



UNIVERSIDAD DEL BÍO-BÍO

# Proyecto de Título: Software Smart Home implementado en una Raspberry Pi

Juan Andrés Morales Aquevedo  
Concepción, 1er Semestre 2016, Universidad del Bío-Bío.

## RESUMEN

El proyecto presentado en este informe, consiste en el diseño y construcción de un sistema de automatización de bajo costo del tipo “Smart Home”. El cual contiene una aplicación en Android como componente principal de control y una plataforma web que contiene otras opciones extras de control y configuración del sistema.

El sistema permite llevar a cabo tareas de control y monitoreo de alumbrado, temperatura y alarma desde la misma aplicación Android. El núcleo del sistema está alojado en la Raspberry que es una tarjeta integrada de bajo costo que opera en conjunto con sensores de temperatura, movimiento y luminosidad.

En el estudio de factibilidad que se realizó, se demuestra que técnicamente y económicamente este proyecto es viable para el ciudadano promedio, ya que no se requieren grandes conocimientos para usarlo ni grandes sumas de dinero para implementarlo debido a su bajo precio para adquirirlo y su bajo precio para mantenerlo.

# Índice General

<b>1</b>	<b>INTRODUCCION.....</b>	<b>5</b>
1.1	DESCRIPCIÓN DE LA EMPRESA.....	6
1.2	DATOS DEL CLIENTE.....	6
1.3	DESCRIPCIÓN DEL ÁREA DE ESTUDIO .....	6
1.4	DESCRIPCIÓN DE LA PROBLEMÁTICA.....	7
1.5	FACTIBILIDAD TÉCNICA.....	8
<b>2</b>	<b>DEFINICIÓN PROYECTO .....</b>	<b>9</b>
2.1	OBJETIVOS DEL PROYECTO.....	9
2.2	AMBIENTE DE INGENIERÍA DE SOFTWARE. ....	10
2.2.1	METODOLOGÍA DE DESARROLLO: .....	10
2.2.2	TÉCNICAS Y NOTACIONES:.....	10
2.2.3	ESTÁNDARES DE DOCUMENTACIÓN, PRODUCTO O PROCESO:.....	10
2.2.4	HERRAMIENTAS DE APOYO:.....	11
2.3	DEFINICIONES, SIGLAS Y ABREVIACIONES.....	11
<b>3</b>	<b>ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE.....</b>	<b>12</b>
3.1	ALCANCES .....	12
3.2	OBJETIVO DEL SOFTWARE.....	13
3.3	DESCRIPCIÓN GLOBAL DEL PRODUCTO.....	14
3.3.1	INTERFAZ DE USUARIO .....	14
3.3.2	INTERFAZ DE HARDWARE.....	14
3.3.3	INTERFAZ SOFTWARE .....	14
3.3.4	REQUERIMIENTOS/RESTRICCIONES DE INTERFAZ DE COMUNICACIÓN .....	15
3.3.5	CONSIDERACIONES AMBIENTALES .....	15
3.4	REQUERIMIENTOS ESPECÍFICOS .....	16
3.4.1	REQUERIMIENTOS FUNCIONALES DEL SISTEMA .....	16
3.4.2	ATRIBUTOS DEL PRODUCTO .....	28
<b>4</b>	<b>PLANIFICACIÓN.....</b>	<b>29</b>
4.1	PROCESO DE NEGOCIO - FUTURO .....	30
4.2	MODELO DE CASOS DE USO .....	31
4.2.1	DIAGRAMA DE CASOS DE USO .....	31
4.2.2	ESPECIFICACIÓN DE LOS CASOS DE USO.....	33
<b>5</b>	<b>DISEÑO.....</b>	<b>52</b>
5.1	DISEÑO DE FÍSICO DE LA BASE DE DATOS .....	52
5.2	DISEÑO INTERFAZ Y NAVEGACIÓN.....	53
<b>6</b>	<b>ESPECIFICACIÓN DE CÓDIGO.....</b>	<b>61</b>
<b>7</b>	<b>RESUMEN ESFUERZO REQUERIDO .....</b>	<b>62</b>
<b>8</b>	<b>CONCLUSIONES.....</b>	<b>63</b>
<b>9</b>	<b>BIBLIOGRAFÍA .....</b>	<b>64</b>

## Índice de Figuras

<b><u>Figura 1: Esquema general del sistema .....</u></b>	<b><u>11</u></b>
<b><u>Figura 2. Carta Gantt del proyecto .....</u></b>	<b><u>28</u></b>
<b><u>Figura 3: Proceso de Negocios Futuros .....</u></b>	<b><u>29</u></b>
<b><u>Figura 4: Diagrama de casos de uso del proyecto .....</u></b>	<b><u>30</u></b>
<b><u>Figura 5. Diseño físico de la base de datos .....</u></b>	<b><u>52</u></b>
<b><u>Figura 5.1 Diseño interfaz y navegación .....</u></b>	<b><u>53</u></b>
<b><u>Figura 5.2 Menú de navegación web .....</u></b>	<b><u>53</u></b>
<b><u>Figura 5.3 Interfaz gráfica Android 1 .....</u></b>	<b><u>54</u></b>
<b><u>Figura 5.4 Interfaz gráfica Android 2 .....</u></b>	<b><u>54</u></b>
<b><u>Figura 5.5 Interfaz gráfica Android 3 .....</u></b>	<b><u>55</u></b>
<b><u>Figura 5.6 Interfaz gráfica Android 3 .....</u></b>	<b><u>55</u></b>
<b><u>Figura 5.7 Interfaz gráfica web 1 .....</u></b>	<b><u>56</u></b>
<b><u>Figura 5.8 Interfaz gráfica web 2 .....</u></b>	<b><u>57</u></b>
<b><u>Figura 5.9 Interfaz gráfica web 3 .....</u></b>	<b><u>57</u></b>
<b><u>Figura 5.10 Interfaz gráfica web 4 .....</u></b>	<b><u>58</u></b>
<b><u>Figura 5.11 Interfaz gráfica web 5 .....</u></b>	<b><u>58</u></b>
<b><u>Figura 6. Modelo Arquitectónico .....</u></b>	<b><u>59</u></b>
<b><u>Figura 7. Esquema de lenguajes utilizados .....</u></b>	<b><u>59</u></b>

---

## 1 INTRODUCCION

---

En la actualidad existen muchos sistemas de automatización y control de inmuebles, mejor conocidos como sistemas “Smart Home”, u hogar inteligente en español, que son capaces de controlar la mayoría de los artefactos de una casa u oficina, De estos, hay variados tipos y tamaños, y si bien tienen más bondades que desventajas, la mayoría de estos se caracterizan negativamente por una misma cosa; su precio; el cual para la mayoría de los ciudadanos promedios es un poco alto, y es por esto que estos sistemas inteligentes quedan inalcanzables para la gran mayoría de la población.

El actual proyecto presentado aquí, se basa en la creación de un sistema Smart Home implementado en una Raspberry, todo esto con un enfoque de disminución en los costos para que el usuario final no tenga que incurrir en grandes gastos monetarios y esta sea una alternativa viable para cualquier persona.

La motivación de este proyecto nace de la necesidad de crear una alternativa más económica y alcanzable que las que no ofrecen las tiendas de electrónica y la ventaja de usar la nueva versión de la minicomputadora Raspberry que ahora incorpora tecnología inalámbrica por el mismo precio que las versiones anteriores.

El sistema que se desarrolló es capaz de controlar y monitorear, el alumbrado, a temperatura y la alarma de un inmueble, todo, a través de un sistema implementado en este monoprocesador, con esto se realizarán las mismas funciones que hace un sistema Smart Home empaquetado, e incluso algunas más, ya que al haber usado una Raspberry, se tuvo la ventaja de que se pudieron incluir unas cuantas funcionalidades extras.

## 1.1 Descripción de la empresa

### 1.2 Datos del Cliente

Nombre: Universidad del Biobío  
Dirección: Collao #1202, Concepción  
Teléfono: 3111200

### Descripción de la Empresa

Nombre: Universidad del Biobío  
Rubro: Universidad  
Servicio: Servicios Académicos

## 1.3 Descripción del área de estudio

El área de estudio corresponde al sector de domótica a nivel de un inmueble.

Objetivos del área: Automatizar un inmueble, aportando servicios de gestión energética, seguridad, y comunicación, que pueden estar integrados por medio de redes interiores y exteriores de comunicación, cableadas o inalámbricas. Se puede definir esta área como la integración de la tecnología en el diseño inteligente de un recinto cerrado.

## 1.4 Descripción de la problemática

En la actualidad la implementación de un sistema de automatización es algo tremendamente costoso, los artefactos que venden como un kit para realizar alguna tarea en específico son eficientes, sin embargo, muchas veces superan el presupuesto de un ciudadano promedio.

Como por ejemplo el kit Smart Home marca Ozom que vende la tienda de Retail Sodimac en Chile, cuyo precio inicial es de \$329,000, (kit Básico)

<http://www.sodimac.cl/sodimac-cl/product/2698382/Kit-Smart-Casa>

Este kit puede controlar las luces, la temperatura, y la alarma del inmueble.

En este proyecto el costo que se debe llevar a cabo es mucho más bajo que comprar uno en las tiendas de Retail, alrededor de \$102000 pesos para que funcione este mismo kit básico de Smart Home. Este desglose de costos se especifica más abajo, en la sección de factibilidad técnica.

Por eso uno de los pilares fundamentales de este proyecto es el bajo costo, que este trae consigo, además de tener ciertas funcionalidades que estos otros kits no traen, como por ejemplo un módulo de emulación de luces y un visualizador de registro de datos, las cuales se especifican más adelante.

## 1.5 Factibilidad Técnica.

El objetivo de esta factibilidad es evaluar los equipos tecnológicos que ya existen en la organización y ver la posibilidad de reutilizar dichos equipos para implementar el sistema.

En este caso, no hay ningún equipo existente, por lo que se debe incurrir el gasto de ellos, el desglose de los valores se especifica más abajo

En esta evaluación se usarán dos enfoques: Hardware y Software.

En cuanto a Hardware, específicamente el mini pc donde debe estar instalado el sistema propuesto, este debe cubrir con los siguientes requerimientos mínimos:

- Raspberry PI AMD Quad Core 1.2 Ghz. 1 GB de Memoria RAM
- Sensor de temperatura DHT11
- Sensor de movimiento HCSR501
- Sensor Laser KY-008

Raspberry PI:	23000 pesos (25 USD)
Sensor de temperatura DHT11:	2000 pesos
Sensor de movimiento HC--SR501	2500 pesos
Sensor Laser KY-008	1500 pesos
Costo Total:	29000 pesos
Instalación	60000 pesos
Total	102000 pesos



---

## 2 DEFINICIÓN PROYECTO

---

### 2.1 Objetivos del proyecto

**Objetivo General:** Desarrollar una aplicación que opere una Raspberry, y que permita gestionar el alumbrado, la temperatura y la alarma de un determinado inmueble.

**Objetivos Específicos:**

- Investigar sobre los lenguajes de programación que usaré en este proyecto, los cuales serán Python, Java y PHP.
- Estudiar el uso de la Raspberry en el campo de la domótica.
- Investigar sobre el uso e implementación de diferentes tipos de sensores para domótica.
- Hacer que la aplicación visualmente sea lo más simple posible y fácil de usar para las personas
- Crear en el sistema un registro de usuarios para que tengan acceso a la alarma.
- Crear en el sistema una sección para visualizar los registros de datos que almacena la Raspberry
- Hacer que el proyecto sea lo más económico y eficiente posible
- Crear un modelo a escala para la demostración final del proyecto

## **2.2 Ambiente de Ingeniería de Software.**

### **2.2.1 Metodología de Desarrollo:**

El proyecto será desarrollado utilizando el método de desarrollo iterativo-incremental y están planificadas solo 2 iteraciones. El corto tiempo se tiene para desarrollar el sistema (60 días en este caso) obliga a no tener más iteraciones.

### **2.2.2 Técnicas y notaciones:**

Para el desarrollo del sistema se utilizará lo siguiente:

- Para la toma de requerimientos se realizan entrevistas con el cliente que necesita el sistema.

Para ayudar en el desarrollo del sistema se utiliza:

- Diagrama de casos de uso (UML): Utilizada para representar la forma en como un usuario (Actor) interactúa con el sistema.
- Modelo Relacional (MR): Utilizado para modelar problemas reales y administrar los datos dinámicamente.
- BPMN (Business Process Model and Notation): Utilizada para proporcionar una notación estándar que sea fácilmente entendible por todos los involucrados e interesados en el negocio.

### **2.2.3 Estándares de documentación, producto o proceso:**

- Estándar para el informe: Plantilla “Planilla de documentación del proyecto de Desarrollo de Software” versión 15 de noviembre de 2011.
- Estándar para los requerimientos funcionales: Adaptación basada en IEEE “Software requirements Specifications” Std 830-1998.
- Estándar para la evaluación del producto software: ISO/IEC 9126
- IEEE Software Test Documentation Std 829 -1998

#### **2.2.4 Herramientas de apoyo:**

- Editor de texto Sublime Text 3.
- Power Designer 16.1.0.3637.
- Putty
- Andriod Studio
- Geany
- Libre Ofiice Writer
- PoweGannt

#### **2.3 Definiciones, Siglas y Abreviaciones**

- Usuario: persona capaz de utilizar el sistema y modificar sus parámetros
- Boolean: tipo de datos que toma el valor de 0 o 1 (verdadero o falso).
- Sistema: conjunto de partes interrelacionadas, hardware, software y recursos humanos que permiten procesar y almacenar información.
- Casos de uso: Modelo utilizado para definir los pasos a seguir para realizar un proceso.
- BPMN: Bussines Process Model Notation
- Raspberry Pi: Mini Computadora que almacena gran parte del Sistema
- Andriod app: Aplicación Movil desarrollada para el sistema.
- Sensor DTH11: Sensor de temperatura que utiliza el sistema
- Sensor HC--SR501: Sensor de Movimiento que utiliza el sistema
- LDR: Foto-resistor electrónico que utiliza el sistema
- Laser KY-008: Laser que se utiliza en conjunto con el LDR en el sistema.

### 3 ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE

Adaptación basada en *IEEE Software requirements Specifications Std 830-1998*.

#### 3.1 Alcances

Se implementará una aplicación en Android que pueda controlar el alumbrado, la calefacción y la alarma de un inmueble. El usuario tendrá la opción de controlar el sistema a través de la app de Android o la opción de una plataforma web que será montada en la misma Raspberry.

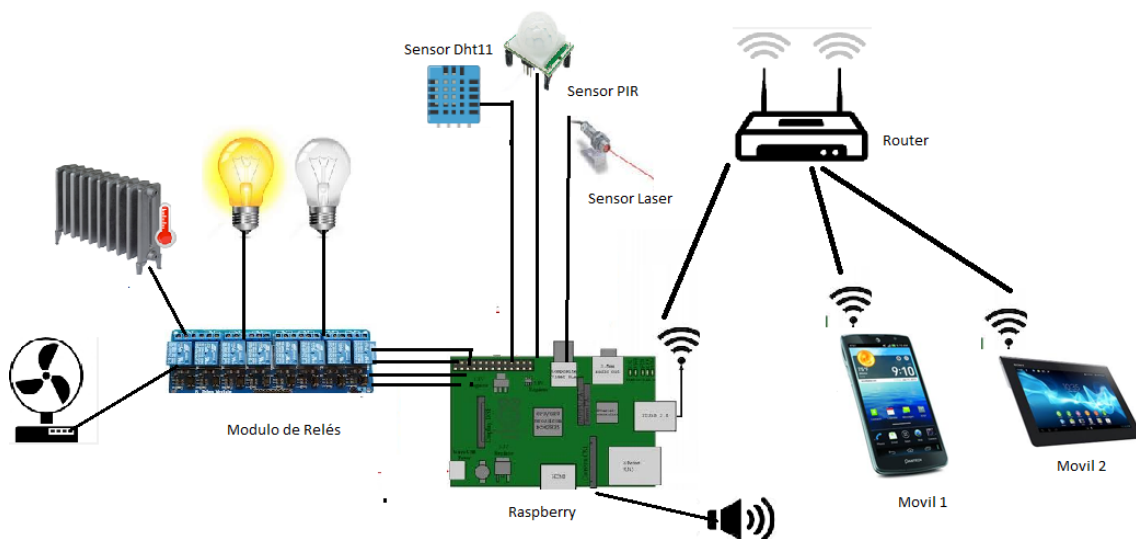


Figura 1: Esquema general del sistema

Dentro de lo que se puede hacer, se pueden apagar y prender las luces, o cualquier artefacto eléctrico conectado a una salida de la maquina (Raspberry), de manera manual o automática, este a su vez tendrá un registro de las operaciones que se hagan con el alumbrado, para crear paralelamente un registro del uso de este, y que finalmente puede ser utilizado por el usuario para crear una simulación de su uso, con el fin de emular que hay individuos en el inmueble.

También se puede gestionar la temperatura que desee el usuario, con esto se activará algún artefacto para llegar al punto ideal, puede ser una calefacción o algún aire acondicionado. Esto también se podrá hacer de manera automática o manual.

Y, por último, el sistema contará también con un módulo de alarma, que puede ser gatillado por cualquier sensor del tipo Trigger True/False, pero en este caso usará un sensor laser (Tripwire), el cual se puede activar y desactivar desde la App Android insertando una contraseña o desde la plataforma web.

Adicionalmente en la plataforma web, existirá la posibilidad de visualizar los registros de cuándo y a qué horas se encendieron luces o artefactos, las temperaturas registradas en cierto periodo, o cuando se activó o desactivó la alarma. Todo esto se puede visualizar en la plataforma web mediante gráficos.

### **3.2 Objetivo del software**

El sistema ayuda principalmente a gestionar las tareas básicas que se deben hacer para controlar un inmueble, está principalmente apuntado a potenciar el confort y la comodidad de hacer mínimos esfuerzos físicos para realizar dichas tareas.

Objetivos Específicos del Software:

- El sistema debe permitir al usuario controlar el alumbrado del lugar
- El sistema debe permitir al usuario iniciar una simulación de alumbrado
- El sistema debe permitir al usuario ver los registros de que se almacenan.
- El sistema debe permitir al usuario configurar los parámetros para que su app se conecte con la Raspberry
- El sistema debe permitir al usuario definir una temperatura ideal del inmueble
- El sistema debe permitir al usuario poder controlar la temperatura del inmueble
- El sistema debe permitir al usuario activar o desactivar la alarma del inmueble
- El sistema debe enviar notificaciones de audio cuando ocurran estos eventos.
- El sistema debe permitir al usuario poder controlar todo lo anterior desde la app de Android o desde la interfaz web

### **3.3 Descripción Global del Producto**

#### **3.3.1 Interfaz de usuario**

El formato general se basa en que la aplicación para Android cuenta con una pantalla principal que contiene 3 Sliders para visualizar las 3 áreas que controla el sistema, en la esquina de la app estará un botón que llevará a la configuración del sistema y otros datos de información menores.

El formato para la interfaz web es similar al de la aplicación, con un sistema de 3 ventanas y con los mismos botones e información, y a su vez será responsivo para dispositivos móviles, este es más básico que el otro, ya que es una interfaz secundaria que posee el sistema.

#### **3.3.2 Interfaz De Hardware**

El sistema interactúa con varios sensores conectados a la RaspBerry, estos salen desde el puerto GPIO que posee la máquina.

- Sensor DTH11: Sensor de temperatura, GPIO.18
- Sensor HC-SR501: Sensor de Movimiento, GPIO.15
- LDR: Foto-resistor electrónico que utiliza el sistema, GPIO.05
- Laser KY-008: Laser que apunta al LDR, GPIO.14
- 

#### **3.3.3 Interfaz Software**

El sistema es un sistema independiente, por lo cual no tendrá comunicación con otros sistemas de software externos excluyentes al sistema operativo y navegador.

### **3.3.4 Requerimientos/Restricciones de Interfaz de comunicación**

Se utilizará el protocolo HTTP para la comunicación de la Raspberry con la aplicación de Android, y el protocolo TCP/IP para el transporte de los paquetes de datos correspondientes.

### **3.3.5 Consideraciones ambientales**

El sistema será utilizado por el usuario que posea la aplicación en su móvil y tenga acceso a la red wifi del inmueble

El hardware del equipo (Raspberry) características:

- AMD Quad Core 1.2 Ghz.
- 1 GB de Memoria RAM 1
- WiFi 802.11n integrado
- Entrada Micro SD
- Bluetooth 4.1

El software del equipo (RaspBerry) donde será instalado el software tendrá las siguientes características:

- PhP 5.0
- Linux Raspbian Jessie
- Servicio Web Apache 2.2
- Python 2.7.12
- Librerías AdaFruit para Python

El equipo móvil donde será instalada la aplicación deberá cumplir con las siguientes características:

- Poseer como mínimo Android versión 2.3
- 3 MB de memoria libre en el móvil.

### 3.4 Requerimientos Específicos

#### 3.4.1 Requerimientos Funcionales del sistema

Id	Nombre	Descripción
RF_01	Gestionar Alumbrado y aparatos del sistema	El sistema permitirá al usuario encender, apagar y ver el estado (on/off) de un determinado aparato luminoso
RF_02	Gestionar Temperatura del sistema	El sistema permitirá al usuario poder establecer la temperatura ideal para el inmueble o bien, finalizar esta tarea
RF_03	Gestionar Alarma del sistema	El sistema permitirá al usuario activar o desactivar la alarma del inmueble.
RF_04	Gestionar el modo de simulación	El sistema permitirá al usuario seleccionar un modo de simulación de alumbrado que desee
RF_05	Gestionar Configuración del sistema	El sistema permite al usuario entrar a la configuración para establecer los parámetros de red
RF_06	Gestionar registros.	El sistema permite al usuario poder visualizar los datos que el sistema guardó internamente.
RF_07	Gestionar Contraseñas	El sistema permite al usuario crear o modificar o eliminar una contraseña por la alarma



## Especificación de Requerimientos

Especificación de Requerimientos			
Código		Nombre	
ER_01.1		Prender luz/aparato	
Descripción	El sistema debe prender/apagar una luz o aparato		
Entradas	Salida	Destino	Restricciones
Datos del usuario:  Botón Encendido de luz	Luz física encendida y el botón de la app muestra el estado de apagado	Tarea de encendido procesada en la Raspberry	No hay
Proceso	El usuario debe abrir la aplicación de su móvil y situarse en la pantalla principal o área de iluminación, debe presionar el botón de encendido deseado.		

Especificación de Requerimientos			
Código		Nombre	
ER_01.2		Apagar luz/aparato	
Descripción	El sistema debe apagar una luz o aparato		
Entradas	Salida	Destino	Restricciones
Datos del usuario:  Botón apagado de luz	Luz física apagada y el botón de la app muestra el estado de apagado	Tarea de apagado procesada en la Raspberry	No hay
Proceso	El usuario debe abrir la aplicación de su móvil y situarse en la pantalla principal o área de iluminación, debe presionar el botón de apagado deseado.		

Especificación de Requerimientos			
Código		Nombre	
ER_01.3		Encender modo automático de luces	
Descripción	El sistema debe prender/apagar una luz o aparato automáticamente		
Entradas	Salida	Destino	Restricciones
Datos del usuario:  Botón Encendido automático de luz	Luces físicas encendidas y el botón de la app muestra el estado de encendido	Tarea de encendido automático procesada en la Raspberry	No hay
Proceso	El usuario debe abrir la aplicación de su móvil o vía web y situarse en la pantalla principal o área de iluminación, debe presionar el botón de encendido automático de luces		

Especificación de Requerimientos			
Código		Nombre	
ER_01.4		Apagar modo automático de luces	
Descripción	El sistema debe terminar la tarea de modo automático		
Entradas	Salida	Destino	Restricciones
Datos del usuario:  Botón apagado de Encendido automático de luces	Luz física apagada y el botón de la app muestra el estado de apagado	Tarea de apagado procesada en la Raspberry	No hay
Proceso	El usuario debe abrir la aplicación de su móvil o vía web y situarse en la pantalla principal o área de iluminación, debe presionar el botón de apagado automático de luces		

Especificación de Requerimientos			
Código		Nombre	
ER_02.1		Establecer temperatura automática	
Descripción	El sistema debe regular la temperatura del inmueble		
Entradas	Salida	Destino	Restricciones
Datos del usuario:  Temperatura (en grados celcius)	El sistema empezará a regular temperatura; activando aparatos para calentar o enfriar, según corresponda	Tarea para regular temperatura será procesada en la Raspberry	La temperatura no puede superar los 40 grados
Proceso	El usuario debe abrir la aplicación de su móvil y situarse en la pantalla de área de ambiente, debe seleccionar la temperatura deseada y presionar el botón de encendido		

Especificación de Requerimientos			
Código		Nombre	
ER_02.2		Apagar temperatura automática	
Descripción	El sistema debe apagar el modo de temperatura automática del inmueble		
Entradas	Salida	Destino	Restricciones
Botón de Finalizar Tarea de temperatura automática	El sistema terminará de regular temperatura; desactivando aparatos para calentar o enfriar, según corresponda	Tarea para terminar de regular temperatura será procesada en la Raspberry	Tiene que estar en ejecución la regulación automática
Proceso	El usuario debe abrir la aplicación de su móvil y situarse en la pantalla de la sección de ambiente, y presionar el botón de apagado de regulación automática		

Especificación de Requerimientos			
Código		Nombre	
ER_02.3		Encender Aumentar temperatura	
Descripción	El sistema debe subir la temperatura del inmueble		
Entradas	Salida	Destino	Restricciones
Datos del usuario:  Temperatura (en grados Celsius)	El sistema empezará a subir temperatura; activando aparatos para calentar, según corresponda	Tarea para regular temperatura será procesada en la Raspberry	La temperatura no puede superar los 40 grados
Proceso	El usuario debe abrir la aplicación de su móvil o vía web y situarse en la pantalla de área de ambiente, debe presionar el botón de “calentar”.		

Especificación de Requerimientos			
Código		Nombre	
ER_02.4		Encender Bajar temperatura	
Descripción	El sistema debe bajar la temperatura del inmueble		
Entradas	Salida	Destino	Restricciones
Datos del usuario:  Temperatura (en grados celcius)	El sistema empezará a bajar temperatura; activando aparatos para enfriar, según corresponda	Tarea para bajar temperatura será procesada en la Raspberry	No hay
Proceso	El usuario debe abrir la aplicación de su móvil o vía web y situarse en la pantalla de área de ambiente, debe presionar el botón de “enfriar”.		

Especificación de Requerimientos			
Código		Nombre	
ER_02.5		Apagar Aumentar temperatura	
Descripción	El sistema debe subir la temperatura del inmueble		
Entradas	Salida	Destino	Restricciones
Datos del usuario:  Temperatura (en grados Celsius)	El sistema desactivará el aparato para subir la temperatura inmediatamente.	Tarea para regular temperatura será finalizada en la Raspberry	La temperatura no puede superar los 40 grados
Proceso	El usuario debe abrir la aplicación de su móvil o vía web y situarse en la pantalla de área de ambiente, debe presionar el botón de “parar”.		

Especificación de Requerimientos			
Código		Nombre	
ER_02.6		Apagar Bajar temperatura	
Descripción	El sistema debe bajar la temperatura del inmueble		
Entradas	Salida	Destino	Restricciones
Datos del usuario:  Temperatura (en grados celcius)	El sistema desactivará el aparato para bajar la temperatura inmediatamente.	Tarea para bajar temperatura será finalizada en la Raspberry	No hay
Proceso	El usuario debe abrir la aplicación de su móvil o vía web y situarse en la pantalla de área de ambiente, debe presionar el botón de “parar”.		

Especificación de Requerimientos			
Código		Nombre	
ER_02.4		Bajar temperatura	
Descripción	El sistema debe bajar la temperatura del inmueble		
Entradas	Salida	Destino	Restricciones
Datos del usuario:  Temperatura (en grados celcius)	El sistema empezará a bajar temperatura; activando aparatos para enfriar, según corresponda	Tarea para bajar temperatura será procesada en la Raspberry	No hay
Proceso	El usuario debe abrir la aplicación de su móvil o vía web y situarse en la pantalla de área de ambiente, debe presionar el botón de “enfriar”.		

Especificación de Requerimientos			
Código		Nombre	
ER_03.1		Activar la alarma	
Descripción	El sistema debe activar la alarma del sistema		
Entradas	Salida	Destino	Restricciones
Datos del usuario:  Botón Encendido de alarma	La alarma es activada y una notificación sonora es emitida	Tarea de encendido de alarma procesada en la Raspberry	La alarma tiene que estar apagada
Proceso	El usuario debe abrir la aplicación de su móvil y situarse en la pantalla de seguridad, y apretar el botón táctil de encendido de alarma		

Especificación de Requerimientos			
Código		Nombre	
ER_03.2		Desactivar la alarma	
Descripción	El sistema debe desactivar la alarma del sistema		
Entradas	Salida	Destino	Restricciones
Datos del usuario:  Botón Apagado de alarma	La alarma es desactivada y una notificación sonora es emitida	Tarea de apagado de alarma procesada en la Raspberry	La alarma tiene que estar activada
Proceso	El usuario debe abrir la aplicación de su móvil y situarse en la pantalla de seguridad, y apretar el botón táctil de desactivación de alarma		

Especificación de Requerimientos			
Código		Nombre	
ER_04.1		Activar Simulación Simple	
Descripción	El sistema debe activar la simulación de alumbrado simple		
Entradas	Salida	Destino	Restricciones
Datos del usuario:  Botón de simulación simple	La simulación simple es activada de manera indefinida.	Tarea de simulación alumbrado es procesada en la Raspberry	No hay
Proceso	El usuario debe abrir la aplicación de su móvil y situarse en la pantalla de seguridad, y apretar el botón táctil de activación de simulación simple		

Especificación de Requerimientos			
Código		Nombre	
ER_04.2		Desactivar Simulación Simple	
Descripción	El sistema debe desactivar la simulación de alumbrado simple		
Entradas	Salida	Destino	Restricciones
Datos del usuario:  Botón de desactivación de simulación simple	La simulación simple es desactivada de manera indefinida.	Tarea de simulación alumbrado es cancelada en la Raspberry	La simulación simple tiene que estar activada
Proceso	El usuario debe abrir la aplicación de su móvil y situarse en la pantalla de seguridad, y apretar el botón táctil de desactivación de simulación simple		

Especificación de Requerimientos			
Código		Nombre	
ER_04.3		Activar Simulación Avanzada	
Descripción	El sistema debe activar la simulación de alumbrado avanzada		
Entradas	Salida	Destino	Restricciones
Datos del usuario:  Botón de simulación avanzada	La simulación avanzada es activada de manera indefinida.	Tarea de simulación avanzada de alumbrado es procesada en la Raspberry	No hay
Proceso	El usuario debe abrir la aplicación de su móvil y situarse en la pantalla de seguridad, y apretar el botón táctil de activación de simulación avanzada		



Especificación de Requerimientos			
Código		Nombre	
ER_04.4		Desactivar Simulación Avanzada	
Descripción	El sistema debe desactivar la simulación de alumbrado avanzada		
Entradas	Salida	Destino	Restricciones
Datos del usuario:  Botón de desactivación de simulación avanzada	La simulación avanzada es desactivada de manera indefinida.	Tarea de simulación avanzada de alumbrado es cancelada en la Raspberry	La simulación avanzada tiene que estar activada
Proceso	El usuario debe abrir la aplicación de su móvil y situarse en la pantalla de seguridad, y apretar el botón táctil de desactivación de simulación avanzada		

Especificación de Requerimientos			
Código		Nombre	
ER_05.1		Editar configuración	
Descripción	El sistema debe configurar la app		
Entradas	Salida	Destino	Restricciones
Datos del usuario:  IP de la Raspberry	La app se tendrá comunicación con la Raspberry vía Wi-fi	Configuración guardada en el móvil donde está la app de Android	La IP debe ser valida
Proceso	El usuario debe abrir la aplicación de su móvil y situarse en la pantalla de configuración y establecer la IP de la Raspberry		

Especificación de Requerimientos			
Código		Nombre	
ER_06.1		Ver datos de alumbrado y temperatura	
Descripción	El sistema debe configurar la app		
Entradas	Salida	Destino	Restricciones
Datos del usuario:  Ninguno	La interfaz web mostrará los todos el uso de alumbrado y su temperatura de un dia o varios días	Salida por pantalla de los datos, mediante una gráfica.	Tienen que ser días validos
Proceso	El usuario debe abrir la interfaz web y seleccionar la opción para ver datos y seleccionar un día.		

Especificación de Requerimientos			
Código		Nombre	
ER_07.1		Cambiar contraseña	
Descripción	El sistema debe cambiar la contraseña de un usuario		
Entradas	Salida	Destino	Restricciones
Datos del usuario:  Nombre usuario Contraseña actual Contraseña nueva	La nueva contraseña será guardada en el sistema y la antigua se borrará de la memoria	Base de datos del sistema.	Contraseña debe ser numérica.
Proceso	El usuario debe abrir la plataforma web y situarse en la sección de “cambio de contraseña” y rellenar los 3 campos que ahí se piden, para después apretar el botón aceptar.		

Especificación de Requerimientos			
Código		Nombre	
ER_07.2		Crear Usuario nuevo	
Descripción	El sistema debe crear un nuevo usuario y su contraseña correspondiente		
Entradas	Salida	Destino	Restricciones
Datos del usuario:  Nombre usuario Contraseña admin Contraseña usuario	El usuario nuevo y su contraseña será creados y almacenados en la memoria	Base de datos del sistema.	Contraseña debe ser numérica.
Proceso	El usuario debe abrir la plataforma web y situarse en la sección de “crear usuario” y rellenar los 3 campos que ahí se piden, para después apretar el botón aceptar.		

Especificación de Requerimientos			
Código		Nombre	
ER_07.3		Eliminar Usuario	
Descripción	El sistema debe eliminar un u usuario y su contraseña correspondiente		
Entradas	Salida	Destino	Restricciones
Datos del usuario:  Nombre usuario Contraseña admin	El usuario nuevo y su contraseña será creados y almacenados en la memoria	Base de datos del sistema.	Contraseña debe ser numérica.
Proceso	El usuario debe abrir la plataforma web y situarse en la sección de “eliminar usuario” y seleccionar el usuario a eliminar, introducir la contraseña de admin y aceptar		

### 3.4.2 Atributos del producto

Tomando como referencia el modelo ISO/IEC 9126), se indican los atributos de calidad del producto software que se desarrolla.

- **USABILIDAD- OPERABILIDAD.**

La aplicación es muy intuitiva y fácil de usar, tiene una interfaz simple con opciones fáciles de interpretar y ejecutar.

- **EFICIENCIA- TIEMPO DE EJECUCIÓN/RESPUESTA.**

El tiempo de respuesta es instantáneo, no hay retraso en la respuesta del sistema, si bien el procesamiento interno de datos quizás tome algunos segundos, esto es imperceptible a la vista del usuario.

- **FUNCIONALIDAD-SEGURIDAD.**

- El sistema se limita a la red interna del inmueble, (a menos que esta sea una red abierta, se recomienda tener habilitada una contraseña de la red ), por lo que sin este acceso ningún otro agente externo puede intervenir, además se debe poseer la aplicación instalada en el móvil.

## 4 PLANIFICACIÓN

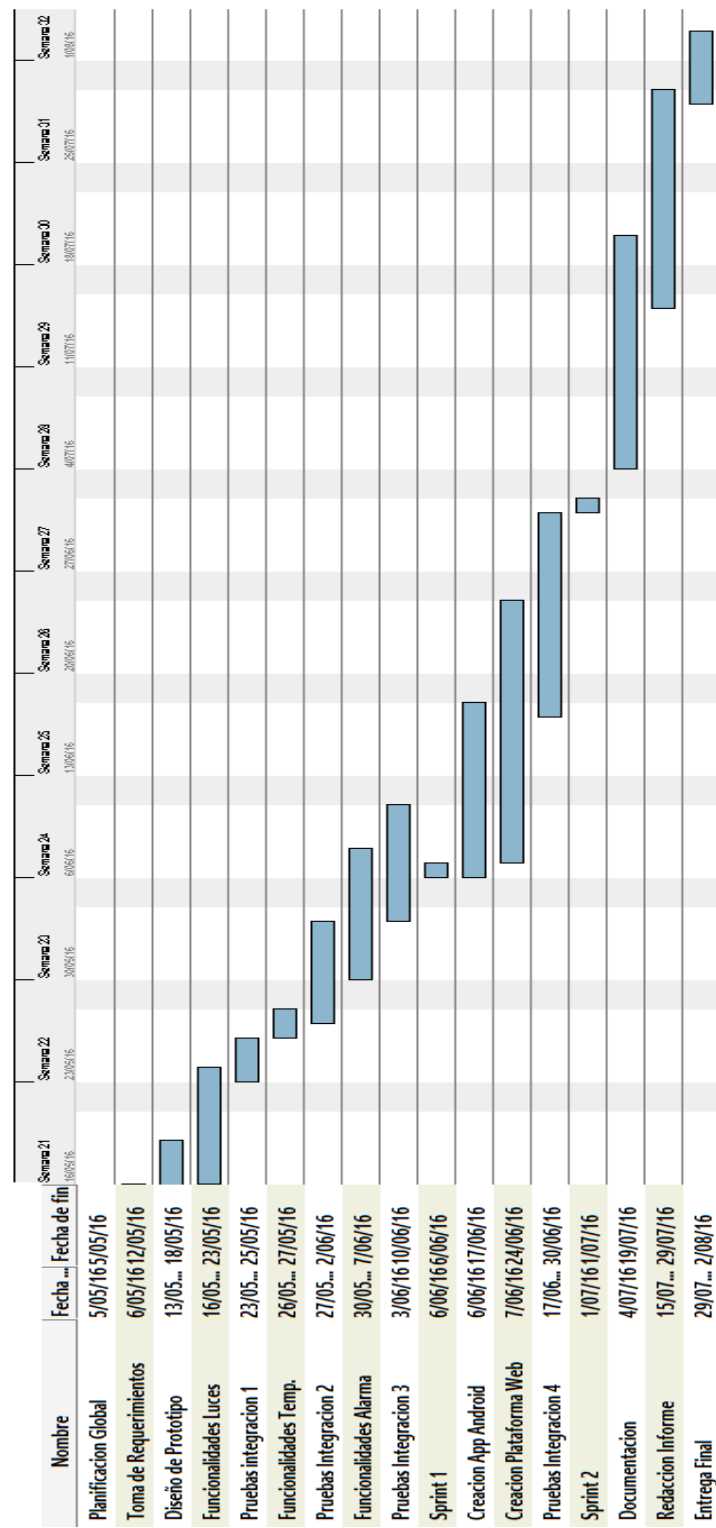


Figura 2. Carta Gantt del proyecto

## 4.1 Proceso de negocio - Futuro

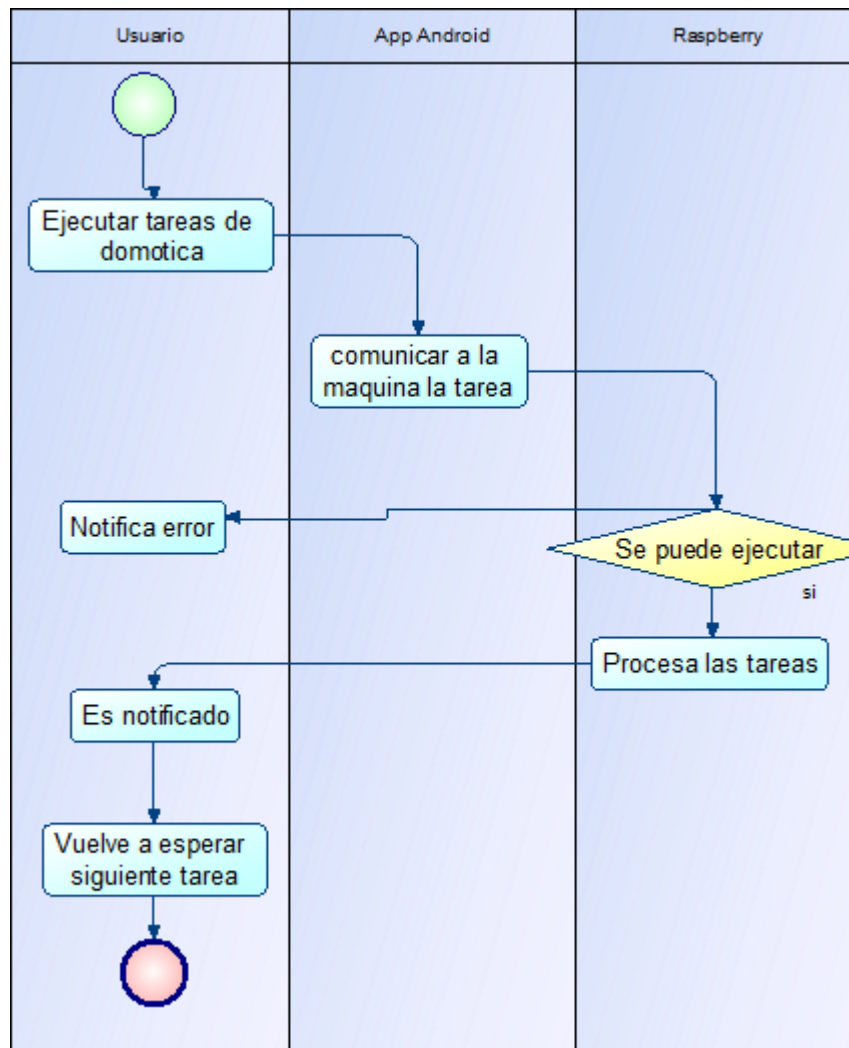


Figura 3: Proceso de Negocios Futuros

En la figura de muestra que el sistema siempre está a la espera de la siguiente tarea a ejecutar, es posible que esta no se puede realizar, porque ya se está llevando a cabo con anterioridad.

## 4.2 Modelo de casos de uso

### 4.2.1 Diagrama de casos de uso

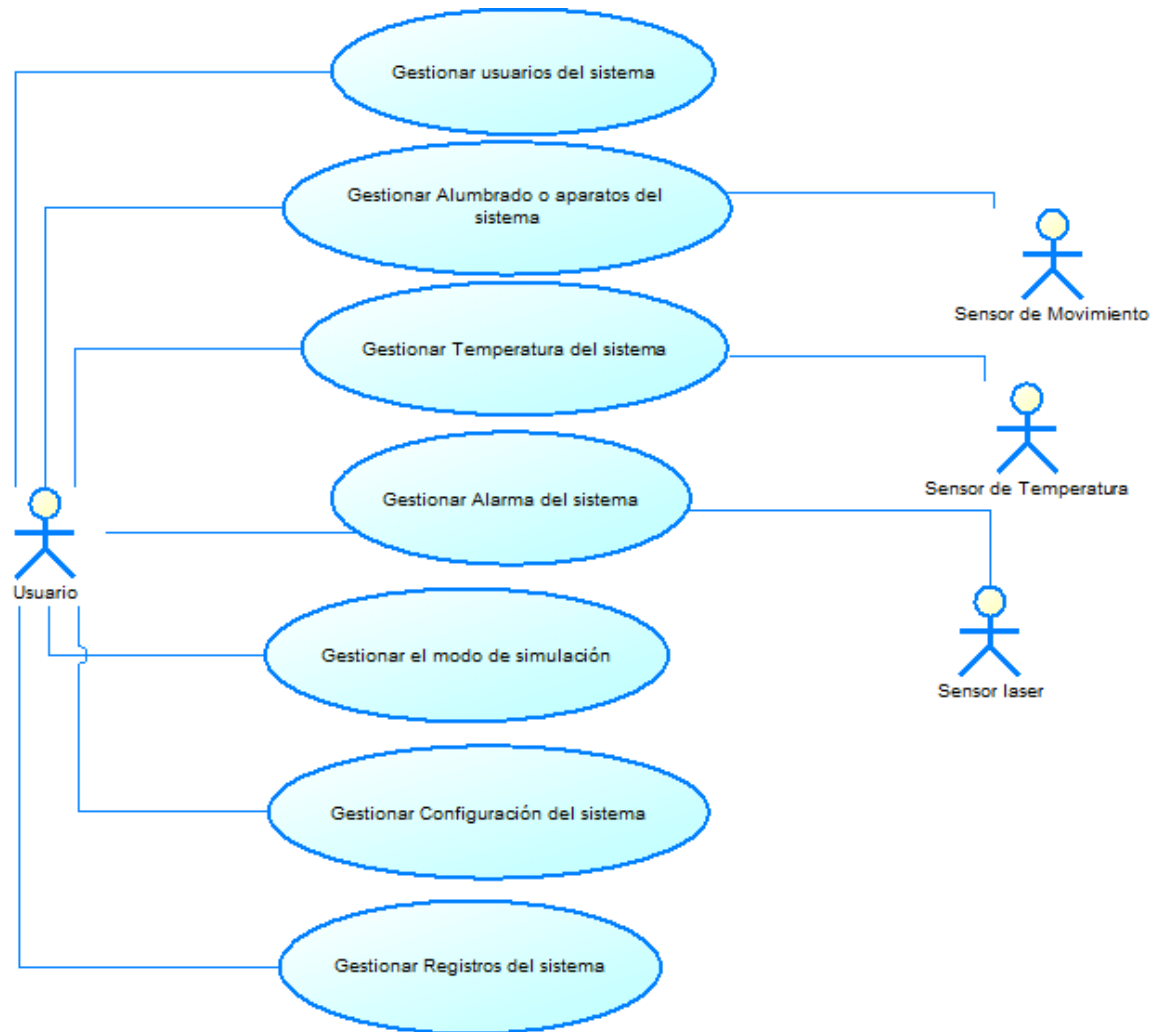


Figura 4: Diagrama de casos de uso del proyecto

## **Actores:**

**Usuario:** Este actor es el encargado de interactuar con el sistema y tiene todos los permisos de administrador del sistema. Los conocimientos técnicos que tiene que tener este actor son; saber manejar un Smartphone a nivel de usuario básico.

**Sensor de movimiento:** Este actor envía una señal al sistema cuando ha detectado algún movimiento, envía información en forma de señales digitales las cuales son interpretadas por la máquina y se usarán para gestionar el modo automático de luces

**Sensor de temperatura:** Este actor detecta la temperatura del ambiente y envía esa información en forma de señales digitales las cuales son interpretadas por la máquina y se usarán para gestionar la temperatura

**Sensor laser:** Este actor envía una señal al sistema cuando ha detectado alguna persona dentro del inmueble, envía información en forma de señales digitales las cuales son interpretadas por la máquina y se usarán para gestionar el módulo de alarma.



## 4.2.2 Especificación de los Casos de Uso

A continuación se presentan todas las tablas con el detalle de la especificación de los casos de uso del sistema

CU_01.1 Prender luz/aparato			
2Versión	1.1		
Dependencias	RF_01 Gestionar Alumbrado y aparatos del sistema		
Precondición	El usuario debe tener la aplicación abierta en su teléfono móvil o la plataforma web.		
	El usuario debe estar en la sección de Alumbrado		
Descripción	El usuario podrá activar una luz del inmueble		
Secuencia Normal	Secuencia	Acción	
	1	1.1	El usuario solicita al sistema encender una luz
		1.2	El usuario deber presionar el botón correspondiente a la luz o aparato a encender.
		1.3	La aplicación se comunica con la Raspberry mandando la señal de encendido
		1.4	La Raspberry enciende la luz que el usuario ha indicado.
		1.5	La Raspberry guarda este registro en su memoria.
Post condición	El usuario queda en la misma pantalla y puede ejecutar otra acción		
Excepciones	Secuencia	Acción	
	1.3	1.2	La luz ya estaba encendida
		1.6	El sistema no es capaz de realizar esta acción
		1.7	Nada cambia por pantalla

CU_01.2 Apagar luz/aparato			
Versión	1.1		
Dependencias	RF_01 Gestionar Alumbrado y aparatos del sistema		
Precondición	El usuario debe tener la aplicación abierta en su teléfono móvil o la plataforma web.		
	El usuario debe estar en la sección de Alumbrado		
Descripción	El usuario podrá apagar una luz del inmueble		
Secuencia Normal	Secuencia	Acción	
	1	1.1	El usuario solicita al sistema apagar una luz
		1.2	El usuario deber presionar el botón correspondiente a la luz o aparato a encender.
		1.3	La aplicación se comunica con la Raspberry mandando la señal de apagado
		1.4	La Raspberry apaga la luz que el usuario ha indicado.
		1.5	La Raspberry guarda este registro en su memoria.
Post condición	El usuario queda en la misma pantalla y puede ejecutar otra acción		
Excepciones	Secuencia	Acción	
	1.3	1.2	La luz ya estaba apagada
		1.6	El sistema no es capaz de realizar esta acción
		1.7	Nada cambia por pantalla

CU_01.2 Encender modo automático de luces			
Versión	1.1		
Dependencias	RF_01 Gestionar Alumbrado y aparatos del sistema		
Precondición	El usuario debe tener la aplicación abierta en su teléfono móvil o la plataforma web.		
	El usuario debe estar en la sección de Alumbrado		
Descripción	El usuario podrá activar una luz automáticamente del inmueble		
Secuencia Normal	Secuencia	Acción	
	1	1.1	El usuario solicita al sistema encender modo automático de luces
		1.2	El usuario deber presionar el botón de encendido automático
		1.3	La aplicación se comunica con la Raspberry mandando la señal de encendido para este modo
		1.4	La Raspberry enciende la luces automáticamente
		1.5	La Raspberry guarda este registro en su memoria.
Post condición	El usuario queda en la misma pantalla y puede ejecutar otra acción		
Excepciones	Secuencia	Acción	
	1.3	1.2	El modo ya estaba encendido
		1.6	El sistema no es capaz de realizar esta acción
		1.7	Nada cambia por pantalla

CU_01.3 Encender modo automático de luces	
Versión	1.1

Dependencias	RF_01 Gestionar Alumbrado y aparatos del sistema		
Precondición	El usuario debe tener la aplicación abierta en su teléfono móvil o la plataforma web.		
	El usuario debe estar en la sección de Alumbrado		
Descripción	El usuario podrá desactivar el modo de encendido automatico		
Secuencia Normal	Secuencia	Acción	
	1	1.1	El usuario solicita al sistema apagar modo automático de luces
		1.2	El usuario deber presionar el botón de apagado de modo automático de luces
		1.3	La aplicación se comunica con la Raspberry mandando la señal de apagado para este modo
		1.4	La Raspberry apaga las luces automáticamente
		1.5	La Raspberry guarda este registro en su memoria.
Post condición	El usuario queda en la misma pantalla y puede ejecutar otra acción		
Excepciones	Secuencia	Acción	
	1.3	1.2	El modo ya estaba apagado
		1.6	El sistema no es capaz de realizar esta acción
		1.7	Nada cambia por pantalla

CU_02.1 Activar regulación de temperatura automática	
Versión	1.1
Dependencias	RF_02 Gestionar Temperatura

Precondición	El usuario debe tener la aplicación abierta en su teléfono móvil o la plataforma web.		
	El usuario debe estar en la sección de Temperatura		
Descripción	El usuario podrá activar la temperatura ideal y regularla		
Secuencia Normal	Secuencia	Acción	
	1	1.1	El usuario solicita al sistema activar regulación de temperatura
		1.2	El usuario deber seleccionar un temperatura que el considere ideal y aceptar la activación del servicio
		1.3	La aplicación se comunica con la Raspberry mandando la señal de regulación
		1.4	La Raspberry comienza a subir o bajar la temperatura según corresponda
		1.5	La Raspberry guarda este registro en su memoria.
Post condición	El usuario queda en la misma pantalla y puede ejecutar otra acción		
Excepciones	Secuencia	Acción	
	1.4	1.5	El sistema no tiene conectado ningún aparato para regular temperatura
		1.6	El sistema queda en un ciclo esperando que se conecte un aparato
		1.7	Nada cambia por pantalla

CU_02.2 Apagar regulación de temperatura automática	
Versión	1.1
Dependencias	RF_02 Gestionar Temperatura
Precondición	El usuario debe tener la aplicación abierta en su teléfono móvil o la plataforma web.

	El usuario debe estar en la sección de Temperatura		
Descripción	El usuario podrá activar la temperatura ideal y regularla		
Secuencia Normal	Secuencia	Acción	
	1	1.1	El usuario solicita al sistema desactivar regulación de temperatura
		1.2	El usuario deber seleccionar apretar el botón de apagado de regulación de temperatura
		1.3	La aplicación se comunica con la Raspberry mandando la señal de corte del servicio
		1.4	La Raspberry detiene dicho servicio
		1.5	La Raspberry guarda este registro en su memoria.
Post condición	El usuario queda en la misma pantalla y puede ejecutar otra acción		
Excepciones	Secuencia	Acción	
	1.4	1.5	El sistema no tiene este servicio corriendo
		1.6	El sistema no hace nada
		1.7	Nada cambia por pantalla

CU_02.2 Activar el aumento de temperatura	
Versión	1.1
Dependencias	RF_02 Gestionar Temperatura
Precondición	El usuario debe tener la aplicación abierta en su teléfono móvil o la plataforma web.

	El usuario debe estar en la sección de Temperatura		
Descripción	El usuario podrá activar el aumento de temperatura		
Secuencia Normal	Secuencia	Acción	
	1	1.1	El usuario solicita al sistema aumentar la temperatura
		1.2	El usuario presionar el botón que dice “calentar”
		1.3	La aplicación se comunica con la Raspberry mandando la señal de regulación
		1.4	La Raspberry comienza a subir la temperatura según corresponda
		1.5	La Raspberry guarda este registro en su memoria.
Post condición	El usuario queda en la misma pantalla y puede ejecutar otra acción		
Excepciones	Secuencia	Acción	
	1.4	1.5	El sistema no tiene conectado ningún aparato para regular temperatura
		1.6	El sistema ejecuta la tarea internamente pero nada se enciende
		1.7	Nada cambia por pantalla

CU_02.3 Activar la disminución de temperatura	
Versión	1.1
Dependencias	RF_02 Gestionar Temperatura
Precondición	El usuario debe tener la aplicación abierta en su teléfono móvil o la plataforma web.
	El usuario debe estar en la sección de Temperatura

Descripción	El usuario podrá activar la disminución de temperatura		
Secuencia Normal	Secuencia	Acción	
	1	1.1	El usuario solicita al sistema la disminución de la temperatura
		1.2	El usuario presionar el botón que dice “enfriar”
		1.3	La aplicación se comunica con la Raspberry mandando la señal de regulación
		1.4	La Raspberry comienza a bajar la temperatura según corresponda
		1.5	La Raspberry guarda este registro en su memoria.
Post condición	El usuario queda en la misma pantalla y puede ejecutar otra acción		
Excepciones	Secuencia	Acción	
	1.4	1.5	El sistema no tiene conectado ningún aparato para regular temperatura
		1.6	El sistema ejecuta la tarea internamente, pero nada se enciende
		1.7	Nada cambia por pantalla

CU_02.4 Parar el aumento o la disminución de temperatura	
Versión	1.1
Dependencias	RF_02 Gestionar Temperatura
Precondición	El usuario debe tener la aplicación abierta en su teléfono móvil o la plataforma web.
	El usuario debe estar en la sección de Temperatura
Descripción	El usuario podrá desactivar el aumento o la disminución de temperatura



Secuencia Normal	1	Acción	
		1.1	El usuario solicita al sistema para la disminución o aumento de la temperatura
		1.2	El usuario presionar el botón que dice “parar”
		1.3	La aplicación se comunica con la Raspberry mandando la señal de regulación
		1.4	La Raspberry para la regulación de temperatura según corresponda
		1.5	La Raspberry guarda este registro en su memoria.
Post condición	El usuario queda en la misma pantalla y puede ejecutar otra acción		
Excepciones	1.4	Acción	
		1.5	El sistema no tiene conectado ningún aparato para regular temperatura
		1.6	El sistema ejecuta la tarea internamente, pero nada se apaga
		1.7	Nada cambia por pantalla

CU_03.1 Activar la alarma del sistema			
Versión	1.1		
Dependencias	RF_03 Gestionar Alarma del sistema		
Precondición	El usuario debe tener la aplicación abierta en su teléfono móvil o la plataforma web.		
	El usuario debe estar en la sección de Seguridad		
Descripción	El usuario podrá activar la alarma del sistema desde el móvil		
Secuencia Normal	Secuencia		Acción
	1	1.1	El usuario solicita al sistema activar la alarma en el sistema

		1.2	El usuario deber presionar el botón que dice 'activar alarma'
		1.3	La aplicación se comunica con la Raspberry mandando la señal de encendido
		1.4	La Raspberry activa la alarma
		1.5	La Raspberry emite una señal de notificación sonora de encendido
Post condición	El usuario queda en la misma pantalla y puede ejecutar otra acción,  Un sensor de rayo láser se prende		
Excepciones	Secuencia	Acción	
	1.3	1.2	La alarma ya estaba activada
		1.6	El sistema no es capaz de realizar esta acción
		1.7	Nada cambia por pantalla

CU_03.2 Desactivar la alarma del sistema			
Versión	1.1		
Dependencias	RF_03 Gestionar Alarma del sistema		
Precondición	El usuario debe tener la aplicación abierta en su teléfono móvil o la plataforma web.		
	El usuario debe estar en la sección de Seguridad		
Descripción	El usuario podrá desactivar la alarma del sistema desde el móvil		
Secuencia Normal	Secuencia	Acción	
	1	1.1	El usuario solicita al sistema desactivar la alarma en el sistema

		1.2	El usuario deber presionar el botón que dice 'desactivar alarma'
		1.3	La aplicación se comunica con la Raspberry mandando la señal de apagado
		1.4	La Raspberry desactivar la alarma
		1.5	La Raspberry emite una señal de notificación sonora de apagado
Post condición	El usuario queda en la misma pantalla y puede ejecutar otra acción,  Un sensor de rayo láser se apaga		
Excepciones	Secuencia	Acción	
	1.3	1.2	La alarma ya estaba desactivada
		1.6	El sistema no es capaz de realizar esta acción
		1.7	Nada cambia por pantalla

CU_04.1 Activar el modo de simulación simple			
Versión	1.1		
Dependencias	RF_03 Gestionar el modo de simulación		
Precondición	El usuario debe tener la aplicación abierta en su teléfono móvil o la plataforma web.		
	El usuario debe estar en la sección de Seguridad		
Descripción	El usuario podrá activar desde el móvil el modo de simulación de alumbrado simple		
Secuencia Normal	Secuencia	Acción	
	1	1.1	El usuario solicita al sistema activar el modo de simulación simple

		1.2	El usuario deber presionar el botón que dice 'activar simulación simple'
		1.3	La aplicación se comunica con la Raspberry mandando la señal de encendido de simulación
		1.4	La Raspberry activa la simulación
		1.5	La Raspberry emite una señal de notificación sonora de comienzo de simulación simple
Post condición	<p>El usuario queda en la misma pantalla y puede ejecutar otra acción,</p> <p>Las luces del inmueble se prenden o apagan acorde a la emulación.</p>		
Excepciones	Secuencia	Acción	
	1.3	1.2	El modo de simulación simple ya estaba activado
		1.6	El sistema no es capaz de realizar esta acción
		1.7	Nada cambia por pantalla

CU_04.2 Desactivar el modo de simulación simple			
Versión	1.1		
Dependencias	RF_03 Gestionar el modo de simulación		
Precondición	El usuario debe tener la aplicación abierta en su teléfono móvil o la plataforma web.		
	El usuario debe estar en la sección de Seguridad		
Descripción	El usuario podrá desactivar desde el móvil el modo de simulación de alumbrado simple		
Secuencia Normal	Secuencia	Acción	
	1	1.1	El usuario solicita al sistema desactivar el modo de simulación simple

		1.2	El usuario deber presionar el botón que dice 'desactivar simulación simple'
		1.3	La aplicación se comunica con la Raspberry mandando la señal de apagado de simulación
		1.4	La Raspberry desactivar la simulación
		1.5	La Raspberry emite una señal de notificación sonora de finalización de simulación simple
Post condición	<p>El usuario queda en la misma pantalla y puede ejecutar otra acción,</p> <p>Las luces del inmueble ya no se encienden o apagan automáticamente.</p>		
Excepciones	Secuencia	Acción	
	1.3	1.2	El modo de simulación simple ya estaba desactivado
		1.6	El sistema no es capaz de realizar esta acción
		1.7	Nada cambia por pantalla

CU_04.3 Activar el modo de simulación avanzada			
Versión	1.1		
Dependencias	RF_03 Gestionar el modo de simulación		
Precondición	El usuario debe tener la aplicación abierta en su teléfono móvil o la plataforma web.		
	El usuario debe estar en la sección de Seguridad		
Descripción	El usuario podrá activar desde el móvil el modo de simulación de alumbrado avanzada		
Secuencia Normal	Secuencia	Acción	
	1	1.1	El usuario solicita al sistema activar el modo de simulación avanzada

		1.2	El usuario deber presionar el botón que dice 'activar simulación avanzada
		1.3	La aplicación se comunica con la Raspberry mandando la señal de encendido de simulación
		1.4	La Raspberry activa la simulación
		1.5	La Raspberry emite una señal de notificación sonora de comienzo de simulación avanzada
Post condición	<p>El usuario queda en la misma pantalla y puede ejecutar otra acción,</p> <p>Las luces del inmueble se prenden o apagan acorde a la emulación.</p>		
Excepciones	Secuencia	Acción	
	1.3	1.2	El modo de simulación avanzada ya estaba activado
		1.6	El sistema no es capaz de realizar esta acción
		1.7	Nada cambia por pantalla

CU_04.4 Desactivar el modo de simulación avanzada			
Versión	1.1		
Dependencias	RF_03 Gestionar el modo de simulación		
Precondición	El usuario debe tener la aplicación abierta en su teléfono móvil o la plataforma web.		
	El usuario debe estar en la sección de Seguridad		
Descripción	El usuario podrá desactivar desde el móvil el modo de simulación de alumbrado avanzada		
Secuencia Normal	Secuencia	Acción	
	1	1.1	El usuario solicita al sistema desactivar el modo de simulación avanzada
		1.2	El usuario deber presionar el botón que dice 'desactivar simulación avanzada

		1.3	La aplicación se comunica con la Raspberry mandando la señal de apagado de simulación
		1.4	La Raspberry desactivar la simulación
		1.5	La Raspberry emite una señal de notificación sonora de finalización de simulación avanzada
Post condición	<p>El usuario queda en la misma pantalla y puede ejecutar otra acción,</p> <p>Las luces del inmueble ya no se encienden o apagan automáticamente.</p>		
Excepciones	Secuencia	Acción	
	1.3	1.2	El modo de simulación avanzada ya estaba desactivado
		1.6	El sistema no es capaz de realizar esta acción
		1.7	Nada cambia por pantalla

CU_05.1 Configurar la IP para comunicarse con la Raspberry			
Versión	1.1		
Dependencias	RF_05 Gestionar Configuración del sistema		
Precondición	El usuario debe tener la aplicación abierta en su teléfono móvil.		
	El usuario debe estar en la sección de configuración		
Descripción	El usuario podrá indicarle al sistema la IP de la Raspberry		
Secuencia Normal	Secuencia	Acción	
	1	1.1	El usuario solicita al sistema desplegar la opción de configuración
		1.2	El usuario debe ingresar el número IP correspondiente a la Raspberry
		1.3	La app va a tratar de comunicarse con la Raspberry

		1.5	La Raspberry no hace nada, el usuario debe comprobar manualmente la conexión
Post condición	El usuario queda en la misma pantalla y puede introducir otra IP,		
Excepciones	Secuencia	Acción	
	1.3	1.2	La IP no corresponde a la de la maquina
		1.6	El sistema no es capaz de establecer comunicación con la Raspberry
		1.7	Nada cambia por pantalla

CU_06.1 Visualizar datos guardados en la memoria			
Versión	1.1		
Dependencias	RF_06 Gestionar Datos del sistema		
Precondición	El usuario debe tener la aplicación abierta en su teléfono móvil o la plataforma web.		
	El usuario debe estar en la sección de datos		
Descripción	El sistema permite al usuario poder visualizar los datos que el sistema guardó internamente.		
Secuencia Normal	Secuencia	Acción	
	1	1.1	El usuario solicita al sistema desplegar la opción de datos guardados
		1.2	El usuario deber ingresar el día o los días en que quiere ver el registro
		1.3	EL sistema muestra por pantalla el registro, mediante una grafica
Post condición	El usuario queda en la pantalla de registros		



Excepciones			
	Secuencia	Acción	
	1.2	1.2	El día de registro no fue encontrado o no existe
		1.6	El sistema no es capaz de mostrar registros
		1.7	Notificación de error es mostrado por pantalla

CU_07.1 Cambiar contraseña usuario			
Versión	1.1		
Dependencias	RF_06 Gestionar contraseñas del sistema		
Precondición	El usuario debe tener abierta la plataforma web.		
	El usuario debe estar en la sección de cambiar contraseña		
Descripción	El sistema debe cambiar la contraseña de un usuario		
Secuencia Normal	Secuencia	Acción	
	1	1.1	El usuario solicita al sistema cambiar una contraseña
		1.2	El usuario deber ingresar el nuevo nombre, contraseña admin y contraseña de usuario y aceptar
		1.3	El sistema muestra por pantalla el nuevo registro
Post condición	El usuario queda en la pantalla de contraseña		
Excepciones	Secuencia	Acción	
	1.2	1.2	La contraseña tiene símbolos no aceptados
		1.6	El sistema no es capaz de guardar el registro

		1.7	Notificación de error es mostrado por pantalla
--	--	-----	--

CU_07.2 Crear nuevo usuario			
Versión	1.1		
Dependencias	RF_06 Gestionar contraseñas del sistema		
Precondición	El usuario debe tener abierta la plataforma web.		
	El usuario debe estar en la sección de crear usuario		
Descripción	El sistema debe crear un nuevo usuario		
Secuencia Normal	Secuencia	Acción	
	1	1.1	El usuario solicita al sistema crear un nuevo usuario
		1.2	El usuario debe rellenar los campos con el nuevo nombre de usuario, su contraseña, y la contraseña de admin y aceptar
		1.3	El sistema muestra por pantalla el nuevo registro
Post condición	El usuario queda en la pantalla de crear usuario		
Excepciones	Secuencia	Acción	
	1.2	1.2	La contraseña tiene símbolos no aceptados
		1.6	El sistema no es capaz de guardar el registro
		1.7	Notificación de error es mostrado por pantalla

CU_07.3 Eliminar usuario			
Versión	1.1		
Dependencias	RF_06 Gestionar contraseñas del sistema		
Precondición	El usuario debe tener abierta la plataforma web.		
	El usuario debe estar en la sección de eliminar usuario		
Descripción	El sistema debe eliminar un usuario		
Secuencia Normal	Secuencia	Acción	
	1	1.1	El usuario solicita al sistema eliminar un nuevo usuario
		1.2	El usuario debe seleccionar un usuario de la lista de usuarios y seleccionarlos, luego debe ingresar la contraseña de admin y aceptar
		1.3	El sistema muestra por pantalla el registro eliminado
Post condición	El usuario queda en la pantalla de crear usuario		
Excepciones	Secuencia	Acción	
	1.2	1.2	La contraseña de admin no es correcta
		1.6	El sistema no es capaz de eliminar el registro
		1.7	Notificación de error es mostrado por pantalla

---

## 5 DISEÑO

---

### 5.1 Diseño de Físico de la Base de datos

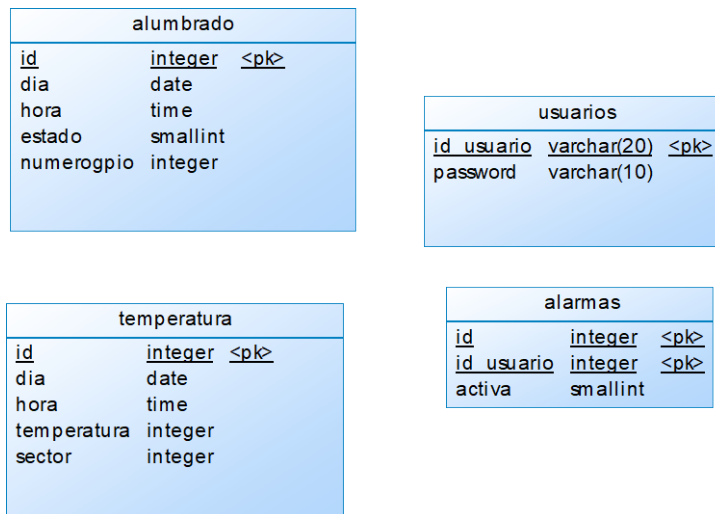


Figura 5. Diseño físico de la base de datos

Ya que el sistema no usa una base de datos extensa, se utilizan registros para almacenar la información alumbrado, temperatura y estado alarmas, y poder utilizarlas en plan de simulación o simplemente para visualizar esta información por pantalla.

## 5.2 Diseño interfaz y navegación

Este es el diseño de navegación de la aplicación para Android. Cabe destacar que se refiere “menú principal” al contenedor que sostiene 3 sliders dentro de la misma ventana, la slider por defecto es “menú alumbrado”.

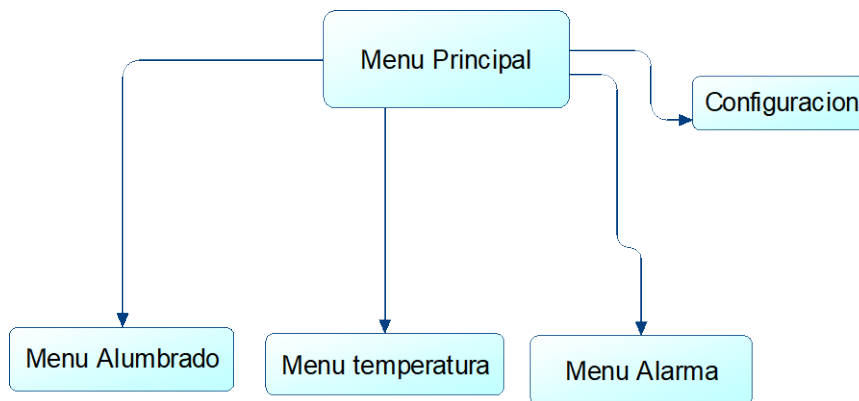


Figura 5.1 Diseño interfaz y navegación.

Este es el menú de navegación web, está compuesto por el menú principal, que contiene la mayoría de las funcionalidades del sistema para facilitar un fácil manejo por parte del usuario, contiene también 2 menú, los cuales son; modificar contraseñas (para uso de la alarma) donde se puede acceder a agregar un nuevo usuario. El segundo menú son los registros donde se puede visualizar mediante gráficos la información guardada en el sistema

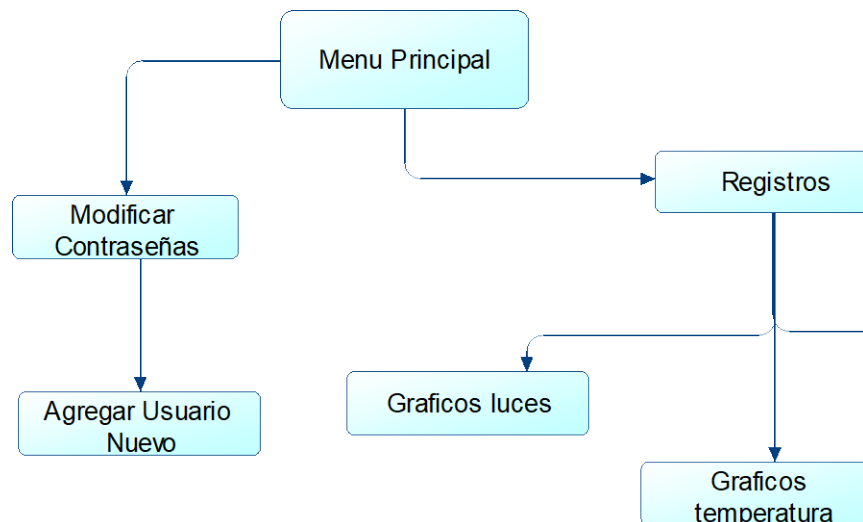


Figura 5.2 Menú de navegación web.

Figura 5.3

Este es el menú de alumbrado, contiene las luces para activar/desactivar, y un botón

especial de “modo automático”, que enciende o apaga todo de manera automática.

Figura. 5.4



Esta es la sección de Temperatura, desde acá se selecciona una temperatura que el usuario desee para que el sistema la ajuste automáticamente.

Adicionalmente se muestra también la temperatura actual del sistema y un botón para actualizar la lectura de esta.

Figura 5.5

Este es el menú de Seguridad, desde acá se puede activar/desactivar la alarma, y también se pueden seleccionar las emulaciones que el usuario desee.

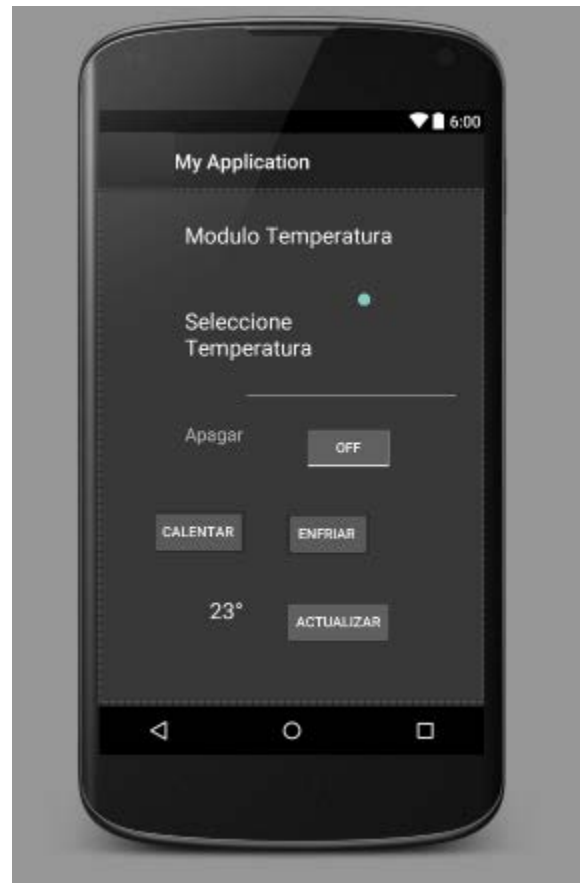


Figura. 5.6

Este es el menú de configuración, esta acción es obligada para establecer comunicación con la Raspberry, solo se debería modificar cuando la aplicación Android no conecte con la maquina.

Se debe poner la IP de esta misma y aceptar. La IP por defecto es 192.168.1.10, esto puede variar dependiendo de la configuración de la red

### Interfaz de navegación web





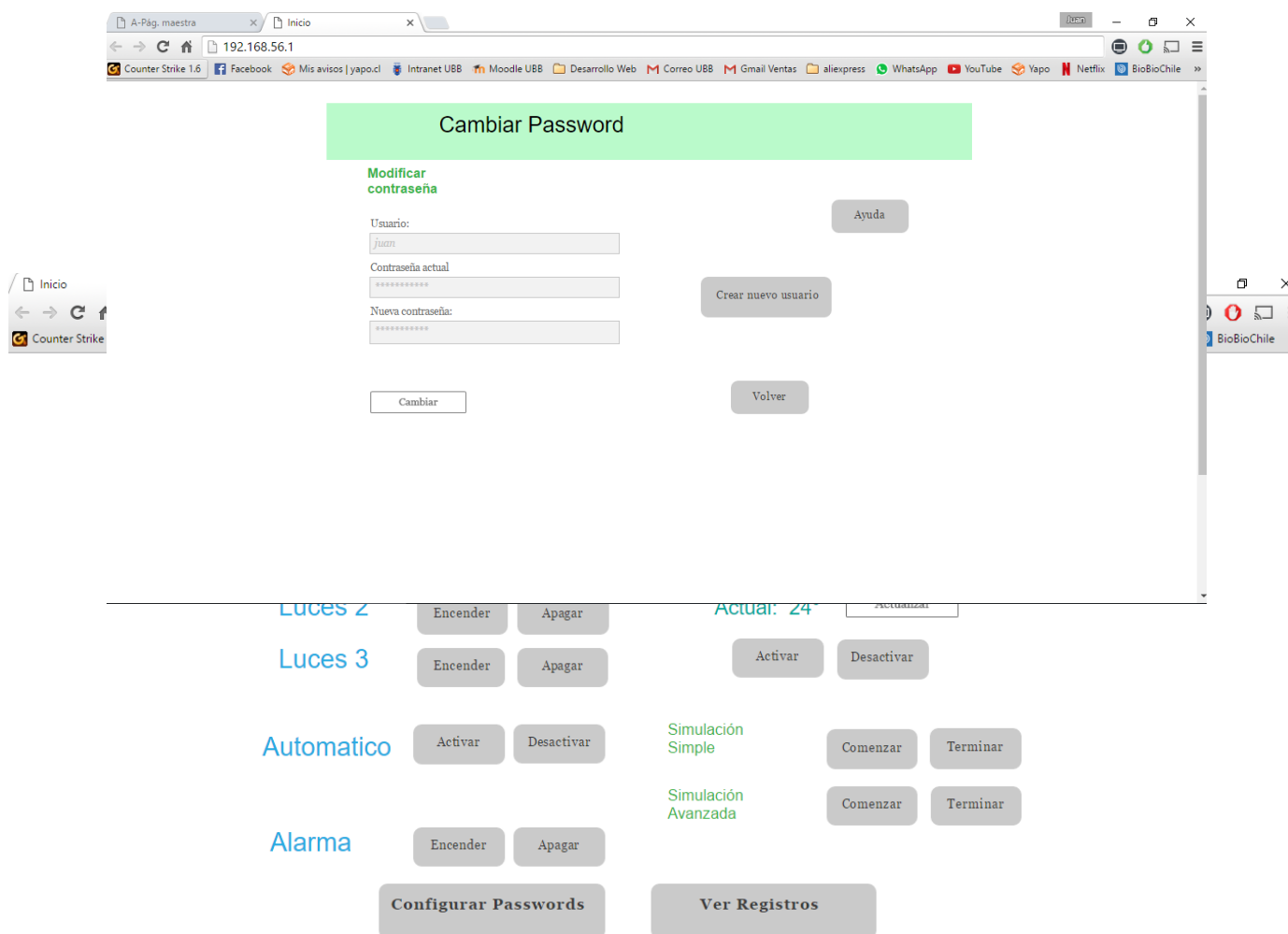
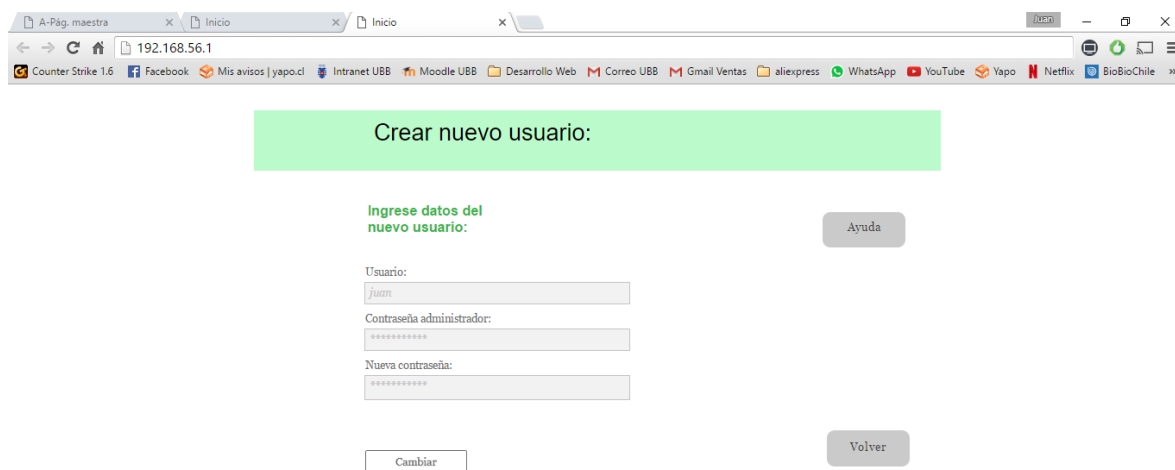


Figura 5.7 Esta plataforma web, permite al usuario controlar las mismas funcionalidades que con tiene la aplicación para Android, pero adicionalmente el usuario tiene la posibilidad de visualizar los registros que la maquina ha guardado en su memoria, así como administrar las contraseñas que contiene el sistema para el funcionamiento de la alarma.

Figura 5.8 Menú para cambiar la contraseña de algún usuario

Figura. 5.9 Menú para crear un nuevo usuario y su contraseña



The screenshot shows a web browser window with the address bar displaying '192.168.56.1'. The browser's toolbar includes various icons for social media and utilities. The main content area features a green header with the text 'Crear nuevo usuario:'. Below this, there is a section titled 'Ingrese datos del nuevo usuario:' in green. This section contains three input fields: 'Usuario:' with the text 'juan' entered, 'Contraseña administrador:' with masked characters, and 'Nueva contraseña:' with masked characters. To the right of the input fields are two buttons: 'Ayuda' and 'Volver'. At the bottom left of the form is a button labeled 'Cambiar'.

Crear nuevo usuario:

**Ingrese datos del nuevo usuario:**

Usuario:  
juan

Contraseña administrador:  
\*\*\*\*\*

Nueva contraseña:  
\*\*\*\*\*

Ayuda

Volver

Cambiar

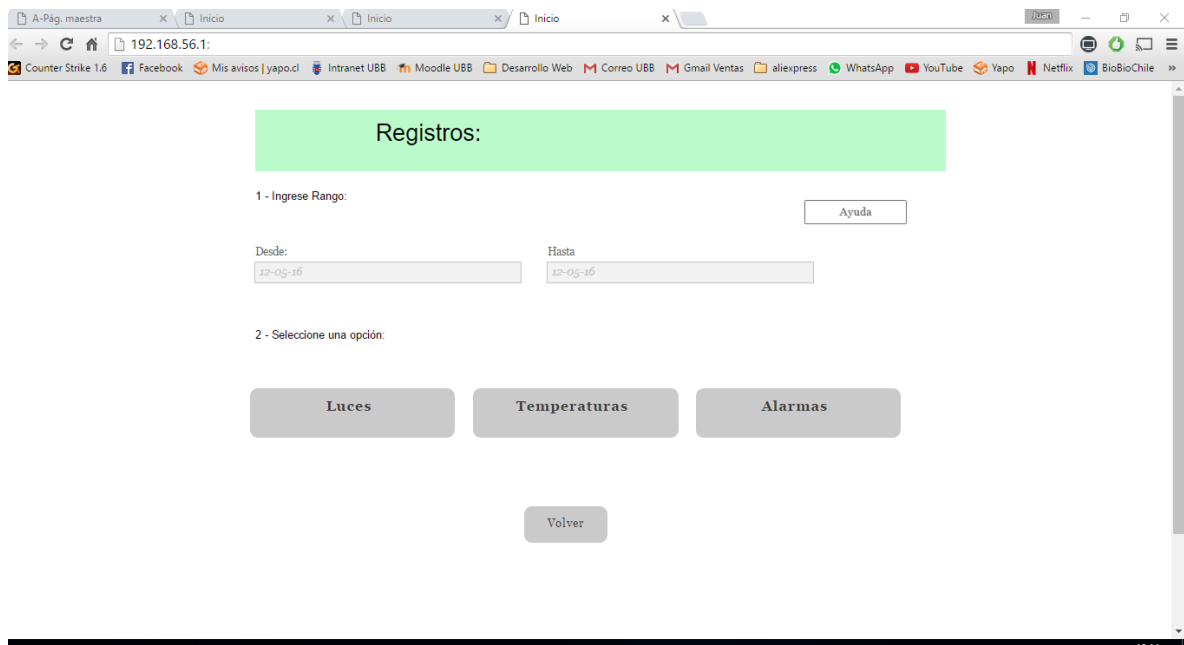


Figura. 5.10 Menú que permite seleccionar un registro para visualizar por pantalla

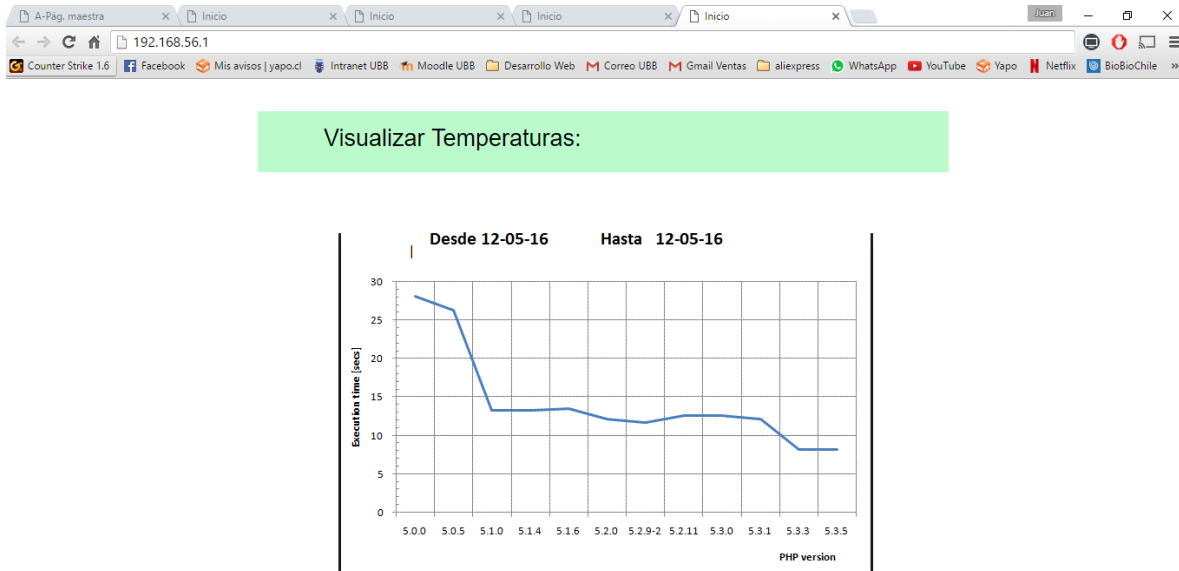


Figura. 5.11

Ventana que permite visualizar la gráfica que el usuario ha seleccionado, en este caso se muestra la temperatura registrada durante un día cualquiera

## Modelo Arquitectónico: Cliente-Servidor

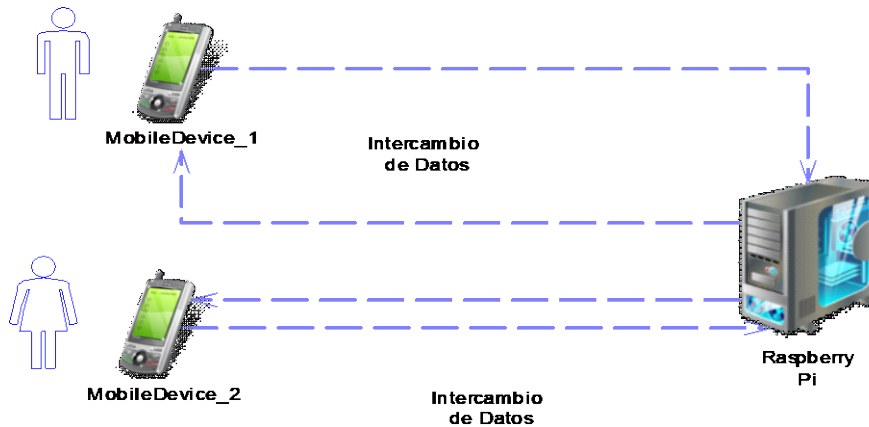


Figura 6. Modelo Arquitectónico

En este modelo arquitectónico la Raspberry actúa como el servidor del sistema, este contiene y ejecuta todos los servicios que los usuarios ordenen. Es posible de administrar muchos clientes al mismo tiempo.

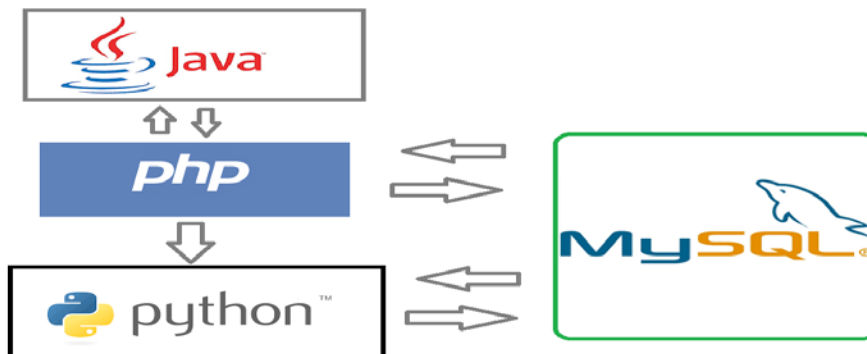


Figura 7. Esquema de lenguajes utilizados

Los lenguajes usados fueron 3, Java, PHP y Python, de los cuales solo 2 se comunican con la base de datos directamente. Java se utilizó para la aplicación de Android, PHP se utilizó como intermediario para enviar y recibir las peticiones de la app, y Python se utilizó para ejecutar las tareas que requieren la interacción con los dispositivos externo tales como sensores y artefactos.

## 6 ESPECIFICACIÓN DE CÓDIGO

Algoritmo en Python	Descripción
<i>Salida_on.py</i>	Es el script más simple de todos, pero el más elemental, activa una salida del Raspberry y hace que se encienda una luz o un artefacto que esté conectado a esa salida.
<i>Salida_off.py</i>	Al igual que el script anterior, este hace lo contrario, y es usado en muchas salidas de luces, lo que hace es mandar la señal de apagado al sistema
<i>Registrar_uso.py</i>	Este Script está permanentemente corriendo en segundo plano y lo que hace es registrar el uso de las salidas, es decir cuando se usan las luces y otros artefactos
<i>Calcular_simulacion.py</i>	Este algoritmo hace un cálculo del uso del alumbrado, con los registros del sistema y entrega un promedio
<i>Inicia_emulacion.py</i>	Este Script toma el control de la salida GPIO, y simula un uso real del alumbrado a partir del promedio que le entregó el algoritmo anterior
<i>Finalizar_gpio.py</i>	Es Script es elemental en el uso del puerto GPIO, se tiene que usar cada vez que finalice el uso de una salida para que no quede nada almacenado en su buffer
<i>Temperatura.py</i>	Este script es uno de los más destacados del sistema, está constantemente trabajando cuando se le activa y regula automáticamente la temperatura del sistema
<i>Calentar.py</i>	Este Script activa una salida para que el inmueble suba su temperatura
<i>Enfriar.py</i>	Este Script activa una salida para que el inmueble baje su temperatura
<i>Finaliza_temp.py</i>	Este Script Finaliza todos los Scripts de temperatura y limpia el puerto GPIO
<i>Alarma_on.py</i>	Este Script trabaja con los sensores y activa la alarma del sistema
<i>Alarma_off.py</i>	Este Script se encarga de verificar que la contraseña de la alarma es correcta y la desconecta.
<i>Automatico.py</i>	Este Script hace que el sistema de alumbrado trabaje automáticamente

## 7 RESUMEN ESFUERZO REQUERIDO

Actividades/fases	N° Horas
<b>REQUERIMIENTOS y DISEÑO</b>	
Desarrollo de la Entrevista	1
Entrevista al Cliente	1
Definición de requerimientos	2
Análisis de Requerimientos	2
Especificación de Requerimientos	4
Entrega primer informe	8
Corrección de requerimientos	3
<b>PROGRAMACION</b>	
Modelo relacional y BPMN	5
Creación de Algoritmos	30
Programación de algoritmos en Python	20
Programación de algoritmos en Android	50
Especificación casos de uso	5
Prototipo interfaz navegación	10
Implementación Local	20
<b>TOTAL</b>	<b>141</b>

---

## 8 CONCLUSIONES

---

En el proyecto desarrollado y que fue expuesto en este presente informe se ha logrado cumplir en su mayoría con los objetivos del desarrollo del sistema, y casi todas las funcionalidades del proyecto, se dejaron de lado o se modificaron algunas por el bien común del software. El resultado final del proyecto deja como resultado un sistema totalmente viable para cualquier inmueble, fácil de implementar y muy sencillo de usar. Este a su vez, deja abierta la opción de seguir desarrollándolo e implementar características como el acceso remoto al sistema o uso de cámaras web.

Respecto al desarrollo del proyecto, se tuvo que realizar de manera ágil para cumplir con los plazos establecidos y no empobrecer la calidad del software. Sin embargo, el proyecto se retrasó 2 meses más de acuerdo a la planificación inicial, ya que no se tenían todos los conocimientos necesarios en profundidad para el desarrollo del sistema, y se tuvo que investigar en las áreas donde fue necesario. En cuanto a los lenguajes que se usaron, Python fue un lenguaje casi obligado para trabajar con la Raspberry, ya que todas las librerías para conectar con elementos externos como sensores estaban escritas en ese lenguaje.

En el ámbito profesional, se usaron muchos de los conocimientos aprendidos durante la carrera, pero además se hizo mucha investigación sobre la materia sobre todo en el ámbito de la electrónica, que es un área muy cercana a la informática. En el ámbito académico, se enriquecieron muchas de las habilidades de desarrollo y programación de las cuales se creían que estaban bien formadas, pero sin lugar a dudas es otra oportunidad para seguir en el camino de la constante formación como futuro profesional.

---

## 9 BIBLIOGRAFÍA

---

Las referencias que se usaron el proyecto fueron en su totalidad 100% digitales  
Aquí se mencionas dichas fuentes

[1] <https://www.raspberrypi.org>

La mas importante de todas, de donde se sacaron todas las instrucciones iniciales.

[2] [https://github.com/szazo/DHT11\\_Python](https://github.com/szazo/DHT11_Python)

Librerías para los sensores de temperatura

[3] <http://www.instructables.com/id/Control-Raspberry-Pi-GPIO-Using-an-App>

Referencias de uso de Android con la Raspberry

[4] <http://php.net/docs.php>

Documentación sobre PHP

[5] <https://www.raspberrypi.org/documentation/linux/usage/users.md>

Ajustes del sistema Raspbian

[6] <https://www.raspberrypi.org/learning/lamp-web-server-with-wordpress/worksheet>

Ajuste Servidor Apache

[7] [https://github.com/adafruit/Adafruit\\_Python\\_DHT](https://github.com/adafruit/Adafruit_Python_DHT)

Librerías de sensores de movimientos

[8] <https://www.stewright.me/2012/09/tutorial-install-phpmyadmin-on-your-raspberry-pi/>

Mas referencias sobre Apache



---

## **10 ANEXO 1: ESTIMACIÓN DE CU**

---

En el anexo 1 se encuentra la estimación de tamaño del software aplicando técnicas basadas puntos de Casos de Uso.

---

## **11 ANEXO 2: PRUEBAS DE SOFTWARE**

---

En este anexo se encuentran las pruebas realizadas al software.

---

## **12 ANEXO 3: DICCIONARIO DE DATOS DEL MODELO DE DATOS**

---

En este apartado, se describen toda la terminología usada para el desarrollo y la interpretación de este proyecto.

## **Anexo 1: Estimación de casos de uso**

Estimación de Tamaño del software aplicando técnicas puntos de Casos de Uso.

Actores	Peso
Usuario	3

UAW: Peso desajustado de actores

1 Actor de tipo Complejo ya que es una persona interactuando a través de una interfaz de usuario.

N°	UUCW (Peso desajustado de casos de uso)	Tipo de CU	Peso	Justificación
1	Prender luz/aparato	Simple	5	Relacionado con 1 tabla en la BD
2	Apagar luz/aparato	Simple	5	Relacionado con 1 tabla en la BD
3	Encender modo automático de luces	Simple	5	Relacionado con 1 tabla en la BD
4	Activar regulación de temperatura automática	Simple	5	Relacionado con 1 tabla en la BD
5	Apagar regulación de temperatura automática	Simple	5	Relacionado con 1 tabla en la BD
6	Activar el aumento de temperatura	Simple	5	Relacionado con 1 tabla en la BD
7	Activar la disminución de temperatura	Simple	5	Relacionado con 1 tabla en la BD
8	Parar el aumento o la disminución de temperatura	Simple	5	Relacionado con 1 tabla en la BD
9	Activar la alarma del sistema	Simple	5	Relacionado con 1 tabla en la BD
10	Desactivar la alarma del sistema	Simple	5	Relacionado con 1 tabla en la BD

11	Activar el modo de simulación simple	Simple	5	Relacionado con 1 tabla en la BD
12	Desactivar el modo de simulación simple	Simple	5	Relacionado con 1 tabla en la BD
13	Activar el modo de simulación avanzada	Simple	5	Relacionado con 1 tabla en la BD
14	Desactivar el modo de simulación avanzada	Simple	5	Relacionado con 1 tabla en la BD
15	Configurar la IP para comunicarse con la Raspberry	Simple	5	Relacionado con 1 tabla en la BD
16	Visualizar datos guardados en la memoria	Simple	5	Relacionado con 1 tabla en la BD
17	Cambiar contraseña usuario	Simple	5	Relacionado con 1 tabla en la BD
18	Crear nuevo usuario	Simple	5	Relacionado con 1 tabla en la BD
19	Eliminar usuario	Simple	5	Relacionado con 1 tabla en la BD

TCF: TFactor

A cada factor se le asigna un valor acorde a la percepción de complejidad que se tiene del factor. Este valor puede ir entre 0 y 5 (En este caso usaremos el valor promedio 3).

Factores de complejidad técnica (TFC)	Peso	Influencia
Sistema Distribuido	2	3
Rendimiento en cuanto a los objetivos	1	3
Eficiencia	1	3
Complejidad de procesamiento interno	1	3
Reusabilidad del código	1	3
Fácil instalación	0,5	3
Fácil operación, usabilidad	0,5	3
Portabilidad	2	3
Editable	1	3
Concurrencia	1	3
Características especiales de seguridad	1	3
Provee acceso directo a terceros	1	3
Facilidades de entrenamiento a usuarios	1	3
Total		42

ECF: EFactor

A cada factor se le asigna un valor acorde a la percepción de impacto que se tiene del factor

Este valor puede ir entre 0 y 5 (En este caso usaremos el valor promedio 3)

Calcular ECF (Factores de complejidad ambiental) Factor ambiental	Peso	Influencia
Familiaridad con objetivos	1,5	3
Experiencia de la aplicación	0,5	3
Experiencia orientada al objeto	1	3
Capacidad de análisis	0,5	3
Motivación	1	3
Estabilidad de requerimientos	2	3
Trabajadores de medio tiempo	-1	3
Dificultad de lenguajes de programación	-1	3
	Total	13.5

a) Calcular UUCP (Puntos de caso de uso desajustados)

$$UUCP = UAW + UUCW$$

$$UUCP = 3 + 95$$

$$UUCP = 98$$

b) Calcular TCF (Factores de complejidad técnica)

$$TCF = 0.6 + (0.01 * TFactor)$$

$$TCF = 0.6 + (0.01 * 42)$$

$$TCF = 1.02$$

c) Calcular EF (Factores ambientales)

$$EF = 1.4 + (-0.03 * EFactor)$$

$$EF = 1.4 + (-0.03 * 13.5)$$

$$EF = 0.995$$

$$UCP = UUCP * TCF * EF \quad UCP = 98 * 1.02 * 0.995 \quad UCP = 99.4602$$

Calculo de horas hombre (HH)

$$UCP * 20 = 1989.204 \text{ HH}$$

## **Anexo 2: Pruebas de Software.**

		Clases validas		Clases invalidas	
Sección	Condición de entrada	Entrada	código	Entrada	Código
1	Prender luz	Presionar luz indicada	Cev1	Presionar otra luz	Cenv2
2	Prender ventilador	Presionar botón de ventilación	Cev3	Presionar botón de calefacción	Cenv4
3	Prender alarma	Presionar botón encendido alarma	Cev5	Presionar botón apagado alarma	Cenv6
4	Encender simulación avanzada	Presionar botón de inicio de simulación	Cev7	Presionar botón apagado simulación	Cenv8

### **Pruebas de aplicación Android**

CP	Clases de equivalencia	Botón
CP1	CEV<1> CEV<2>	“Botón encendido luz 1”
CP2	CEV<3>CEV<4>	“Botón encendido ventilador”
CP3	CEV<5> CEV<6>	“Botón encendido alarma”
CP4	CEV<7>CEV<8>	“Botón encendido de simulación”

**Resultados esperados de pruebas aplicación Android**

Prueba	Resultado
CP1	“Luz seleccionada enciende correctamente”
CP2	“Ventilador enciende correctamente”
CP3	“La alarma se enciende y el sensor entra en estado de alerta”
CP4	“Las luces se encienden o apagan de acuerdo al programa de simulacion”

**Anexo 3: Diccionario de Datos**

En este apartado, se describen toda la terminología usada para el desarrollo y la interpretación de este proyecto.

1. Numero gpio: Numero del puerto de salida gpio de la Raspberry
2. Estado alumbrado: estado tipo booleano que indica encendido o apagado
3. Id del usuario: identificador único tipo int que lleva cada usuario
4. Id del alumbrado: dato tipo int que indica el alumbrado que se registró
5. Id de la temperatura: dato tipo int que indica el sensor de temperatura
6. Password: dato tipo int que indica la contraseña que tiene cada usuario
7. Activa: estado tipo booleano que indica si la alarma esta activada
8. Hora de alumbrado: dato tipo timestamp que indica la hora en que fue tomado el estado de una salida
9. Dia de alumbrado: dato tipo date que indica el dia en que fue tomado el estado de una salida
10. Hora de temperatura: dato tipo timestamp que indica la hora en que fue tomado el estado de temperatura
11. Dia de temperatura dato tipo date que indica el dia en que fue tomado el estado de temperatura

12. Sector de temperatura: dato tipo int que indica el sector donde fue tomada la temperatura
13. Usuario: persona capaz de utilizar el sistema y modificar sus parámetros
14. Boolean: tipo de datos que toma el valor de 0 o 1 (verdadero o falso).
15. Sistema: conjunto de partes interrelacionadas, hardware, software y recursos humanos que permiten procesar y almacenar información.
16. Login: ingreso al sistema, autenticado mediante un nombre de usuario y una contraseña
17. Casos de uso: Modelo utilizado para definir los pasos a seguir para realizar un proceso.
18. BPMN: Bussines Process Model Notation
19. Raspberry Pi: Mini Computadora que almacena gran parte del Sistema
20. Android app: Aplicación Móvil desarrollada para el sistema operativo Android.
21. Sensor DTH11: Sensor de temperatura que utiliza el sistema
22. Sensor HC--SR501: Sensor de Movimiento que utiliza el sistema
23. LDR: Foto-resistor electrónico que utiliza el sistema
24. Laser KY-008: Laser que se utiliza en conjunto con el LDR en el sistema.