresar la dirección IP de arduino (Host), el puerto al que corresponde (2000 por default puerto para transmitir paquetes de arduino), nombre de la base de datos (domotica), usuario de la base de datos (postgres) y contraseña o password (postgres) y luego dar clic en conectar.



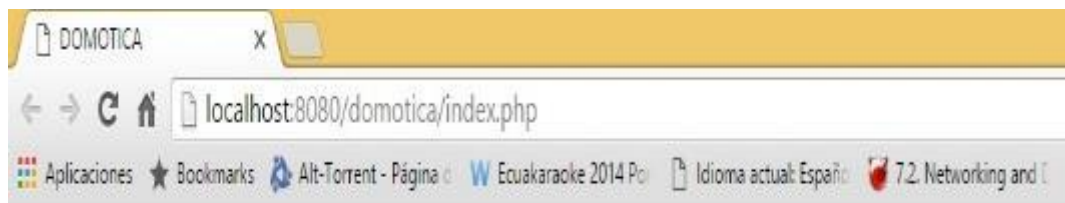
Se observa que la conexión fue establecida correctamente cuando presenta en la pantalla CONECTADO.

## ACCESO A LA APLICACIÓN DESDE EL PORTAL WEB

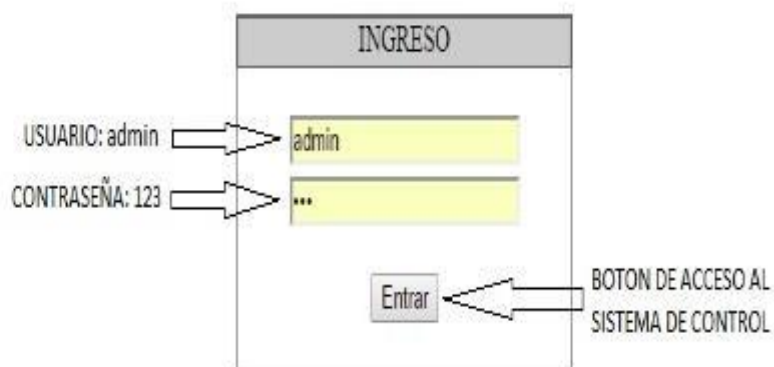
Ingresar en la barra de dirección del navegador la dirección IP o dominio del servidor en este caso localhost:8080/domótica/index.php o la dirección ip de la maquina 192.168.1.50:8080/domótica/index.php o de forma externa mediante la ip\_publica\_privada:8080/domótica/index.php'.



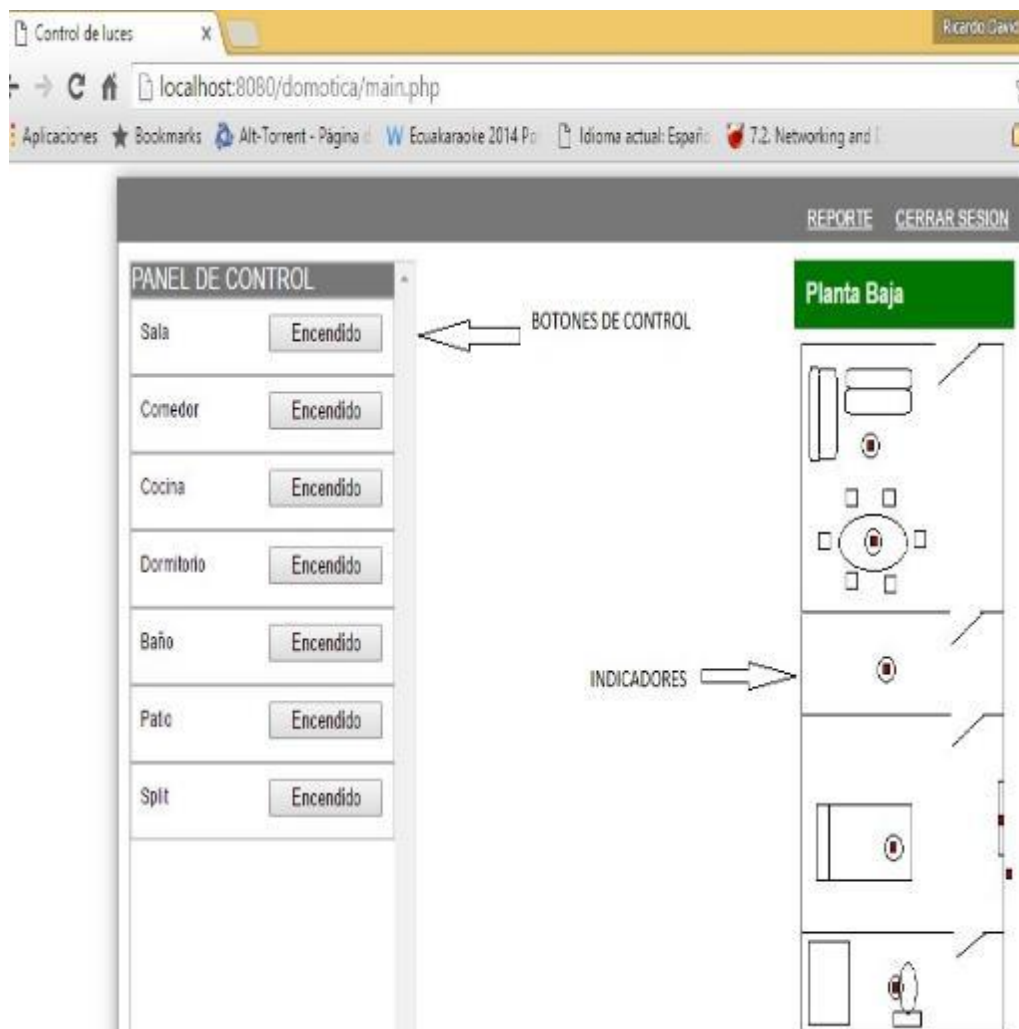
En la siguiente pantalla se debe ingresar el usuario y la clave suministrado por el administrador del sistema la misma que se encuentra asignada en la base de datos, luego dar clic en el botón ENTRAR.



## SISTEMA DE CONTROL DOMOTICO



En la siguiente ventana se puede observar los diferentes botones de control en el entorno web.

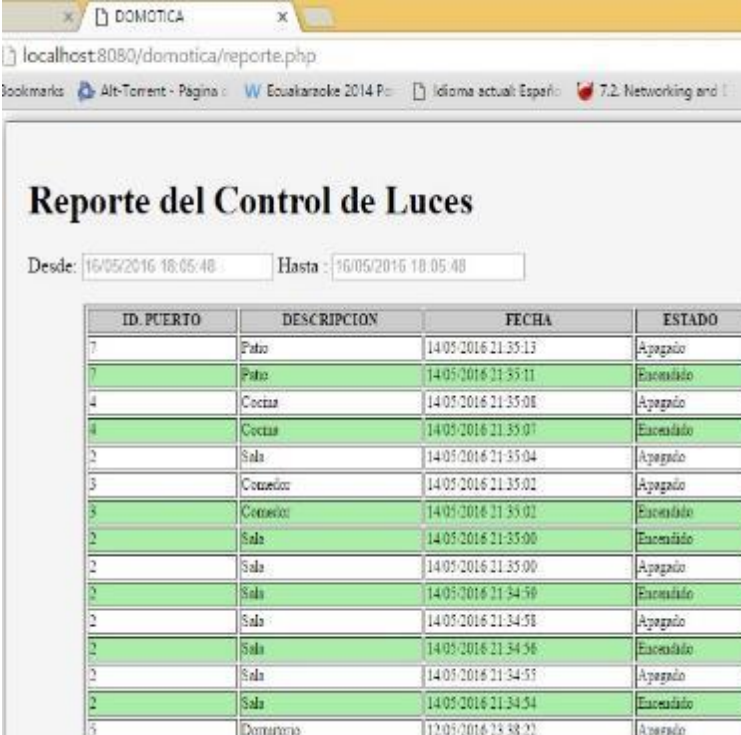


Para realizar un encendido o apagado de una luminaria o artefacto electrónico se debe de posicionar sobre el mismo y dar clic para encenderlo o apagarlo.

El mismo proceso se debe de realizar en caso de que se quiera apagar una luminaria o artefacto electrónico. Se puede observar en el siguiente reporte de control de luminaria o artefacto electrónicos encendidos y luminaria o artefacto electrónicos se encuentran apagados o encendidos donde también se pueden cambiar los rangos de búsqueda pero respetando el formato siguiente:


Desde: dia/mes/año espacio hora/minuto/segundo

Hasta: día/mes/año espacio hora/minuto/segundo



ID	PUERTO	DESCRIPCION	FECHA	ESTADO
7		Patio	14/05/2016 21:35:13	Apagado
7		Patio	14/05/2016 21:35:11	Encendido
4		Cocina	14/05/2016 21:35:08	Apagado
4		Cocina	14/05/2016 21:35:07	Encendido
2		Sala	14/05/2016 21:35:04	Apagado
3		Comedor	14/05/2016 21:35:02	Apagado
3		Comedor	14/05/2016 21:35:01	Encendido
2		Sala	14/05/2016 21:35:00	Encendido
2		Sala	14/05/2016 21:35:00	Apagado
2		Sala	14/05/2016 21:34:59	Encendido
2		Sala	14/05/2016 21:34:58	Apagado
2		Sala	14/05/2016 21:34:56	Encendido
2		Sala	14/05/2016 21:34:55	Apagado
2		Sala	14/05/2016 21:34:54	Encendido
5		Dormitorio	12/05/2016 23:38:22	Apagado

Por ultimo si se desea programar un bloque automáticamente para su encendido y posterior apagado a una determinado hora y minuto especifico se debe acceder al mismo y configurar la hora y minuto de encendido y la hora y minuto de apagado, después activar la casilla de verificación automática y finalmente dar clic en el botón guardar.



DOMOTICA

Baño

Automatico ☐

ON 23 : 45

OFF 2 : 1

Guardar

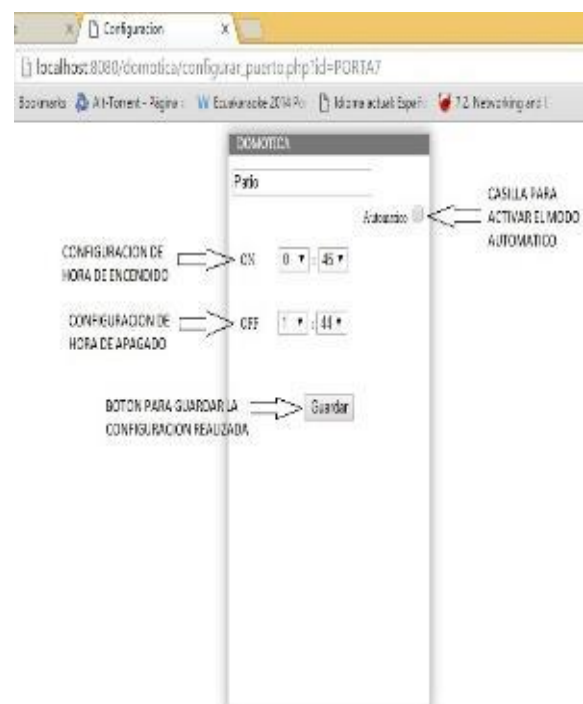
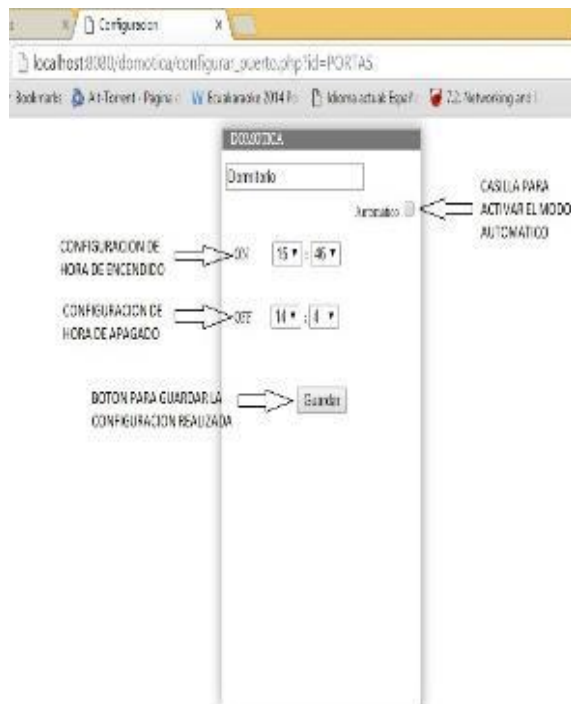
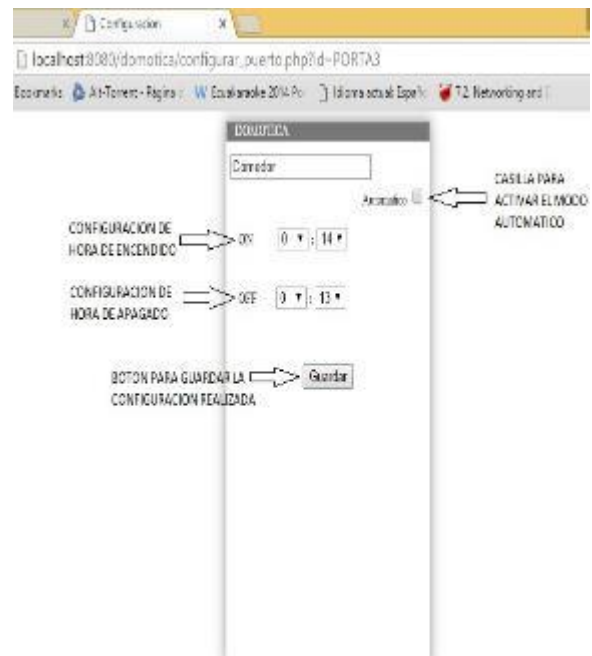
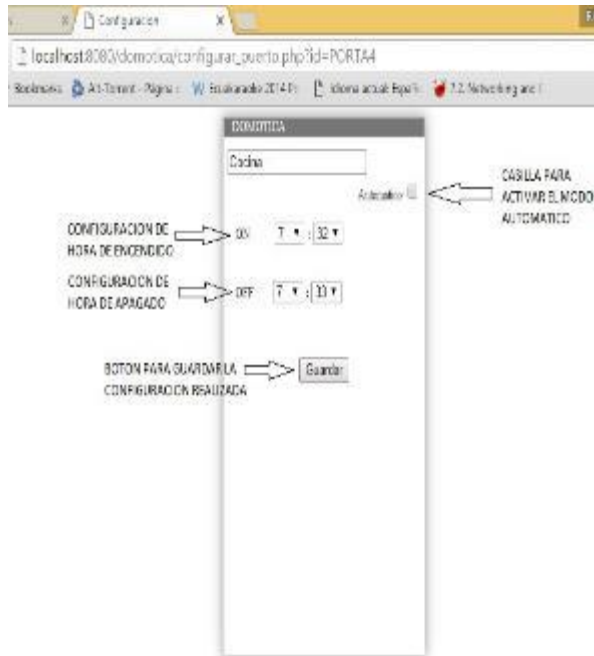
CONFIGURACION DE HORA DE ENCENDIDO

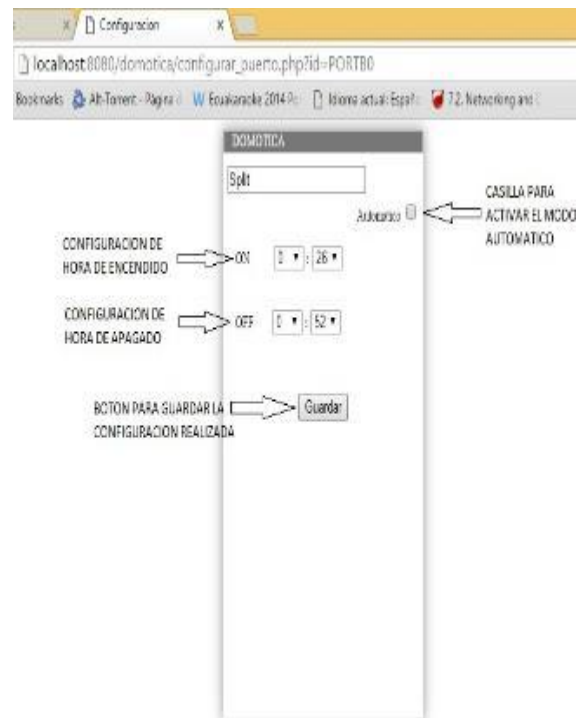
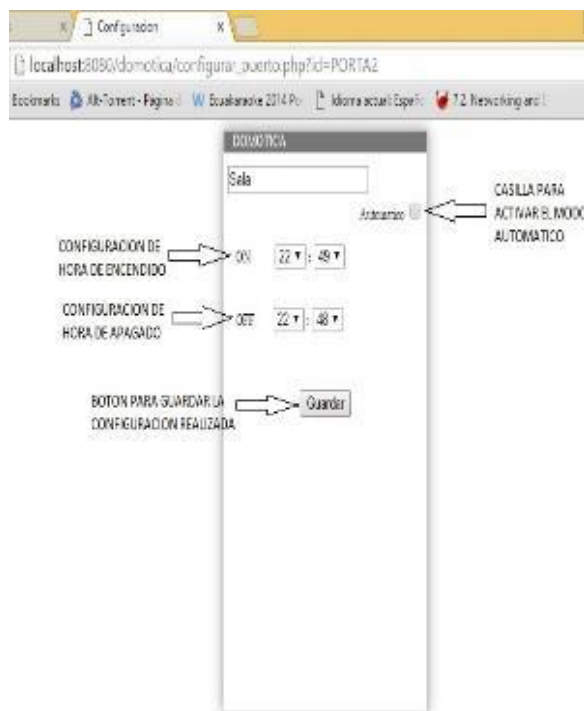
CONFIGURACION DE HORA DE APAGADO

BOTON PARA GUARDAR LA CONFIGURACION REALIZADA

CASILLA PARA ACTIVAR MODO AUTOMATICO

A continuación se muestran cada una de las pantallas de configuración para cada botón de control (cocina, comedor, dormitorio, patio, patio, split):





## ANEXO D

### **PROGRAMACION DE LA PAGINA WEB EN PHP**

#### **LOGOUT.PHP // FUNCION PARA SALIR Y REGRESAR AL MODULO DE INICIO O LOGIN**

```
<?php
session_start();
session_unset();
session_destroy();
print("<script>window.location.replace('index.php');</script>");
?>
```

#### **INDEX.PHP // MODULO PARA INGRESAR EL USUARIO Y CONTRASEÑA**

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>DOMOTICA</title>
</head>
<body>
<h1 align="center">SISTEMA DE CONTROL DOMOTICO</h1>
<div>
<div style="width:25%; border:#666666 thin solid; margin:auto; " align="center" >
<div style=" height:20px; border:#666666 thin solid;
background:#CCCCCC";color:#000000 >INGRESO</div>
<form method="post" action="main.php">
<input type="text" name="usuario" placeholder="usuario" style=" margin-bottom:5px;
margin-top:20px;"><br>
<input type="password" placeholder="clave" name="clave" style=" margin-
bottom:20px;"><br>
<input type="submit" name="entrar" value="Entrar" style=" margin-bottom:20px;" >
</form>
</div>
```

```
</div>
</body>
</html>
```

### **MODULO.PHP // FUNCION DE CONECCION A LA BASE DE DATOS**

```
<?php
function conectar()
{
//error_reporting(E_ALL ^ E_WARNING);
if (! $x= pg_connect("host=localhost user=postgres password=postgres port=5432
dbname=domotica"))
{
echo 'error de coneccion a la base de datos';
return ;
}
return $x;
}
function permiso($idPuerto)
{
$con=conectar();
$result=pg_query($con,"select * from privilegios where (idUserio=$_SESSION[usuario]
and idPuerto=$idPuerto) or (idUserio=$_SESSION[usuario] and idPuerto=255) ");
if (pg_num_rows($result)>0)
return 1;
else
return 0;
}
?>
```

### **MAIN.PHP //MODULO O PANTALLA PRINCIPAL**

```
<?php
session_start();
include("modulo.php");
$con= conectar();
```



```

extract($_POST);

$result=pg_query($con,"select * from usuarios where usuario='$usuario'");

if(pg_num_rows($result)<1){echo 'usuario no existe!!  <a href="index.php">Volver a
Intentar</a>';exit();}

$row= pg_fetch_array($result);

$_SESSION['usuario']=$row['usuario'];

if($row['clave']!=$clave){echo 'clave incorrecta!!  <a href="index.php">Volver a
Intentar</a>';exit();}

$tabla=pg_query($con,"select * from puertos order by idpuerto ");

$result=pg_query($con,"select items.* ,puertos.valor from items left join puertos on
items.idpuerto=puertos.idpuerto order by items.iditem  ");

?>

<html>

<head>

<script type="text/javascript">

function setValor(objeto)

{

idSwitch = objeto.id;

if (window.XMLHttpRequest)

xmlhttp=new XMLHttpRequest();

xmlhttp.open("GET","actualizar_estado_de_puertos.php?id="+idSwitch);

xmlhttp.send(null);

}

function todos(v)

{

if (window.XMLHttpRequest)

xmlhttp=new XMLHttpRequest();

xmlhttp.open("GET","actualizar_estado_de_puertos.php?todos="+v);

xmlhttp.send(null);

}

function ir(obj)

{

```

```

window.open("configurar_puerto.php?id="+obj.id);
}
</script>
</head>
<title>Control de luces</title>
<body style="font-family:Arial, Helvetica, sans-serif; ">
<script type="text/javascript">
var myVar=setInterval(function(){myTimer()},200);
function myTimer() {
if (window.XMLHttpRequest)
xmlhttp=new XMLHttpRequest();
xmlhttp.open("GET","estado_de_luces.php");
xmlhttp.onreadystatechange=function()
{
    if (xmlhttp.readyState==4)
    {
        var obj = JSON.parse(xmlhttp.responseText);
        for( i=0;i<12;i++ )
        document.getElementById(obj[i][0]).style.background=obj[i][1];
    }
}
xmlhttp.send(null);
}
var myVar2=setInterval(function(){myTimer2()},300);
function myTimer2() {
if (window.XMLHttpRequest)
xmlhttp2=new XMLHttpRequest();
xmlhttp2.open("GET","estado_de_los_puertos.php");
xmlhttp2.onreadystatechange=function()
{
    if (xmlhttp2.readyState==4)

```

```

    {
        var obj = JSON.parse(xmlhttp2.responseText);
        for( i=0;i<12;i++ )
        {
            if (obj[i][7]==true)
            {
                document.getElementById(obj[i][0]).value=obj[i][3];
                document.getElementById(obj[i][0]).disabled=obj[i][6];
                document.getElementById(obj[i][1]).innerHTML=obj[i][2];
                document.getElementById(obj[i][4]).innerHTML=obj[i][5];
            }
        }
    }
}

xmlhttp2.send(null);
}
</script>

<div style="box-shadow: 0px 0px 20px #999;width:800px; position:relative;
margin:auto;border:#777 thin solid; background:#FFF; height:auto;">

<div style=" padding:5px; background-color:#777; color:#fff; position:relative;height:25px "
><a style=" color:#FFFFFF;float:right;padding:10px; font-size:12px" href="logout.php"
>CERRAR SESION</a><a style=" margin-
left:20px;color:#FFFFFF;float:right;padding:10px; font-size:12px" href="reporte.php"
target="_blank" >REPORTE</a></div>

<div style="height:auto; padding:10px; " >

<div style="border:#ccc thin solid ;margin-left:auto; width:250px ; height: 450px; overflow-
y:scroll;; float:left ;position:relative;margin-bottom:10px">

<div style="background-color:#777; color:#fff; border:#aaa solid thin; " >PANEL DE
CONTROL</div>

<?php
while ($row= pg_fetch_array($tabla))
//for ($i=0 ;$i<=6;$i++)
{
//$row= pg_fetch_array($tabla);

```

```

if ($row['habilitado']=="t")
{
echo '<div style=" border:#aaa solid thin;padding:5px; height:auto" >';
echo '<table width="100%" style=" font-size:12px" >

<tr><td colspan="1" ><div onClick="ir(this);" style=" cursor:pointer;" title="'. $row['nombre']
.'" id="'. $row['nombre'] ." >'. $row['descripcion'] .'</div></td>

<td width="102" >';
if($row['valor']=="t")

echo '<input style=" width:100px;" type="button" id="'. $row['idpuerto'] ."
value="Encendido"  onClick="setValor(this)" />';

else

echo '<input style=" width:100px;" type="button" id="'. $row['idpuerto'] ."
value="Apagado"  onClick="setValor(this)" />';

echo '</td></tr>

<tr>

<td></td>

<td><span id="automatico'. $row['idpuerto'] ." style="font-size: 12px;color:#A00; margin-
left:10px" ></span>

</td>

</tr>

</table></div>';
}
}
?>

</div>

<div style="background-color:#070; height: 100px;position:relative ; float:right">

<div style="color:#eee; padding:10px; font-weight:bold">Planta Baja</div>

<div style="position:relative;" id="plano">



<?php $row= pg_fetch_array($result);?>

<div id="<?php echo 'item'. $row['iditem'];?>" title="<?php echo $row['nombre'].'&#13ID
PUERTO : '. $row['idpuerto'].'&#13VATIOS : '. $row['vatio'];?>" style="background-
color:#fff; border:#000000 solid thin;width:4px;position:absolute; height:4px;top:62px;
left:64px;" ></div>

```

```

<?php $row= pg_fetch_array($result);?>

<div id="<?php echo 'item'.$row['iditem'];?>" title="<?php echo $row['nombre'].'&#13ID
PUERTO : '.$row['idpuerto'].'&#13VATIOS : '.$row['vatio'];?>" style="background-color:#fff;
border:#000000 solid thin;width:4px;position:absolute; height:4px;top:115px; left:67px;"
></div>

<?php $row= pg_fetch_array($result);?>

<div id="<?php echo 'item'.$row['iditem'];?>" title="<?php echo $row['nombre'].'&#13ID
PUERTO : '.$row['idpuerto'].'&#13VATIOS : '.$row['vatio'];?>" style="background-
color:#fff; border:#000000 solid thin;width:4px;position:absolute; height:4px;top:186px;
left:78px;" ></div>

<?php $row= pg_fetch_array($result);?>

<div id="<?php echo 'item'.$row['iditem'];?>" title="<?php echo $row['nombre'].'&#13ID
PUERTO : '.$row['idpuerto'].'&#13VATIOS : '.$row['vatio'];?>" style="background-
color:#fff; border:#000000 solid thin;width:4px;position:absolute; height:4px;top:285px;
left:84px;" ></div>

<?php $row= pg_fetch_array($result);?>

<div id="<?php echo 'item'.$row['iditem'];?>" title="<?php echo $row['nombre'].'&#13ID
PUERTO : '.$row['idpuerto'].'&#13VATIOS : '.$row['vatio'];?>" style="background-
color:#fff; border:#000000 solid thin;width:4px;position:absolute; height:4px;top:363px;
left:86px;" ></div>

<?php $row= pg_fetch_array($result);?>

<div id="<?php echo 'item'.$row['iditem'];?>" title="<?php echo $row['nombre'].'&#13ID
PUERTO : '.$row['idpuerto'].'&#13VATIOS : '.$row['vatio'];?>" style="background-
color:#fff; border:#000000 solid thin;width:4px;position:absolute; height:4px;top:300px;
left:186px;" ></div>

<?php $row= pg_fetch_array($result);?>

<div id="<?php echo 'item'.$row['iditem'];?>" title="<?php echo $row['nombre'].'&#13ID
PUERTO : '.$row['idpuerto'].'&#13VATIOS : '.$row['vatio'];?>" style="background-
color:#fff; border:#000000 solid thin;width:4px;position:absolute; height:4px;top:270px;
left:179px;" ></div>

</div>

</div>

</div>

<div style=" height:25px; background-color:#777; color:#fff; clear:both;"
align="center"></div>

</div>

</body>

</html>

```

**ACTUALIZAR CONFIGURACION.PHP //MODULO QUE ACTUALIZA EN LA BASE DE DATOS LOS CAMBIOS AL SELECCIONAR LA CASILLA AUTOMATICO**

```
<?php
session_start();
include("modulo.php");
$con= conectar();
extract($_GET);
if(isset($auto))
$automatico="TRUE";
else
$automatico="FALSE";
if($horaon=="")
$horaon=0;
if($minutoon=="")
$minutoon=0;
if($horaoff=="")
$horaoff=0;
if($minutooff=="")
$minutooff=0;
$tabla=pg_query($con,"update puertos set
descripcion='$descripcion',automatico='$automatico',horaon=$horaon,minutoon=$minuto
on,horaoff=$horaoff,minutooff=$minutooff where nombre='$nombre' ");
if($tabla)
print "<script>alert('Se actualizo el Equipo Correctamente');window.close();</script>";
else
print "<script>alert('Hubo un error al actualizar el Equipo');window.close();</script>";
?>
```

**ACTUALIZAR ESTADO DE PUERTOS.PHP //MODULO LEE DE LA BASE DE DATOS Y ACTUALIZA LOS ESTADOS DE LOS BOTONES DE CONTROL EN LA PANTALLA**

```
<?php
include("modulo.php");
```

```

$con= conectar();
extract($_GET);
if (isset($todos))
{
if ($todos==1)
$valor="t";
else
$valor="f";
pg_query($con,"update puertos set valor='$valor'");
exit();
}
if($id>1)
{
$tabla=pg_query($con,"select idpuerto,valor from puertos where idpuerto=$id");
$row=pg_fetch_array($tabla);
if($row[1]=="f")
$valor="t";
else
$valor="f";
pg_query($con,"update puertos set valor='$valor' where idpuerto=$id");
}
?>

```

**ESTADO DE PUERTOS.PHP //MODULO LEE DE BASE DE DATOS Y DIBUJA LA PANTALLA DE CONTROL**

```

<?php
include("modulo.php");
$con= conectar();
//extract($_GET);
$tabla=pg_query($con,"select * from puertos ");
$c=0;
$puertos = array();

```

```

while($row=pg_fetch_array($tabla))
{
if($row['valor']=="f")
$valor="Apagado";
else
$valor="Encendido";
if($row['habilitado']=="f")
$ena=false;
else
$ena=true;
if( $row['horaon']<10)
$horaon = '0'.$row['horaon'];
else
$horaon= $row['horaon'] ;
if( $row['minutoon']<10)
$minutoon= '0'.$row['minutoon'] ;
else
$minutoon= $row['minutoon'];
if( $row['horaoff']<10)
$horaoff= '0'.$row['horaoff'] ;
else
$horaoff= $row['horaoff'] ;
if( $row['minutooff']<10)
$minutooff= '0'. $row['minutooff'];
else
$minutooff= $row['minutooff'];
$hora= $horaon.':'.$minutoon.' - '.$horaoff.':'.$minutooff;
if( $row['automatico']=="f")
{
$hora="";
$habilitado=false;

```



```

}
else
$habilitado=true;

$puertos[$c]=array($row['idpuerto'],$row['nombre'],$row['descripcion'],$valor,'automatico'.
$row['idpuerto'],$hora,$habilitado,$ena);

$c=$c+1;
$hora="";
}

$e = json_encode($puertos);

echo $e;

?>

```

### **CONFIGURAR PUERTO.PHP //MODULO DIBUJA EN LA PANTALLA PARA CONFIGURAR LOS BOTONES DE CONTROL**

```

<?php
include("modulo.php");

$con=conectar();

extract($_GET);

$tabla=pg_query($con,"select * from puertos where nombre='$id'");
$row=pg_fetch_array($tabla);

?>

<html xmlns="http://www.w3.org/1999/xhtml">
<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<title>Configuracion</title>
</head>

<body style="font-size:12px">

<form action="actualizar_configuracion.php" method="get" target="_self">

<input style="margin-top:10px;" type="hidden" name="nombre" id="nombre"
value="<?php echo $id ?>">

<div style="box-shadow: 0px 0px 20px #999;border:#ccc thin solid ;margin:auto;
width:250px ; height:450px; position:relative;margin-bottom:10px;">

<div style="background-color:#777; color:#fff; border:#aaa solid thin; "><span style="
margin-left:10px">DOMOTICA</span></div>

```

```

<div style="position:relative; padding:5px">

<div style="position:relative"><input style="margin-top:5px;" name="descripcion"
id="descripcion" type="text" value="<?php echo $row['descripcion'];?>"/><span
style="float:right;margin-top:5px;"> Automatico <input name="auto" id="auto"
type="checkbox" <?php if ($row['automatico']=="t") echo "checked"; else echo
"unchecked"; ?> /> </span> </div>

<div id="horas" style=" width:200px; margin-top:30px;padding:10px;">

<div style=" margin-bottom:30px;">ON

<span style=" margin-left:27px;">

<select name="horaon" id="horaon">

    <?php
    for ($i=0;$i<24;$i++)
    {
        if ($i == $row['horaon'])
            echo " <option selected=selected value=$i >$i</option>";
        else
            echo " <option value=$i >$i</option>";
    }
    ?>

</select>

</span>

<b> : </b>

<span>

<select name="minutoon" id="minutoon">

    <?php
    for ($i=0;$i<60;$i++)
    {
        if ($i == $row['minutoon'])
            echo " <option selected=selected value=$i >$i</option>";
        else
            echo " <option value=$i >$i</option>";
    }

```

```

        ?>
    </select>

</span>
</div>
<div>OFF
<span style=" margin-left:20px">
<select name="horaoff" id="horaoff">
    <?php
    for ($i=0;$i<24;$i++)
    {
    if ($i == $row['horaoff'])
    echo " <option selected=selected value=$i >$i</option>";
    else
    echo " <option value=$i >$i</option>";
    }
    ?>
    </select>

</span>
<b> : </b>
<span>
<select name="minutooff" id="minutooff">
    <?php
    for ($i=0;$i<60;$i++)
    {
    if ($i == $row['minutooff'])
    echo " <option selected=selected value=$i >$i</option>";
    else
    echo " <option value=$i >$i</option>";
    }
    ?>
    </select>

```

```

</span>
</div>
</div>
</div>
<div style=" text-align:center">
<input style="margin-top:30px; " type="submit" id="guardar" value="Guardar" />
</div>
</form>
</body>
</html>

```

### **ESTADO DE LUCES.PHP //CAMBIO DE ESTADO (ENCENDIDO O APAGADO) CADA LUMINARIA EN EL PLANO DE CASA DIBUJANDOLO**

```

<?php
include("modulo.php");
$con= conectar();
extract($_GET);

$tabla=pg_query($con,"select items.* ,puertos.valor from items left join puertos on
items.idpuerto=puertos.idpuerto order by puertos.idpuerto ");

$c=0;

$puertos = array();
//$valores = array();
while($row=pg_fetch_array($tabla))
{
if($row['valor']=="f" || $row['valor']==null )
$color="#fff";
else
$color="#a00";
$puertos[$c]=array('item'.$row['iditem'],$color);
$c=$c+1;
}
$e = json_encode($puertos);
echo $e;

```

?>

**MODULO REPORTE.PHP //MODULO PARA GENERAR LOS REPORTE DE LOS  
EVENTOS GENERADOS DURANTE UN RANGO DETERMINADO**

<?php

session\_start();

include("modulo.php");

date\_default\_timezone\_set("America/Guayaquil") ;

\$hoy=date("d/m/Y H:m:s");

?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />

<title>DOMOTICA</title>

<script type="text/javascript">

<!--

function isNumberKey(e)

{

var key = window.Event ? e.which : e.keyCode

if ((key >= 47 && key <= 58) || key==8 || key==32 || key==13)

return true;

return false;

}

//-->

function busqueda()

{

txtdesde= document.getElementById("txtdesde").value;

txthasta= document.getElementById("txthasta").value;

if (window.XMLHttpRequest)

xmlhttp=new XMLHttpRequest();

```

xmlhttp.open("GET","consultar_resumen.php?txtdesde="+txtdesde+"&txthasta="+txthasta
);
xmlhttp.onreadystatechange=function()
{
    if (xmlhttp.readyState==4)
        document.getElementById("consumo").innerHTML= xmlhttp.responseText;
}
xmlhttp.send(null);
}
</script>
</head>
<body onload="busqueda();">
<div style="box-shadow: 0px 0px 20px #999;width:700px; position:relative;
margin:auto;border:#777 thin solid; background:#f5f5f5; height:auto; padding:20px;">
<h1>Reporte del Control de Luces </h1>
<div>Desde: <input placeholder="<?php echo $hoy;?>" type="text" name="txtdesde"
id="txtdesde" maxlength="19" onkeypress="return isNumberKey(event)"
onchange="busqueda();" /> Hasta : <input maxlength="19" placeholder="<?php echo
$hoy;?>" type="text" name="txthasta" id="txthasta" onkeypress="return
isNumberKey(event)" onchange="busqueda();" /></div><br>
<div id="consumo" style=" border:#ccc thin solid ; width:600px ; position:relative;margin-
bottom:10px;margin:auto">
<table border="1" width="100%" style="font-size:12px">
<tr bgcolor="#CCCCCC"><th >ID. PUERTO</th><th>DESCRIPCION</th><th>FECHA</th><th>ESTADO</th></tr>
</table>
</div>
</div>
</body>

```

## **PROGRAMACION DE LA APLICACIÓN EN VISUAL BASIC.NET**

### **MODULO1.VB**

Imports System.IO.Ports

Imports System.Net.Sockets

Imports System.Net

Imports System.IO

Module Module1

Public socket As Net.Sockets.TcpClient

Public dataStream As System.IO.StreamWriter

End Module

### **FORM1.VB**

Imports System.IO.Ports

Imports System.Net.Sockets

Imports System.Net

Imports System.IO

Public Class Form1

Public c As Integer

Public conn As New Npgsql.NpgsqlConnection

Public row As Npgsql.NpgsqlDataReader

Public comando As Npgsql.NpgsqlCommand = conn.CreateCommand()

Public intento As Integer

Private Sub Timer1\_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs)  
Handles Timer1.Tick

Dim port As String

Dim valor As String

Dim cadena As String

Dim sum As String

Dim check As Integer

Timer1.Enabled = False

sum = ""

valor = ""

port = ""

cadena = "!"

check = 0

comando.CommandText = "select \* from puertos order by idpuerto"

```

row = comando.ExecuteReader()
While row.Read()
    If row.GetInt32(row.GetOrdinal("idpuerto")) < 10 Then
        port = "0" & row.GetInt32(row.GetOrdinal("idpuerto")).ToString
    Else
        port = row.GetInt32(row.GetOrdinal("idpuerto")).ToString
    End If
    If row.GetBoolean(row.GetOrdinal("valor")).GetHashCode = 0 Then
        sum = port
    Else
        check = row.GetInt32(row.GetOrdinal("idpuerto")) + 1
        If check < 10 Then
            sum = "0" & check.ToString
        Else
            sum = check.ToString
        End If
    End If
    valor = row.GetBoolean(row.GetOrdinal("valor")).GetHashCode.ToString
    If row.GetBoolean(row.GetOrdinal("automatico")) Then
        If Date.Now.Hour = row.GetInt32(row.GetOrdinal("horaon")) And Date.Now.Minute
        = row.GetInt32(row.GetOrdinal("minutoon")) Then
            check = row.GetInt32(row.GetOrdinal("idpuerto"))+
            If check < 10 Then
                sum = "0" & check.ToString
            Else
                sum = check.ToString
            End If
            cadena = cadena & "1" & port & sum
        End If
        If Date.Now.Hour = row.GetInt32(row.GetOrdinal("horaoff")) And
        Date.Now.Minute = row.GetInt32(row.GetOrdinal("minutooff")) Then
            sum = port

```



```

        cadena = cadena & "0" & port & sum
    End If
Else
    cadena = cadena & valor & port & sum
End If
End While
Try
    Module1.dataStream.Write(cadena)
    Module1.dataStream.Flush()
    row.Close()
    comando.Dispose()
    Timer1.Enabled = True
    intento = 0
Catch ex As Exception
    Label2.Text = "Error de conexion con Arduino..."
    Timer1.Enabled = False
    Timer2.Enabled = True
    Module1.dataStream.Flush()
    Module1.dataStream.Close()
    Module1.socket.Close()
    conn.Close()
End Try
End Sub

Private Sub Form1_FormClosed(ByVal sender As Object, ByVal e As
System.Windows.Forms.FormClosedEventArgs) Handles Me.FormClosed
    conn.Close()
    Timer1.Enabled = False
End Sub

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Try
        If Button1.Text = "Conectar" Then

```

```

    intento = 0
    Label2.Text = "Conectando..."
    Try
        conn.ConnectionString = "Server=localhost;Database=" + txtdb.Text +
";Port=5432;User=" + txtusuario.Text + ";password=" + txtclave.Text
        conn.Open()
    Catch ex As Exception
        MsgBox("Error de conexion a Postgres , usuario o password incorrecto",
MsgBoxStyle.OkOnly, "Aviso")
        Label2.Text = ""
        Exit Sub
    End Try
    ' Module1.socket = New TcpClient(txthost.Text, Val(txtpuerto.Text))
    ' Module1.dataStream = New StreamWriter(Module1.socket.GetStream)
    " Button1.Text = "Desconectar"
    ' Label2.Text = "Conectado con Arduino!!"
    'Timer1.Enabled = True
    'Me.WindowState = FormWindowState.Minimized
Else
    Label2.Text = ""
    Timer1.Enabled = False
    Button1.Text = "Conectar"
    Module1.dataStream.Flush()
    Module1.dataStream.Close()
    Module1.socket.Close()
    conn.Close()
End If
Catch ex As Exception
    Label2.Text = "Error de conexion con Arduino...intentelo nuevamente"
    conn.Close()
    Exit Sub
End Try

```

```

        Label2.Text = "ok"
    End Sub

    Private Function numero(ByVal s As String) As Integer
        Select Case s
            Case "02" : Return 2
            Case "03" : Return 3
            Case "04" : Return 4
            Case "05" : Return 5
            Case "06" : Return 6
            Case "07" : Return 7
            Case "08" : Return 8
            Case "09" : Return 9
            Case "10" : Return 10
            Case "11" : Return 11
            Case "12" : Return 12
            Case "13" : Return 13
            Case Else : Return 0
        End Select
    End Function

    Private Sub Timer2_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Timer2.Tick
        Label2.Text = ""
        Button1.Text = "Conectar"
        Timer2.Enabled = False
    End Sub

    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
    End Sub
End Class

```

### **PROGRAMACION PARA EL ARDUINO UNO Y WIFITYNE SHINE**

```
#include<stdlib.h> //LIBRERIA PARA FUNCIONES PRINCIPALES
```

```
//CONFIGURACION SETEO DE PUERTOS
```

```

void setup()
{
  for(int i =2 ; i<14;i++)
  pinMode(i,OUTPUT); //DECLARA SALIDAS
  for(int i =2 ; i<14;i++)
  digitalWrite(i,1);    //APAGA LOS PUERTOS
  Serial.begin(9600);    // abre el puerto serie, establece la velocidad a 9600 bps
}

```

### **//BUCLE PURNICIPAL**

```

void loop()
{
  char letra = 0;
  String cadena = "";
  int valor=0;
  int sum=0;
  int port=0;
  int x =0;
  int f=0;
  delay(3000);
  int t=2000;
  Serial.print("$$$");
  delay(t);
  //Serial.print("set ip dhcp 1\r");
  Serial.print("set ip dhcp 0\r");
  delay(t);
  Serial.print("set ip a 192.168.1.50\r");
  delay(t);
  Serial.print("set ip gateway 192.168.1.1\r");
  delay(t);
  Serial.print("set ip netmask 255.255.255.192\r");
  delay(t);
}

```

```

Serial.print("set ip protocol 2\r");
delay(t);
Serial.print("set wlan join 0\r");
delay(t);
Serial.print("set wlan phrase 1201412648\r");
delay(t);
Serial.print("join CNT INTRIAGO \r"); //el nombre de la Wlan
delay(t);
Serial.print("exit\r");
delay(t);
delay(1000);
while(1)
{
    cadena="";
    port=-1;
    while( Serial.available() <= 0){}
    while(Serial.read()!='!'){
        delay(2);      //debe haber un delay entre cada lectura al buffer serial de arduino
        de lo contrario habrar erores de lectura
        for(int i=0;i<60;i++)
        {
            letra=Serial.read();
            cadena=cadena+letra;
            delay(2);
        }
        Serial.flush();
        // Serial.print(cadena);
        for(int i=0;i<60;i=i+5)
        {
            valor=mictoi(cadena.substring(i,i+1));
            port=mictoi(cadena.substring(i+1,i+3));

```

```

sum=mictoi(cadena.substring(i+3,i+5));
if (port>1 && port<=13 && (port + valor == sum))
{
    if(valor==0 or valor==1)
    {
        if(valor==1)
        valor=0;
        else
        valor=1;
        // if (port==8)
        //{
        // if (valor ==1)
        // x=1;
        //else
        //x=0
        //if (valor!=x)
// {
//     digitalWrite(port,1);
//     delay(500);
//     digitalWrite(port,0);
// }
// if (ant != valor)
// {
//     digitalWrite(port,1);
//     delay(500);
//     digitalWrite(port,0);
// }
//}
digitalWrite(port,valor);
}
}

```

```

        port=0;
        valor=0;
        sum=0;
    }
}
}

//-----Funciones -----
int mictoi(String s)
{
    int x=0;
    int nx=0;
    int t=s.length();
    int f=0;
    for(int i =0;i<t; i++)
    {
        switch (i)
        {
            case 0 : f=0;break;
            case 1 : f=10;break;
            // case 2 : f=100;break;
        }
        x=numero(s.substring(i,i+1));
        if (x<0)
            return -1;
        nx=(nx*f)+x;
    }
    return nx;
}

int numero(String s)
{

```

```
if ( s=="0") return 0;
if ( s=="1") return 1;
if ( s=="2") return 2;
if ( s=="3") return 3;
if ( s=="4") return 4;
    if ( s=="5") return 5;
if ( s=="6") return 6;
if ( s=="7") return 7;
if ( s=="8") return 8;
if ( s=="9") return 9;
return -1;
}
```



## BIBLIOGRAFÍA

- ¿Qué significa php? (14 de Noviembre de 2012). *http://www.masadelante.com*. Obtenido de <http://www.masadelante.com/faqs/php>
- academia android. (11 de Diciembre de 2014). *android stdio v1.0: características y comparativa con Eclipse*. Obtenido de academiaandroid: <http://academiaandroid.com/android-studio-v1-caracteristicas-comparativa-eclipse/>
- Actividad Integradora. (8 de Junio de 2015). *ANDROID STUDIO Y SUS PRINCIPALES CARACTERÍSTICAS*. Obtenido de micalculadoraenequipo.blogspot: [http://micalculadoraenequipo.blogspot.com/2015\\_06\\_01\\_archive.html](http://micalculadoraenequipo.blogspot.com/2015_06_01_archive.html)
- ÁLVAREZ, R. (22 de Agosto de 2014). *iOS vs Android: la eterna batalla por combatir la fragmentación*. Obtenido de applesfera: <http://www.applesfera.com/apple-1/ios-vs-android-la-eterna-batalla-por-combatir-la-fragmentacion>
- Andrade, P. (s.f.). *Domótica en la vivienda*. Obtenido de monografías: <http://www.monografias.com/trabajos93/domotica-vivienda/domotica-vivienda.shtml>
- Android OS. (2012). *Android OS*. Obtenido de androidos.readthedocs: <http://androidos.readthedocs.io/en/latest/data/historia/>
- Android Studio. (s.f.). *Conoce Android Studio*. Obtenido de developer.android: <https://developer.android.com/studio/intro/index.html?hl=es-419>
- Arcia, M. E. (s.f.). *About en español*. Obtenido de tabletas.about: <http://tabletas.about.com/od/Conoce-las-tabletas/fl/Que-es-Android.htm>
- Ardións, A. (01 de Febrero de 2016). *Android Studio vs Eclipse*. Obtenido de androidstudiofaqs: <http://androidstudiofaqs.com/conceptos/android-studio-vs-eclipse>
- crm-ulatina. (s.f.). *Step 3: ¿Qué partes componen el Arduino?* Obtenido de instructables: <http://www.instructables.com/id/Arduino-Tutorial-0-Introducci%C3%B3n/step3/Qu%C3%A9-partes-componen-el-Arduino/>
- David. (16 de Julio de 2010). *Visual Basic. NET*. Obtenido de es.slideshare.net: <http://es.slideshare.net/DavidPaLMundo/visual-basic-net-4772730>
- FotoNostra. (s.f.). *¿Qué es un Smartphone?* Obtenido de fotonostra: <http://www.fotonostra.com/digital/smartphones.htm>
- Francisco. (21 de Mayo de 2013). *Historia de Android: La Evolución a lo largo de sus versiones*. Obtenido de androidzone: <http://androidzone.org/2013/05/historia-de-android-la-evolucion-a-lo-largo-de-sus-versiones/>
- Fumás, E. (11 de Junio de 2014). *Apache HTTP Server: ¿Qué es, cómo funciona y para qué sirve?* Obtenido de ibrugor: <http://www.ibrugor.com/blog/apache-http-server-que-es-como-funciona-y-para-que-sirve/>
- Ibrugor. (2015). *¿Qué es Apache servidor Web?* Obtenido de sites.google.com: <https://sites.google.com/site/tecceilpiii/home/3---apache---servidor-web>
- Lenguajes de Programación. (s.f.). *Lenguajes de programación*. Obtenido de lenguajes-de-programacion: <http://www.lenguajes-de-programacion.com/lenguajes-de-programacion.shtml>
- Lledó, E. (Diciembre de 2012). *Diseño de un sistema de control domótico*. Obtenido de riunet.upv.es: <https://riunet.upv.es/bitstream/handle/10251/18228/Memoria.pdf?sequence=1>
- Mariael. (27 de Junio de 2015). *1.- ¿Qué es apache,mysql,php?* Obtenido de github.com: <https://github.com/MARIAEL/SERVIDOR-WEB/blob/master/README.md>

moraeduardo2003. (s.f.). *Las Memorias de un Computador*. Obtenido de monografias: <http://www.monografias.com/trabajos16/memorias/memorias.shtml>

obrero, y. (18 de febrero de 2013). *Sistemas inteligentes*. Obtenido de informatica-domotica-tecnologia.blogspot: <http://informatica-domotica-tecnologia.blogspot.com/2013/02/los-sistemas-inteligentes-pueden.html>

Pérez Váldez, D. (26 de Octubre de 2007). *¿Qué son las bases de datos?* Obtenido de maestrosdelweb: <http://www.maestrosdelweb.com/que-son-las-bases-de-datos/>

Reyes, K. (s.f.). *Domótica*. Obtenido de Monografias: <http://www.monografias.com/trabajos35/domotica/domotica.shtml>

Robelo, B. (4 de Septiembre de 2011). *Lenguaje SQL para PostgreSQL*. Obtenido de es.slideshare: <http://es.slideshare.net/brobelo/postgresql-9132474>

Roldayan. (20 de Septiembre de 2013). *Desarrollo de aplicaciones en Android (II). Herramientas SDK*. Obtenido de mibqyo: [http://www.mibqyyo.com/articulos/2013/09/20/desarrollo-de-aplicaciones-en-android-ii-herramientas-sdk/#/vanilla/discussion/embed/?vanilla\\_discussion\\_id=0](http://www.mibqyyo.com/articulos/2013/09/20/desarrollo-de-aplicaciones-en-android-ii-herramientas-sdk/#/vanilla/discussion/embed/?vanilla_discussion_id=0)

Sanz, D., Saucedo, M., & Torralbo, P. (s.f.). *Introducción a Android*. Obtenido de it-docs: <http://www.it-docs.net/ddata/18.pdf>

SomosLibres. (s.f.). *Arduino más alla del hardware libre*. Obtenido de somoslibres: <http://www.somoslibres.org/modules.php?name=News&file=article&sid=5581>

TinySine. (2013). *TyniSine WIFI Shield User Manual*. Obtenido de tinyosshop: <http://www.tinyosshop.com/datasheet/Tinysine%20WIFI%20shield.pdf>

Todo Tabletas Electrónicas. (s.f.). *Que es una Tableta electrónica*. Obtenido de todotabletaselectronicas.blogspot: <http://todotabletaselectronicas.blogspot.com/2012/08/que-es-una-tableta-electronica.html>

Wikipedia. (Noviembre de 2016). *Android*. Obtenido de es.wikipedia.org: <https://es.wikipedia.org/wiki/Android>