



FACULTAD DE INGENIERÍA Y CIENCIAS APLICADAS

PROTOTIPO DE SISTEMA DOMÓTICO CENTRALIZADO A TRAVÉS DE
UNA PÁGINA WEB CON RECONOCIMIENTO FACIAL.

AUTOR

Jefferson David López Zapata

AÑO

2020



FACULTAD DE INGENIERIAS Y CIENCIAS APLICADAS

PROTOTIPO DE SISTEMA DOMÓTICO CENTRALIZADO A TRAVÉS DE UNA
PÁGINA WEB CON RECONOCIMIENTO FACIAL.

Trabajo de Titulación presentado en conformidad con los requisitos
establecidos para optar por el título de Ingeniero en Sistemas de
Computación e Informática.

Profesor Guía

Mr. Eddy Mauricio Armas Pallasco

Autor

Jefferson David López Zapata

Año

2020

DECLARACIÓN DEL PROFESOR GUÍA

"Declaro haber dirigido el trabajo, desarrollo de sistema domótico centralizado a través de una página web con reconocimiento facial que permite automatizar una vivienda, a través de reuniones periódicas con el estudiante Jefferson David López Zapata, en el semestre 2020-1, orientando sus conocimientos y competencias para un eficiente desarrollo del tema escogido y dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación".



Eddy Mauricio Armas Pallasco

Magister en Gerencia de Sistemas y tecnologías de
Información

C.I.1711715803

DECLARACIÓN DEL PROFESOR CORRECTOR

"Declaro haber revisado este trabajo, de sistema domótico centralizado a través de una página web con reconocimiento facial, de Jefferson David López Zapata, en el semestre 2020-2021 dando cumplimiento a todas las disposiciones vigentes que regulan los trabajos de Titulación".



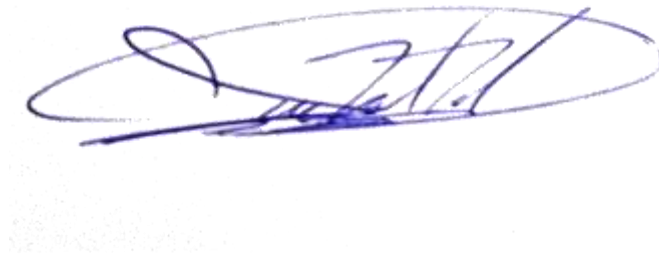
Verónica Fernanda Falconí Ausay

Magister en Ciencias de la Computación y Comercio Electrónico

C.I. 0502395270

DECLARACIÓN DE AUTORÍA DEL ESTUDIANTE

“Declaro que el proyecto de titulación es de mi autoría, se ha utilizado las fuentes respectivas y citas, protegiendo los derechos legales del autor”

A handwritten signature in blue ink, appearing to be 'Jefferson David López Zapata', enclosed within a large, loopy oval shape.

Jefferson David López Zapata

C. I. 1721659462

AGRADECIMIENTOS

Agradezco a mi familia, amigos y conocidos que me impulsaron y apoyaron en toda mi carrera, también agradezco a la universidad que me ha dado las pautas para mi aprendizaje.

DEDICATORIA

Este título lo dedico a mis padres que con tanto esfuerzo me apoyaron en toda mi carrera y fueron mi fortaleza en mis dificultades

Resumen

Hoy en día la tecnología avanza a gran escala, por este motivo es necesario crecer de la misma forma y se propuso construir un prototipo de sistema 'domótico, el cual nos ayudará en la seguridad de una casa con un reconocimiento facial y tener un acceso restringido, la vigilancia por medio de una cámara y, por otro lado, saber a qué temperatura se encuentra el hogar y si existen fugas de gas.

Otra característica es el automatizar acciones que se las realiza diariamente de manera manual como encender luces o abrir una cerradura. Todo eso es posible hacerlo con un sistema centralizado montado en un minicomputador llamado raspberry que además de ser eficiente es muy económico. El control de este sistema se lo hace por medio de una página web que tiene un acceso como administración para crear usuarios y como usuario para controlar los diferentes sensores, actuadores y cámaras.

Abstract

Nowadays technology advances on a large scale, for this reason it is necessary to grow in the same way and it was proposed to build a prototype of a home automation system, which helps us in the security of a house with facial recognition and having restricted access, surveillance by means of a camera and, on the other hand, knowing what temperature the home is at and if there are gas leaks.

Another feature is to automate actions that are carried out daily practices manually such as lights or opening a lock. All this can be done with a centralized system mounted on a minicomputer called raspberry, which besides being efficient is very economical. Control of this system is done through a web page that has access as an administration to create users and as a user to control the different sensors, actuators, and cameras.

INDICE

CAPÍTULO I. INTRODUCCIÓN.....	1
1.1 Antecedentes	1
1.2 Alcance	2
Software.....	2
Hardware	3
1.3 Justificación	3
1.4 Objetivo General	4
1.5 Objetivos Específicos.....	4
CAPÍTULO II. MARCO TEÓRICO.....	4
2.1. Sistema centralizado.....	4
Ventajas de un sistema centralizado	5
Figura 1. Ejemplo de sistema Centralizado	6
2.2 Metodologías ágiles	6
Metodología XP.....	6
Metodología Kanban.....	7
2.3. Metodología Scrum	8
Figura 2: Metodología Scrum	8
Figura 3: Los hitos de la Metodología de trabajo Scrum.	10
2.4. Raspberry pi 3.....	11
2.5. Arduino.....	12
2.7. Reconocimiento Facial.....	13
2.8 Sensores utilizados en el sistema domótico	14
2.8.1 Sensor de temperatura y humedad	15
2.7.2 Sensor de gas	16

2.7.3 Actuator Relé	17
2.8. Lenguajes de programación.....	18
2.8.1 Framework Kumbia	18
2.8.2 Lenguaje PHP	19
2.8.3 Lenguaje Python	20
CAPITULO III. ANALISIS Y DISEÑO	20
3.1 Diagrama del prototipo.....	20
3.2 Planificación.....	21
3.3 Planning Poker.....	23
3.4 Duración y Velocidad.....	24
CAPITULO IV DISEÑO CONSTRUCCION	24
4.1 Implementación.....	25
4.2 Sprint 0.....	25
4.3 Sprint 1.....	26
4.4. Sprint 2.....	34
4.5 Sprint 3.....	40
4.6 Sprint 4.....	46
4.7 Sprint 5.....	52
CAPITULO V CASOS DE PRUEBA	59
5.1 Caso de prueba historia de usuario 1.....	59
CAPITULO VI CONCLUSIONES Y RECOMENDACIONES	69
6.1 Conclusiones.....	69
□ 6.2 Recomendaciones.....	70
□	

REFERENCIAS	71
-------------------	----

CAPÍTULO I. INTRODUCCIÓN

1.1 Antecedentes

El término domótico nace en España en la década de los noventa, proviene de la palabra latín domus que significa casa y del griego automática (que actúa autónomamente). (Sáez F. y Domínguez H, 2006, p.7)

La domótica es un conjunto de sistemas cuyo objetivo principal es automatizar una vivienda, para gestionar energía, seguridad, bienestar y comunicación, los cuales son manejados por redes de cableado o inalámbricas. (Sáez F. y Domínguez H, 2006, p.7)

Hoy en día se puede encontrar distintos sistemas domóticos como es el caso de Samsung SmartThing, el cual consta de un hub (un dispositivo capaz de enlazar módulos ya sean sensores, como también cerraduras o motores compatibles), los cuales permiten que una casa sea totalmente autónoma. (unocero, 2018)

En Ecuador esta tecnología no es tan popular ya que solo existen sistemas básicos, como es el caso de la empresa Innotec el cual vende kit para iluminación, sin embargo, no hay sistemas que midan temperatura, humedad, manejo de motores, reconocimiento facial, dejando de lado el objetivo principal de un sistema domótico que es automatizar una vivienda, además, los costos son muy elevados ya que hay que importarlos, lo que hace muy difícil poder adquirirlos. (INNOTECH, 2017).

Basado en lo expuesto anteriormente, se propone realizar un prototipo de sistema domótico, el cual permitirá automatizar una vivienda por medio de una plataforma centralizada, utilizando un raspberry Pi, con la capacidad de medir

temperatura y humedad, fugas de gas, apertura de puertas, control de luminaria, vigilancia por medio de una página web, como también reconocimiento facial para permitir el ingreso a una persona en específico en el hogar.

1.2 Alcance

En el proyecto se construirá un prototipo electrónico de un sistema domótico centralizado, el cual tendrá la opción de controlar la apertura y cierre de las puertas, encendido y apagado de luces para evitar su manejo manual, por medio de una página web; este permitirá obtener los valores de gas y temperatura para prevenir accidentes e incluirá la administración de cámaras de seguridad. (Rodríguez A., 2014, p.10)

Se realizará un análisis de sistemas domóticos existentes, para poder mejorar sus funcionalidades, como es el caso del reconocimiento facial que además de reconocer el rostro del dueño de una vivienda se podrá darle acceso a la misma, a través de la página web.

A continuación, se detalla el hardware y software del sistema domótico propuesto.

Software

El sistema domótico contiene un servidor apache, la página web que se desarrollará con un framework llamado kumbia (basado en php), la cual cuenta con administración de roles y usuarios para las distintas funcionalidades: ya que como administrador se tendrá la opción de gestionar los roles y usuarios, mientras que como usuario tendrá la opción de controlar los diferentes módulos por medio de una interfaz gráfica amigable al usuario.

Para el reconocimiento facial se utilizará una librería de software libre la cual permitirá guardar los puntos de un rostro registrado para luego identificar el rostro en base a los puntos registrados, de la misma forma todo esto se podrá visualizar en la página web.

Hardware

Inicialmente se utilizará un sensor de temperatura y humedad el cual medirá los niveles registrados en grados Celsius y la humedad relativa (%) y serán reflejados en la página web.

Un sensor de gas mide los valores internamente al presionar el botón de la interfaz gráfica, en caso de existir una fuga de gas se accionará una alarma, de esta manera se podrá tomar acciones para evacuar el lugar o cerrar la llave de gas.

Por otra parte, los módulos relés serán los encargados de encender y apagar los focos al momento de accionar el botón en el sistema, el servomotor permite abrir o cerrar la puerta según le convenga al usuario; por último, para asegurar la vigilancia del lugar, se utilizarán sistemas de seguridad guiados por una cámara para monitorear el lugar.

Para complementar un sistema seguro se instalará una segunda cámara para reconocimiento facial por medio de usb.

1.3 Justificación

Un sistema centralizado permitirá albergar todos los sensores, cámaras y su página web, de modo que permita ser controlado en un solo dispositivo, algo que no existe en otros sistemas domóticos logrando una alta autonomía y mejorando su velocidad ya que se trata de un computador.

Otro ámbito importante es la seguridad en la vivienda ya que en muchas ocasiones se está lejos de casa y en caso de haber algún incendio o robo no se puede tomar acciones, por este motivo se instalará un sensor para determinar las fugas de gas, una cámara de vigilancia por la cual se observará a la persona que se encuentre en la entrada del hogar y brindar acceso automático si es dueño o familia de la vivienda o en caso contrario se dará acceso por medio de la página web.

El utilizar una página web permite que se pueda conectar desde cualquier parte del mundo sin tener que pagar una plataforma o instalar una aplicación, además la misma página web se utilizará en cualquier dispositivo electrónico que tenga acceso a internet.

1.4 Objetivo General

Desarrollar un prototipo de sistema domótico centralizado a través de una página web para visualizar y controlar una vivienda a través de sensores y reconocimiento facial.

1.5 Objetivos Específicos

1. Analizar los requerimientos del sistema domótico orientado al monitoreo y control de una vivienda, teniendo una idea clara de que quiere el cliente.
2. Diseñar el hardware y software del sistema domótico mediante un diagrama de prototipo, de este modo se acoplen y se pueda trabajar en conjunto.
3. Construir el sistema electrónico mediante diagramas con cada sensor y actuador, teniendo una buena gestión de espacio y cables.
4. Validación del funcionamiento de cada sensor y actuador mediante pruebas de ejecución en la página web, de este modo no existan errores.

CAPÍTULO II. MARCO TEÓRICO

2.1. Sistema centralizado

Un sistema centralizado es una arquitectura, donde todos los sensores y actuadores del sistema están conectados a una sola unidad de procesamiento, además, permite que un Ingeniero pueda elegir un sistema centralizado que se ajuste a las necesidades. Los sensores son capaces de recolectar información como por ejemplo medir gas, temperatura, humedad etc. Y los actuadores son utilizados para controlar objetos electrónicos o eléctricos como por ejemplo encender luces, activar motores y demás.

Algo que caracteriza a esta arquitectura es que pueden actuar como receptor de información ya sea de manera inalámbrica o alámbrica, como también pueden actuar como controlador remoto, en el caso del proyecto planteado se utiliza las dos características, ya que recibe información de los sensores y se muestra en la página web, como también transmite la información al presionar el botón en la web y los ejecuta en los actuadores. (Guinaldo M, Sanchez J, Dormido S, 2017, p.20).

Ventajas de un sistema centralizado

- Al ser un sistema centralizado se reduce la posibilidad que se genere algún error.
- Al realizar alguna reparación los técnicos no tienen que ir de equipo en equipo sino al equipo central, de modo que se realizarán los cambios de manera efectiva y rápida.
- Capacidad de crecimiento incremental, ya que se puede conectar otros sensores y actuadores en cuanto sea posible y funcionar perfectamente.

(SIMAD,2017)

Tomando en cuenta las ventajas existentes se utilizará un solo dispositivo (raspberry pi 3), en efecto, el raspberry pi reducirá el riesgo de generar algún error ya que procesa información muy pequeña en un procesador que lo abastece.

Las reparaciones que se pueden realizar son muy pequeñas puesto que, los únicos daños que pueden surgir serian de cableado, sin embargo, si se produjera un daño en el sensor se lo reemplazaría y el problema será solucionado,

El sistema domótico puede alcanzar mayor crecimiento, ya que se es compatible con varios sensores y actuadores que se usan en Arduino y otros dispositivos.

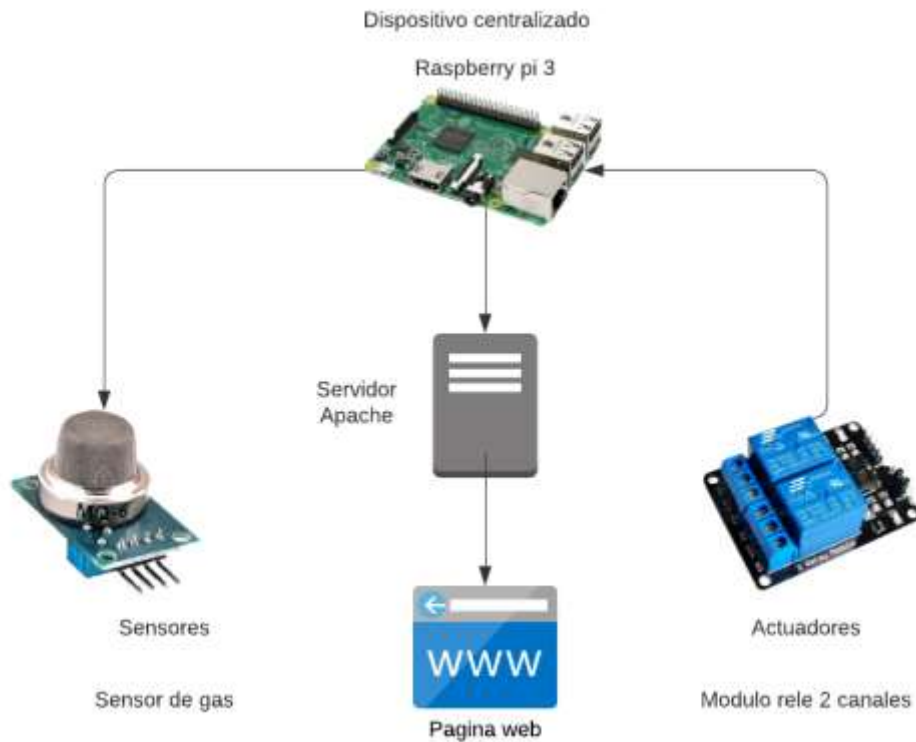


Figura 1. Ejemplo de sistema Centralizado

Como se observa en la figura 1 el dispositivo principal es el raspberry pi 3, que a través de un servidor el cual contiene una página web, se puede manejar cualquier sensor o actuador que se encuentre conectado al sistema centralizado.

2.2 Metodologías ágiles

Metodología XP

La metodología XP consta de pruebas y errores para el desarrollo, lo cual permite una participación continua con el cliente para un resultado exitoso del proceso, por lo tanto, promueve una colaboración entre el equipo en un buen ambiente laboral, también cabe destacar que la programación es en parejas.

El ciclo de vida esta compuesto por varias etapas que son:

- Exploración
- Planificación de la entrega
- Iteraciones
- Producción
- Mantenimiento
- Muerte del proyecto

(Armijos K. 2019, p.10)

Metodología Kanban

Kanban consiste en mejorar el flujo de trabajo del equipo, maximizando productividad y calidad en el producto final. La metodología kanban a diferencia de las otras se adapta a procesos de trabajo sin realizar grandes cambios, de esta manera se puede aplicar otras metodologías de gestión de tareas, tal es el caso de Scrum.

Como funciona

En un tablero se divide en varias columnas como mínimo tres. En la parte izquierda se colocan las tareas por hacer, en las siguientes columnas se colocan todas las tareas en curso, en caso de que las tareas cuenten con varias fases se las puede dividir en varias columnas, finalmente se agregan las tareas ya realizadas al lado derecho.

En total se utilizan seis practicas diferentes en método Kanban:

- Visualización: Tableros con visualización general de los flujos de trabajo.
- Limitación: Cuando una tarea ya se finaliza el equipo podrá seguir con la siguiente.
- Gestión: Solventar problemas.
- Regulación: Flujos de trabajo más claros.
- Feedback: Retroalimentación para mejorar los flujos de trabajo.
- Kaizen: Siempre se trabaja para mejorar.

(IONOS Digital Guide, 2020)

Por consiguiente, la metodología XP se realiza entre dos personas y el proyecto está siendo realizado solo por una persona, por otro lado, también XP se maneja pruebas continuas haciendo el trabajo mucho más lento.

En la metodología Kanba las entregas son prontas y para su flujo de trabajo se utiliza scrum, dicho esto se toma la decisión de trabajar netamente con Scrum ya que cada ciclo de desarrollo es mucho más amplio, eficaz y se acopla mejor a un sistema electrónico ya que en cada sprint se entregará un sensor totalmente funcional.

2.3. Metodología Scrum

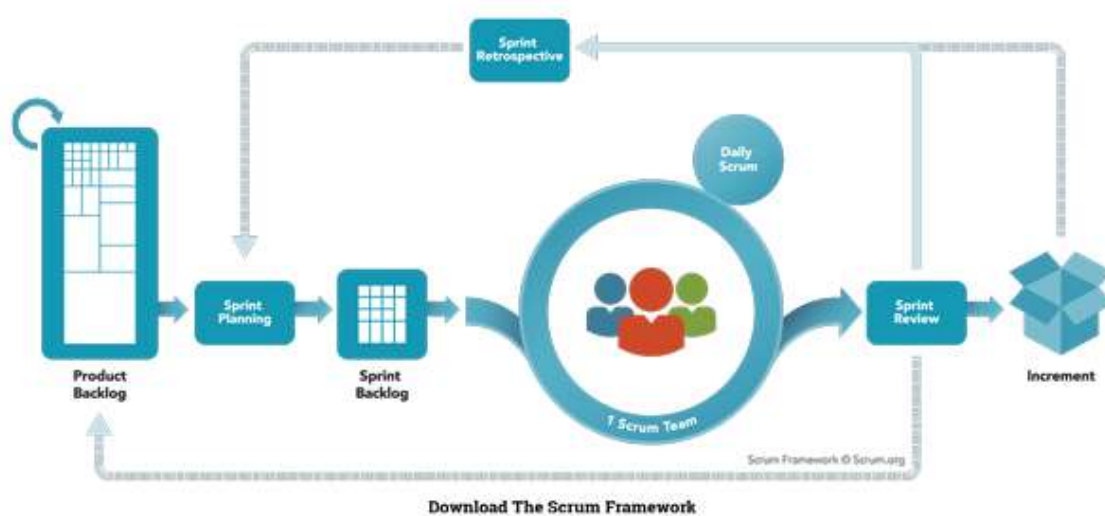


Figura 2: Metodología Scrum

Tomado de (Scrum, 2020)

Scrum es un marco de trabajo para operar de manera colectiva, esta metodología ciertamente es utilizada en proyectos complejos, teniendo como finalidad la entrega en periodos de tiempo cortos.

Como resultado scrum se basa en tres pasos fundamentales:

- Transparencia: En el proyecto scrum todos los participantes tienen el conocimiento del proyecto y en qué etapa se encuentra, a manera que se obtendrá una visión global.
- Inspección: La inspección se centra en buscar problemas en el proyecto, la organización y avance en equipo.

- Adaptación: Cuando el proyecto hay algún cambio, el equipo de desarrollo es capaz de solventarlo ofreciéndole al cliente lo que el desea.

Roles

Product Owner: Es el rol que recoge todos los requerimientos del cliente y se mantiene en contante comunicación con él.

Scrum Master: Es el encargado de que todas las etapas y técnicas de scrum sean cumplidas.

Equipo de desarrollo: Son los encargados de cumplir con todos los requerimientos del cliente y las tareas que prioriza el Product Owner.

Sprint

Sprint es básicamente el foco de todos lo hitos de Scrum, en cada interacción se realizan una serie de pasos los cuales se encuentra en un sprint. La duración de cada sprint es de un mes teniendo en cuenta que un sprint muy largo podría llegar a detener la comunicación con el cliente y poner en peligro el proyecto.

Hitos de Scrum

Los hitos contenidos en cada sprint podemos encontrar:

- Sprint Planning
- Daily Meeting
- Sprint Review
- Sprint Retrospective



Figura 3: Los hitos de la Metodología de trabajo Scrum.

Tomado de (Abellán E, 2020)

Sprint Planning: Es una reunión colectiva con el equipo de trabajo el cual propone todas las tareas que se va a realizar, esta reunión tiene una duración de 8 h para sprint de un mes.

Las preguntas que e equipo debe hacerse son:

- ✚ ¿Qué se va a hacer en el Sprint?
- ✚ ¿Cómo lo vamos a hacer?

Dairly meeting: Son reuniones diarias máximo de 15 minutos.

En esta reunión el equipo define tres preguntas:

- ✚ ¿Qué hice ayer?
- ✚ ¿Qué voy a hacer hoy?
- ✚ ¿Tengo algún problema que necesito que me soluciones?

Sprint review: Es un entregable al cliente junto con una reunión donde el cliente revisa y valida los cambios de cada sprint, ofrece un feedback sobre nuevas tareas que posteriormente se agregaran al product backlog, su duración máxima es de 4 horas para sprint de un mes.

Sprint Retrospective: Se utiliza para evaluar al equipo Scrum y mejorar para el siguiente sprint, la duración máxima de esta etapa es de 3 horas para sprint de un mes.

Herramientas de Scrum

Product Backlog: Es una serie de tareas que se debe cumplir en el proyecto, para esto es necesario la participación del Product Owner que es el encargado de recopilar y organizar los requerimientos del cliente.

Sprint backlog: Son series de tareas recopiladas en el producto backlog las cuales se eligen en el sprint planning, teniendo en cuenta que todo el equipo debe estar al tanto de todas las tareas.

(Scrum, 2020)

Basado en lo propuesto anteriormente el product backlog del sistema domótico está construido por historias de usuario, por lo que se tomara cada historia de usuario y se desarrollara en cada sprint tanto en la parte electrónica de igual modo en la página web, siendo así un sensor totalmente funcional, cumpliendo así con los objetivos de cada etapa de scrum.

2.4. Raspberry pi 3

Es un minicomputador que trabaja con batería o red, funciona con un sistema operativo ARM Linux, aunque existen diferentes versiones del dispositivo, este dispositivo puede ser utilizado en robótica, sistemas que utilicen sensores de varios indoles, como también Matlab para inteligencia artificial u otros temas.

El sistema operativo se carga y ejecuta en una tarjeta micro (SD), como también se puede manejar utilizando un teclado, mouse y pantalla o simplemente con cable de red o wifi con ssh de forma remota. (Kyaw A, Phat H, Joseph, 2017 p.287)

Sus especificaciones son:

- Procesador ARMV8 de 4 núcleos a 64 bits 1.4 GHZ
- Memoria ram de 1GB LPDDR2 SDRAM.
- Bluetooth 4.2
- Wifi 2.4 Y 5 GHZ IEEE 802.11 b/g/n/ac.
- Ethernet (300 Mbps)
- 4 puertos USB 2.0
- Puerto HDMI
- Mini USB para alimentación
- GPIO de 40 pines
- Puerto CSI para conectar cámara
- Puerto DSI para conectar una pantalla táctil
- Salida de audio estéreo y video compuesto

GPIO

Es un sócalo de multipropósito con entradas y salidas, dichas entradas y salidas pueden ser utilizadas por una variedad de funciones en algunos casos están

disponibles en todos los pines y en otros son específicos como se puede mostrar a continuación:

PWM (modulación de ancho de pulso para manejo de motores)

- Software PWM disponible en todos los pines.
- Hardware PWM disponible en GPIO 12, 13, 19

SPI (Interface serial periféricos para conexión de ciertos sensores)

- SPI0: MOSI (GPIO10); MISO (GPIO9); SCLK (GPIO11); CE0 (GPIO8), CE1 (GPIO7)
- SPI1: MOSI (GPIO20); MISO (GPIO19); SCLK (GPIO21); CE0 (GPIO18); CE1 (GPIO17); CE2 (GPIO16)

I2C (transmisión serial bidireccional de alta velocidad)

- Data: (GPIO2); Clock (GPIO3)
- EEPROM Data: (GPIO0); EEPROM Clock (GPIO1)

Serial (Transmisión y recepción de señales)

- TX (GPIO14); RX (GPIO15)

(Raspberry pi fundation, 2020)

Cabe destacar que el GPIO es igual en todas las 4 versiones del raspberry pi estándar.

2.5. Arduino

Arduino es una plataforma electrónica de código abierto, capaz de realizar varias tareas con sensores y actuadores. Arduino puede ser programado dándole instrucciones al microcontrolador a través del IDE Arduino. (Arduino, 2020)

2.6. Diferencias entre Arduino y Raspberry

Raspberry	Arduino
Maneja sistema operativo	Maneja IDE Arduino
Bus de USB	No tiene bus USB
Puerto Ethernet Y WIFI	No contiene wifi ni ethernet

Ejecución de programas más complejos pero el tiempo de programación y construcción es lento.	Limitación de programación en proyectos complejos, sin embargo, el tiempo de programación y construcción es más rápido.
Procesamiento más grande ya que cuenta con un procesador de 4 núcleos.	Cuenta con un microcontrolador de 8 bits para proyectos de menor escala.
Maneja frameworks de desarrollo ya que tiene sistema operativo.	No maneja sistema operativo por lo tanto es imposible instalar un framework.

Tabla 1.

(Diseño propio, 2020)

Debido que el Arduino no contiene sistema operativo y solo es posible utilizar sensores, mas no se puede manejar servidor con página web, lo que es requerido para el sistema domótico planteado se decide utilizar el raspberry pi 3 que cumple con las necesidades.

2.7. Reconocimiento Facial

El reconocimiento facial utiliza una técnica llamada el aprendizaje métrico profundo, el mismo que crea un vector de características con valores reales, para el sistema el vector de características consta de 128d (es decir, una lista de 128 números con valores reales) que se utiliza para cuantificar el rostro.

El entrenamiento de red se realiza utilizando trillizos:

A single 'triplet' training step:

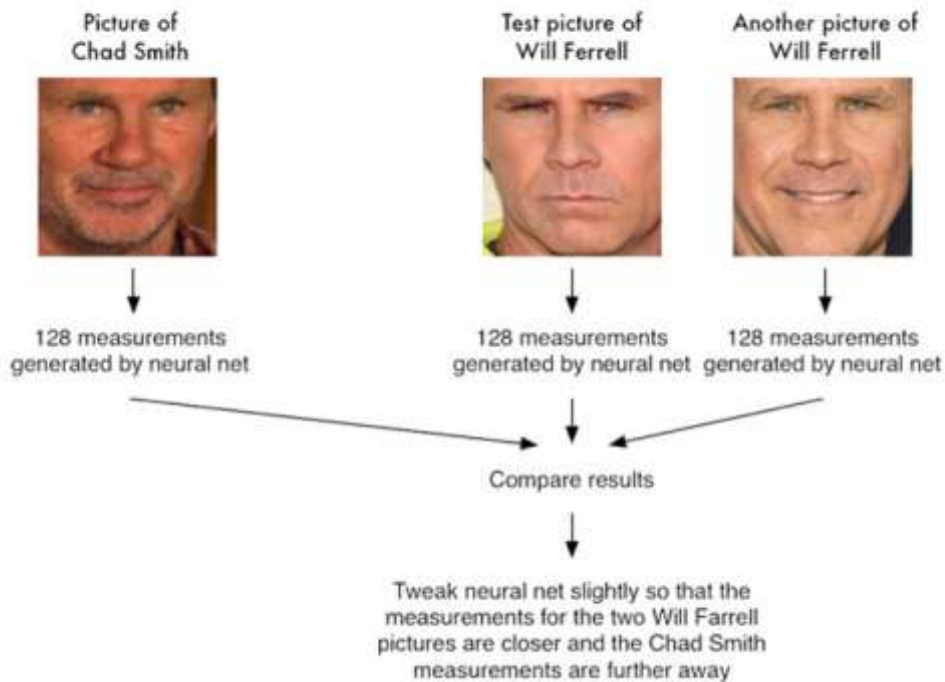


Figura 4: Modern Face Recognition with Deep Learning

Tomado de (Geitgey A, 2016)

El triplete o trillizos consta de tres imágenes con dos de la misma persona y se ajusta los pesos de la red neuronal de las dos imágenes con la tercera para acercar el vector de distancia métrica.

Como se observa las dos imágenes son de la misma persona, sin embargo, la tercera es una imagen aleatoria de otra persona, de esta manera la red neuronal se cuantifica las compara para que las dos imágenes de la misma persona tengan datos cercanos y los datos de la tercera sea diferente.

De este modo primeramente se realiza un escaneo de los rostros y permitirá realizar un reconocimiento y posteriormente abrir la puerta a la persona adecuada.

(Rosebrock A, 2018)

2.8 Sensores utilizados en el sistema domótico

2.8.1 Sensor de temperatura y humedad

Es un sensor digital de temperatura y humedad, mide el aire que circula y muestra los datos digitalmente ya que no posee salida analógica, funciona con un voltaje de 3 - 5v y posee tres pines, los cuales están distribuidos de la siguiente manera:

1. VCC positivo del sensor.
2. Datos de salida digital
3. Gnd negativo del sensor.

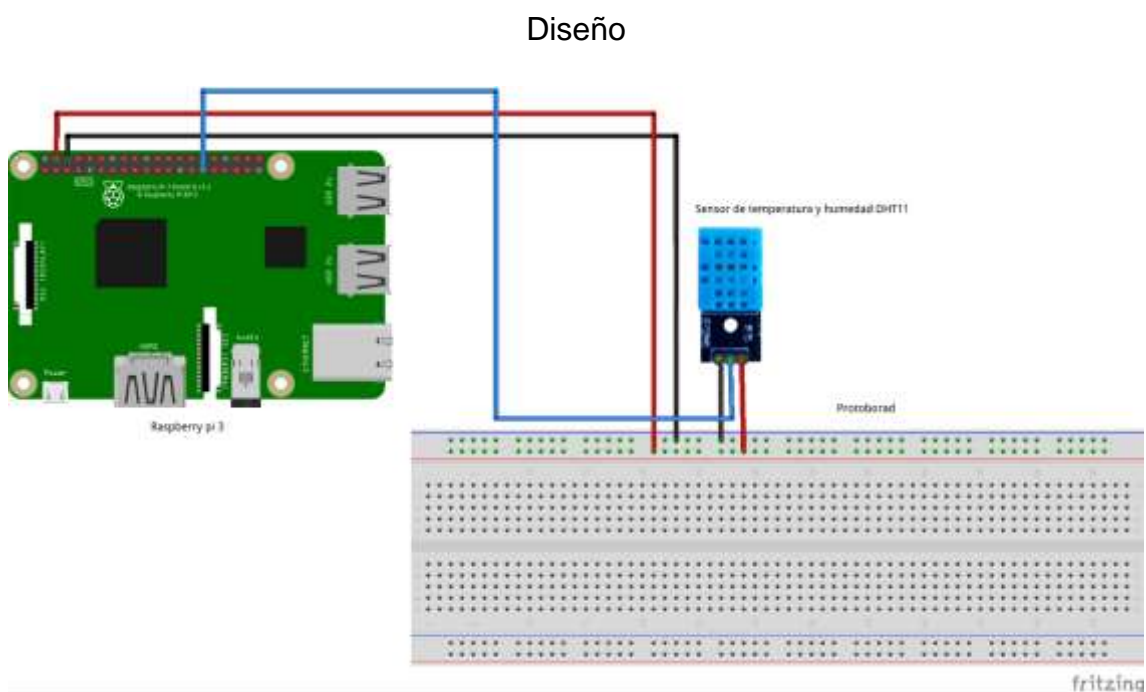


Figura 5. Conexión de modulo DHT11

Especificaciones técnicas

- Power supply: DC 3.5~5.5V
- Resolution: 16Bit
- Repeatability: $\pm 0.2^{\circ}\text{C}$
- Range: At 25°C $\pm 2^{\circ}\text{C}$
- Tiempo de respuesta: 1 / e (63%) of 25°C 6s

(AOSONG, 2018)

Con este sensor podremos medir la temperatura del ambiente y su humedad que en caso de subir la temperatura el usuario pueda tomar acciones y salir del lugar o simplemente que se active un ventilador o alarma.

2.7.2 Sensor de gas

El módulo de gas MQ6 utilizado para la detección de fugas de gas butano y gas LPG (gas doméstico), tiene una alta sensibilidad, tiempos de respuesta rápido además consta de un potenciómetro para poder ajustar su sensibilidad.

El sensor es capaz de detectar concentraciones de gas entre 300 y 1000ppm (partículas por millón).

El dispositivo posee baja conductividad cuando el ambiente esta neutro o no contiene ningún gas, por otro lado, si detecta presencia de gas LPG la conductividad de los sensores es alta, por lo tanto, se utiliza un electro circuito el cual convierte conductividad en señal de salida ya sea analógica o digital.

La conexión de sus pines son:

1. Vcc positivo del sensor
2. Aout salida analógica
3. Dout salida digital
4. GND negativo del sensor

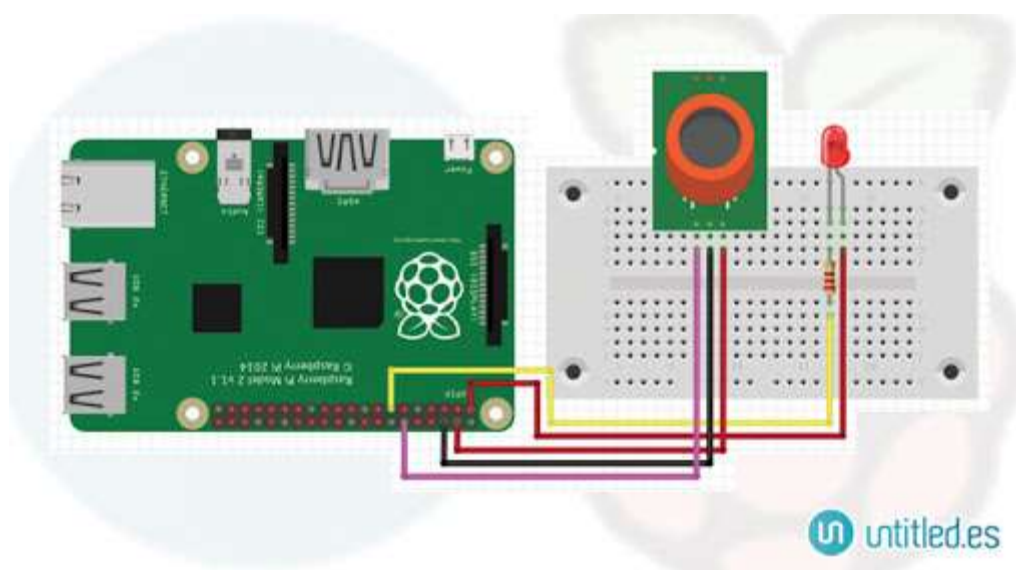


Figura 6. Esquema de conexión del módulo de gas.

Especificaciones Técnicas

- Alta sensibilidad y tiempo de respuesta rápido.
- Detecta concentraciones de gas desde 300 a 10000 ppm.
- Voltaje de entrada 5 V
- Voltaje de Salida 2.5V~4.0V (in 2000ppm C3H8)
- Dispositivo base sensor MQ6.
- Consumo: 950mW
- Temperatura 20°C±2°C ; 55%±5%RH
- Resistencia de carga ajustable.
- Tamaño: 40x20mm.
- Sensible a gases de tipo LPG

(Winsen, 2014)

El sensor de temperatura se lo utilizara para medir fugas de gas en una vivienda y si en un caso hay una fuga de gas sonara una alarma indicando que tiene que salir del lugar o cerrar la válvula de gas.

2.7.3 Actuador Relé

Es un módulo que posee dos relés capaces de soportar cargas hasta 250 a 10A, funciona como un interruptor por medio de un optoacoplador y un led que indica el estado del relé.

Este módulo activa la salida N/A (normalmente abierta) recibiendo un 0 lógico o 0 voltios y desactiva su salida con 1 lógico o 5 voltios.

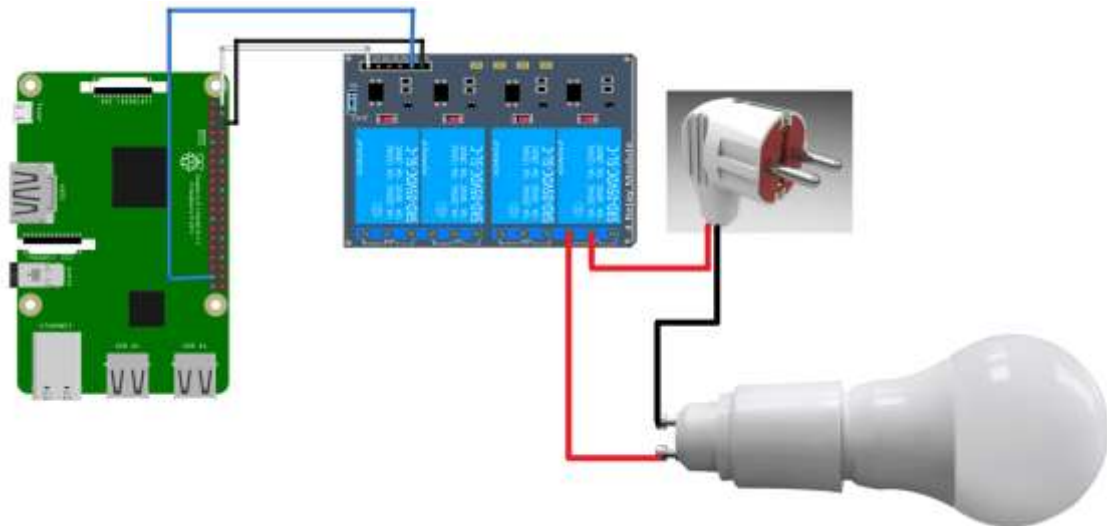


Figura 7. Esquema de conexión de módulos relé.

Especificaciones Técnicas

- Voltaje de Operación: 5V DC
- Señal de Control: TTL (3.3V o 5V)
- N° de Relays (canales): 2 CH
- Modelo Relay: SRD-05VDC-SL-C
- Capacidad máx: 10A/250VAC, 10A/30VDC
- Corriente máx: 10A (NO), 5A (NC)
- Tiempo de acción: 10 ms / 5 ms
- Para activar salida NO: 0 Voltios
- Entradas Optoacopladas
- Indicadores LED de activación´

(Winsen, 2014)

En este caso el módulo Relé será el encargado de encender una luz y abrir una chapa eléctrica para poder Ingresar a la casa.

2.8. Lenguajes de programación

2.8.1 Framework Kumbia

Es un framework para el desarrollo de páginas web, es de código abierto, proporciona facilidad de construcción para aplicaciones robustas en entornos

comerciales, es compatible con varios sistemas operativos tal es el caso de Windows, Linux, MacOs.

Las características de Kumbia son:

- Plantillas muy sencillas
- Basado en PHP
- Modelo de objetos
- Separación para programar MVC
- Soporte Ajax
- Generación de Formularios
- Seguridad

(Kumbia, 2020)

2.8.2 Lenguaje PHP

Es un lenguaje de código abierto muy adecuado para el desarrollo web, además, se puede incrustar en HTML por medio de etiquetas.

PHP sirve principalmente para programar scripts del lado del servidor, de este modo recopila datos de formularios, genera contenidos dinámicos, enviar y recibir cookies.

Campos en lo que se usa scripts en PHP:

- Scripts del lado del servidor
- Scripts de línea de comandos
- Escribir aplicaciones de escritorio

Cabe recalcar que este lenguaje de programación puede ser usado en sistemas operativos como Linux, MacOs, Windows y muchos más, como también puede funcionar en la mayoría de los servidores que contengan PHP FastCGI, lighttpd, y nginx.

En ámbitos de programación tiene la posibilidad de utilizar programación por procedimientos o programación orientada a objetos, o ambas.

Las ventajas de este lenguaje es que se puede generar html libremente, de este modo se puede crear imágenes, ficheros PDF y películas en flash. También se puede generar tipos de texto como, por ejemplo, XHTML, XML.

Dicho esto, se ha decidido trabajar con el Framework basado en PHP Kubia, ya que es fácil en su programación, adaptabilidad y que es compatible con Linux, por lo tanto, se creara una página web para manejar todos los sensores y actuadores para el sistema domótico.

(PHP, 2020)

2.8.3 Lenguaje Python

Es un lenguaje de programación interpretado, orientado a objetos. Su sintaxis es de fácil legibilidad, por lo que facilita la depuración, además, es un lenguaje de muy fácil aprendizaje.

(LUCA, 2020)

En raspberry se puede encontrar el entorno Tonny el cual utiliza RELP (Read-Evaluate-Print-Loop) donde permite ingresar código Python, contiene una ventana Shell por la misma que se obtiene los resultados del código sin un print.

Por lo antes mencionado, todas las funciones que nos ofrece Python son necesarias para la programación de cada sensor y los resultados serán mostrados en la página web.

(Raspberry, 2020)

CAPITULO III. ANALISIS Y DISEÑO

En este capítulo se muestra los requerimientos del cliente analizado previamente y se los organiza según su importancia en el proyecto, luego se detalla cada historia de usuario, dando lugar al product backlog el cual identifica cada Sprint. Además, cada Sprint contendrá un bosquejo para obtener una idea del diseño de la página web.

3.1 Diagrama del prototipo

Tomando en cuenta los requerimientos que se muestra en la figura 8, se puede observar que el Raspberry que conforma el sistema centralizado dentro el mismo, existe un servidor apache 2 con protocolo http para ingresar a la página web, por otro lado, cuenta con un diseño de hardware cada sensor y actuador por medio de un protoboard y conexiones respectivas, finalmente se puede acceder por medio de un dominio a la página web ya sea por computador o por un móvil.

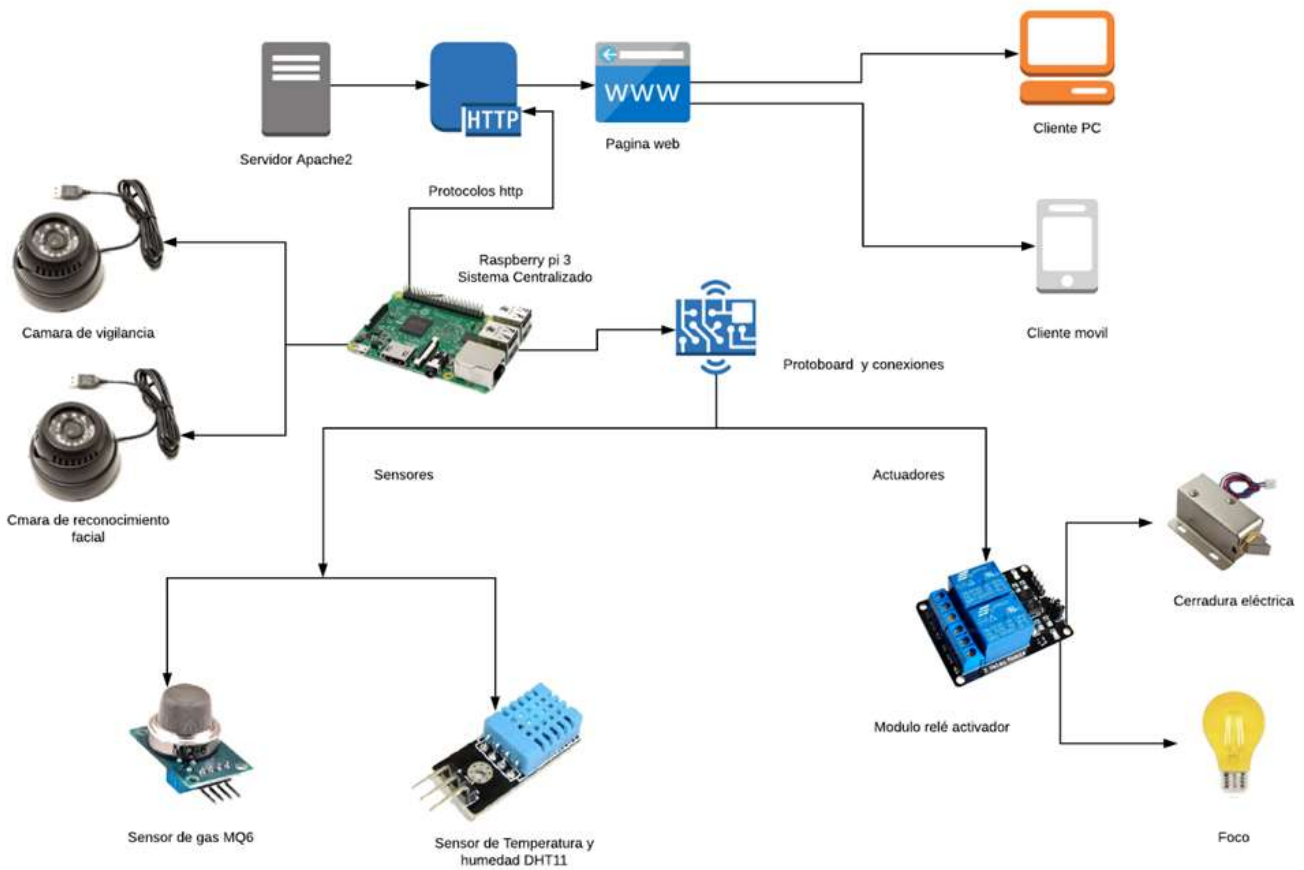


Figura 8. Esquema deseado de la solución

3.2 Planificación

En esta fase del proyecto se recoge las funcionalidades del proyecto, para luego definir las como historias de usuario que corresponden a cada una de las iteraciones del proyecto.

Lista de funcionalidades

- Creación de página web para controlar los sensores y actuadores.
- Diagrama de funcionamiento.
- Login para el ingreso a la aplicación web.
- Gestión de clientes, permitir crear, leer, actualizar y borrar.
- Autenticación como administrador para gestión de clientes.
- Ingreso como usuario para manejar los sensores y actuadores.
- Protección contra fugas de gas para prevenir incendios con la activación de alarma.
- Comprobación de temperatura y humedad para corroborar un incendio.
- Control de las luces del ingreso para visualizar el ingreso al domicilio.
- Abrir la cerradura para ingreso al hogar.
- Visualización por medio de cámaras de vigilancia de las entradas al hogar.
- Reconocimiento Facial de la entrada principal.

Se tomará en cuenta las funcionalidades de mayor importancia para definir las como historias de usuario.

Lista de historias de usuario

- Diagrama de flujo
- Gestión de usuarios
- Autenticación al sistema administrador y como usuario
- Comprobación de fuga de gas
- Control de luces
- Comprobación de temperatura y humedad
- Control apertura de cerradura
- Integración de alarma para sensor de gas
- Visualización a través de cámaras de vigilancia.
- Reconocimiento facial

3.2 Lista de Producto (Product Backlog)

El Product Owner organiza el Sprint Planning para definir las tareas al equipo Scrum, además, el equipo determina cada Sprint y los elementos para el próximo.

El Product Backlog consta de ciertas tareas importantes para el desarrollo que son las siguientes:

- El Product Backlog necesita una comunicación con el cliente constantemente.
- El Product Backlog se prioriza y ordena según la importancia que desee el cliente.
- Se estima todas las tareas del Product Backlog.

3.3 Planning Poker

Es una estimación para todo tipo de proyectos ágiles, la cual permite realizar una estimación inicial del proyecto de forma rápida, la técnica puede realizar de varias formas ya sea por cartas o una aplicación llamada Planning Poker, en cualquiera de los dos casos se utiliza una escala de números (1,2,3,5,8,13,20,40 y 100) que son de cantidad relativa y necesaria para la implementación.

En este caso en concreto se procedió a utilizar la serie hasta 8, siendo 8 las tareas más complejas y 1 las tareas más simples.

(Laurie W, 2010, p.1)

Ya enlistadas las historias de usuario se genera una tabla, en la cual se muestra el detalle de cada historia de usuario con su estimación y la prioridad que tiene, como también como se define cada Sprint como se muestra en la tabla 2.

Tabla 2.

Número	Historia de usuario	Sprint	Estimación	Prioridad
1	Diagrama de flujo	0	1	Alta
2	Gestión de usuarios	1	1	Alta
3	Autenticación al sistema administrador y como usuario.	1	1	Alta

4	Comprobación fugas de gas.	2	5	Alta
5	Control de luces.	2	2	Media
6	Comprobación de temperatura y humedad.	3	5	Media
7	Control de apertura de cerradura.	3	2	Media
8	Integración de alarma para sensor de gas.	4	2	Alta
9	Visualización a través de cámaras de vigilancia.	4	5	Media
11	Reconocimiento facial.	0	8	Alta
Total			32	

Tabla 2. Product Backlog.

3.4 Duración y Velocidad

La duración y velocidad del proyecto es de 10 semanas, de las cuales cada dos semanas se presenta el cumplimiento de cada sprint. Cabe mencionar que, mientras se encuentre en desarrollo del Sprint existe la posibilidad de cambios o mejoras de este modo, las iteraciones variaran, dando lugar a que el proyecto no debe presentar cambios significativos.

CAPITULO IV DISEÑO CONSTRUCCION

Para la construcción del proyecto se utiliza un gráfico el cual es la base de todo lo que se va a construir se inicia con un Sprint inicial el cual es la creación de login y administración de usuarios, ya que son parte fundamental para el proyecto seguido de los demás Sprint.

Los Sprints son detallados de la siguiente manera:

- Id de la historia de usuario
- Nombre de la historia de usuario
- El usuario que participara en el sistema

- El usuario responsable del desarrollo

Para la descripción se divide en tres partes esenciales que son:

- Cómo quiero realizar la historia de usuario
- Que quiero hacer con la historia de usuario
- Para que quiero realizar la historia de usuario.

Dicho esto, la historia de usuario debe responder a estas preguntas en la descripción.

- Y finalmente las tareas que va a cumplir la historia de usuario.

Cabe destacar que la historia de usuario con la estimación más alta serán las primeras en realizarse y las que tiene una estimación más baja serán ordenadas para cumplir los tiempos estimados.

4.1 Implementación

Para la duración del proyecto se utiliza una herramienta llamada Trello, en la que se definen las historias de usuario que correspondan a cada Sprint, además esta herramienta tiene tres fases que son:

1. Lista de Tarea Iniciales
2. En proceso
3. Hecho

Como se aprecia cada fase se cambia hasta llegar a hecho en la cual el Sprint acaba.

4.2 Sprint 0

Como sprint inicial se realizará un diagrama de flujo, de este modo se muestra los procesos de todo el sistema y las tareas que realiza.

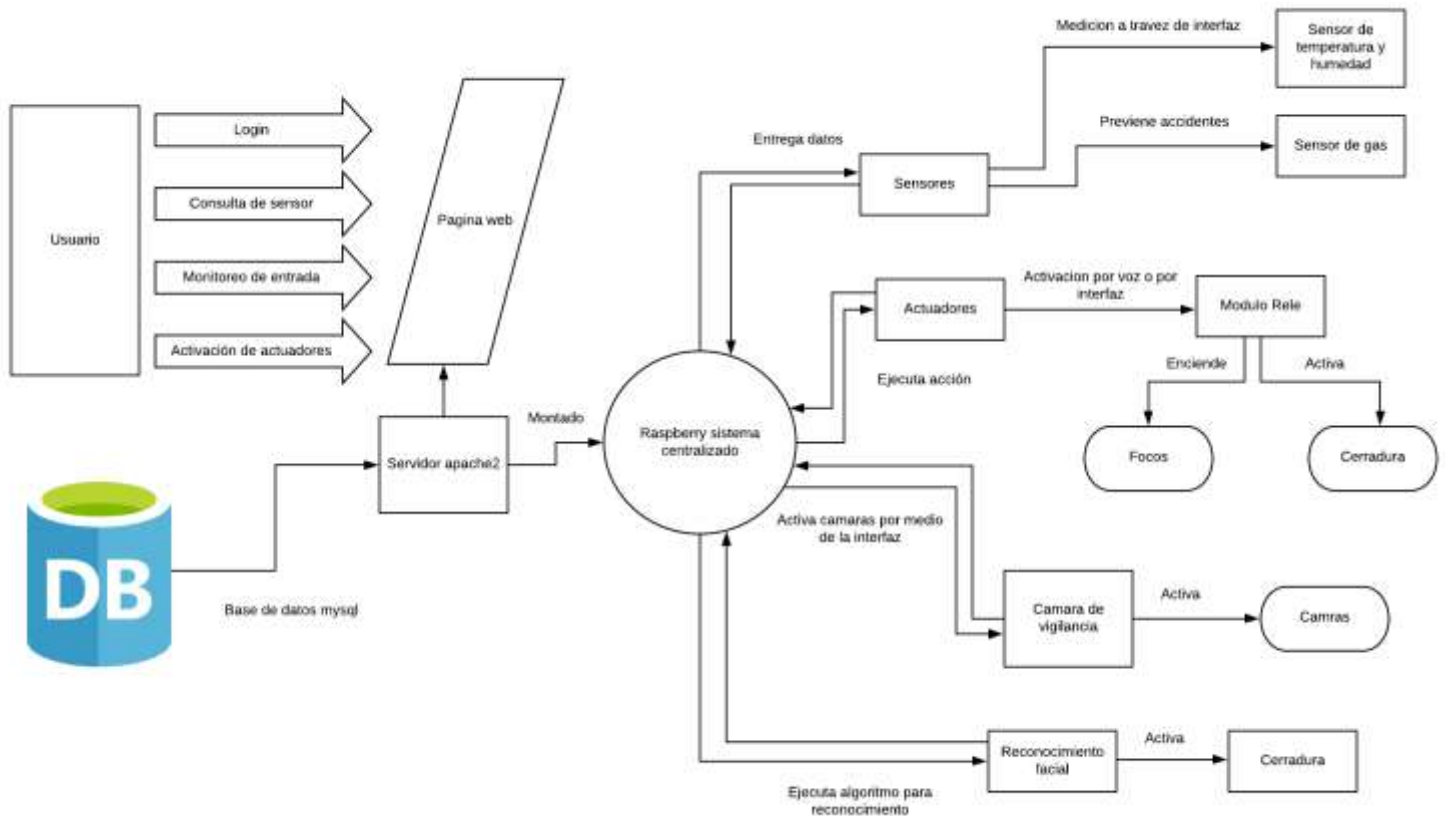


Figura 9.- Diagrama de flujo de datos.

El gráfico de la figura 9 muestra el funcionamiento del prototipo, de este modo se ve que el raspberry pi 3 es un sistema centralizado en el cual se encuentra instalado un servidor apache 2 y en el mismo una página web.

En la página web se encontrará un login con una base de datos que sirve como acceso ya sea como administrador o usuario, en el caso de los usuarios se permitirá manejar los diferentes botones para accionar los actuadores, sensores y Cámara.

4.3 Sprint 1

Como sprint 1 se selecciona gestión de clientes ya que se necesita una administración y login de cada usuario para controlar los sensores y actuadores, de este modo se ha seleccionado dos historias de usuario.

Tabla 3.

Sprint 1			
Id	Nombre de la historia de usuario	Prioridad	Estimación
HU01	Gestión de usuarios	Alta	1
HU02	Autenticación al sistema administrador y como usuario	Alta	1

Tabla 3: Sprint 1.

Historia de usuario 1.

Historia de Usuario
Id: HU01
Usuario: Administrador
Nombre de Historia: Gestión de usuarios
Responsable: David López
Estimación:3
Descripción: Como administrador quiero crear, eliminar, actualizar un usuario para el sistema y otorgarle un rol específico para Ingresar al sistema.
Tareas: <ul style="list-style-type: none"> • Creación de login • Creación del módulo para gestionar los clientes • Creación del botón para ingresar un cliente en el sistema • Creación de los campos necesarios para ingresar el nuevo cliente • Creación el botón para guardar los datos del cliente ingresado • Mostrar la opción para consultar la información de un cliente • Mostrar la opción para actualizar la información de un cliente • Mostrar la opción para eliminar la información de un cliente

Tabla 4: Historia de usuario de gestión de clientes

Historia de usuario 2.

Historia de Usuario
Id: HU02
Usuario: Administrador y cliente
Nombre de Historia: Autenticación al sistema administrador y como usuario.
Responsable: David López
Estimación: 5
Descripción: Como administrador quiero acceder al sistema para poder visualizar usuarios, gestionar usuarios y asignar roles en la página web. Como cliente quiero acceder al sistema para visualizar interactuar con los sensores, actuadores, cámaras, control por comando y reconocimiento facial.
Tareas: <ul style="list-style-type: none"> • Creación de la página principal • Contratación de hosting temporal • Codificar el login para el ingreso al sistema • Crear el menú de administrador • Asignación de roles • Asignación de contraseñas • Crear página independiente para usuario.

Tabla 5: Historia de usuario de autenticación del sistema como administrador.

Se muestra bosquejo de la interfaz gráfica de la siguiente manera.

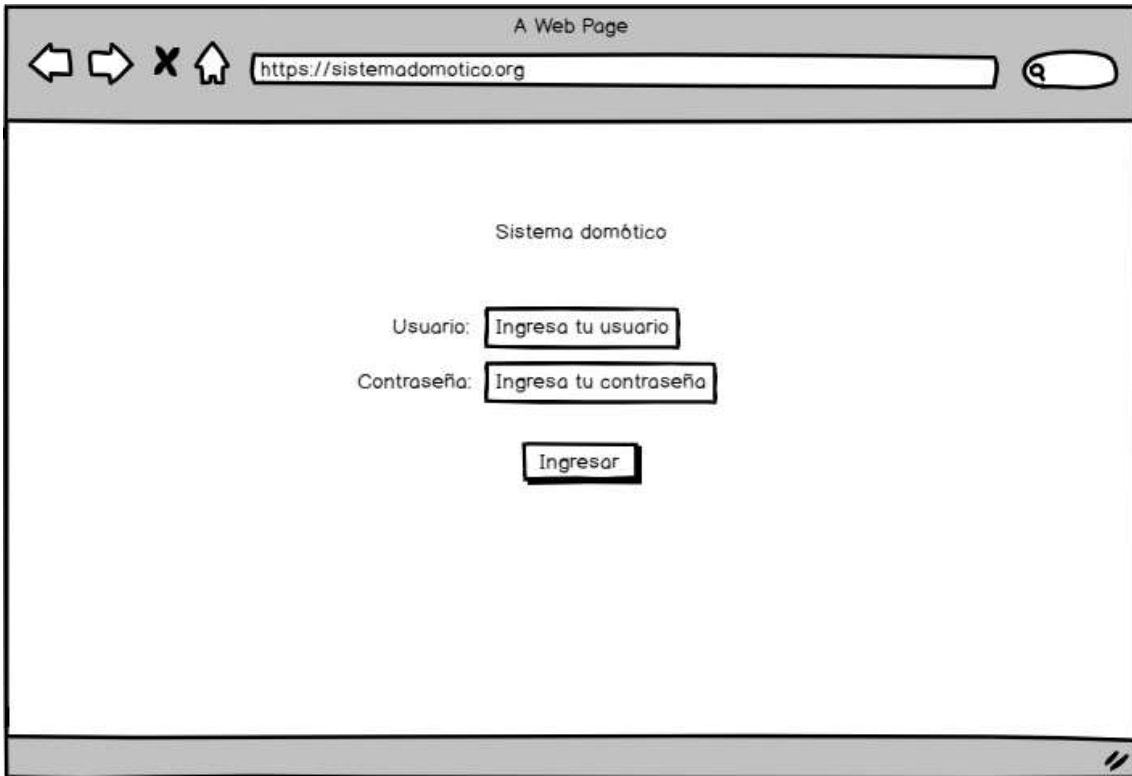


Figura 9. Bosquejo de login.

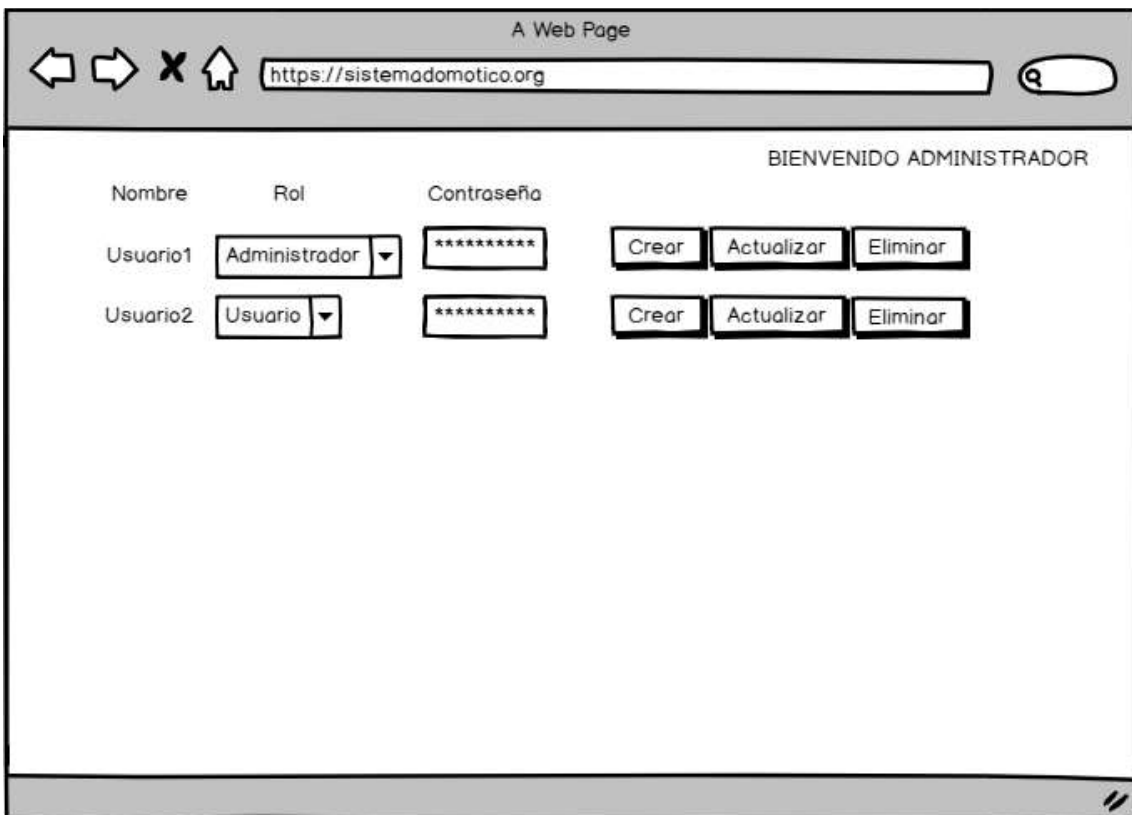


Figura 10. Bosquejo de gestión de usuarios.

4.3.1 Riesgos

En el primer Sprint se tendrá una gestión de usuario y una administración, por lo tanto, se tendrá un manejo de framework adecuado y las herramientas adecuadas.

Los riesgos que se pueden presentar son:

- Inconvenientes de compatibilidad del framework.
- Desarrollo del proyecto lento.

Para no cometer estos errores se debe leer la documentación y observar si el framework y las herramientas son compatibles con el sistema operativo instalado.

4.3.2 Estado del Proyecto

Al utilizar la metodología SCRUM luego de desarrollar cada Sprint, se presenta un entregable, en este caso el entregable permitirá verificar errores y nuevas funcionalidades.

Las funcionalidades nuevas son las siguientes:

- Permitir cambiar contraseña a cada usuario como administrador.
- Al crear un nuevo usuario agregar un enlace para regresar al listado de usuarios creados y visualizar el nuevo usuario.

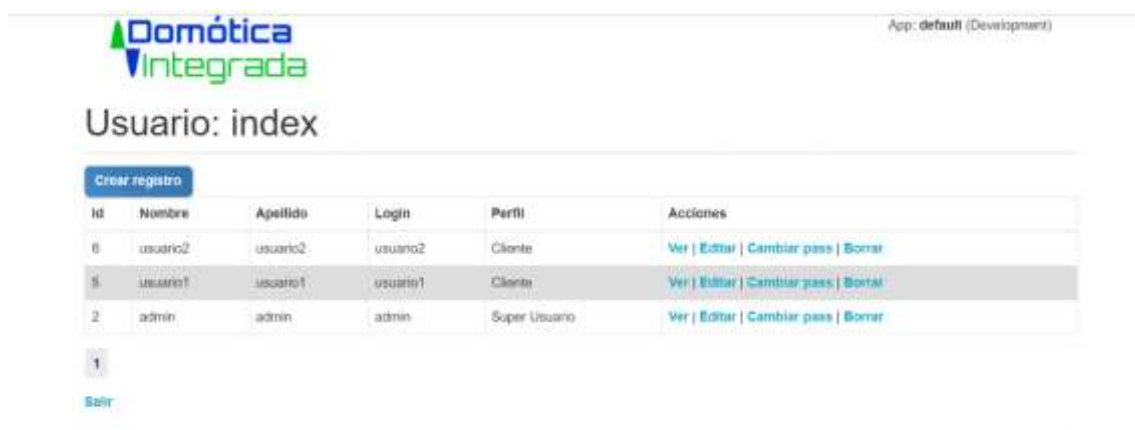


Figura 11. Estado del proyecto Sprint 1.

Usuario: crear

Nombre *

Apellido *

Login *

Password *

Perfil *

Seleccione

Figura 12. Estado del proyecto Sprint 1.

4.3.3 Resultados

El uso del framework kumbiaphp ayudo a realizar el primer Sprint y entregable funcionar del proyecto, por lo que hizo posible mantener una página web, en la cual se controla todos los dispositivos conectados al raspberry, juntamente con un servidor apache y una base de datos para guardar la información de cada uno de los usuarios.

Cabe destacar que el Sprint se completó con éxito y en el tiempo adecuado, tal como se muestra a continuación en la siguiente imagen.



Domótica Integrada

App: default (Development)

Usuario: crear

Nombre *

Apellido *

Login *

Password *

Perfil *

Seleccione

Figura 13. Registro de usuario.

En la segunda historia de usuario se muestra la autenticación con roles como administrador y usuario, de esta forma al ingresar como administrador permite la gestión de usuarios y como usuario permite el control de cada actuador y sensor.

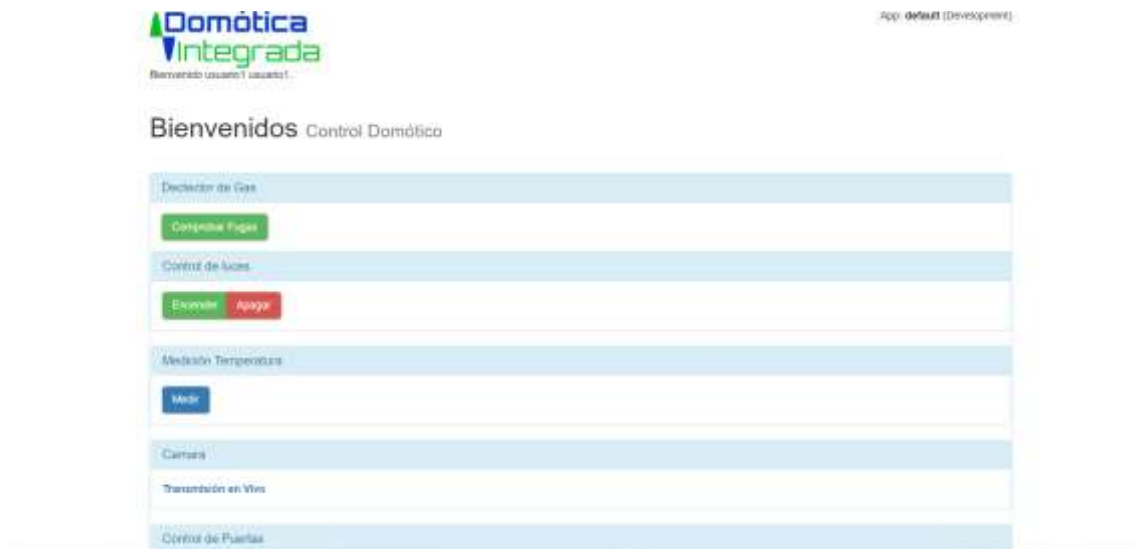


Figura 14. Ingreso como usuario.



Figura 15. Ingreso como administrador.

Completando las tareas del Sprint se pasa las tareas completadas utilizando la herramienta trello.

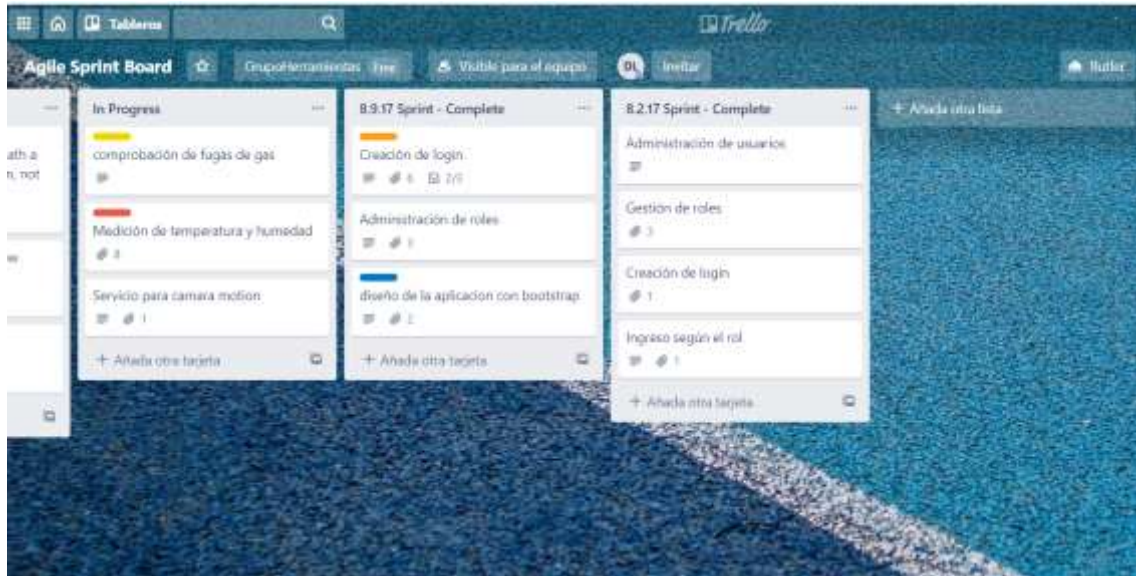


Figura 16. Tareas completadas.

Ya actualizada la tabla en trello se procede actualizar los puntos ganados en el sprint backlog.

Sprint backlog.

Numero	Historia de usuario	Sprint	Esfuerzo	Puntos Ganados
1	Administración de usuarios	1	1	1
2	Autenticación al sistema administrador y como usuario	1	1	1

Tabla 6. Sprint backlog del primer sprint.

4.3.4 Retrospectiva

El desarrollo del primer Sprint no duro mucho tiempo, ya que es un crud con el framework kumbiaphp y sus respectivos roles para ingresos a diferentes ambientes, además cabe mencionar que se utiliza una base de datos para este sprint.

Que funciona bien

El crud funciona perfectamente como también el ingreso como usuario o como administrador.

Que se debe mejorar

El diseño del login y la interfaz deben ser mejores sin embargo tomaría mas tiempo de desarrollo.

Que se aprendió

El framework utilizado es muy intuitivo y fácil de usar sin embargo requiere de un conocimiento extenso para programar en este.

Que problemas se tuvo

Al momento de validar datos se encontró la forma de validar que no se ingrese el mismo usuario, sin embargo, no se puede controlar con un mensaje en pantalla.

4.4. Sprint 2

Para el sprint 2 se seleccionado dos historias de usuario la primera, para prevenir fugas de gas para esto el sensor detectara si hay presencia de gas o no, como segundo lugar se procede a configurar el rele, para que se enciendan las luces al momento de presionar el botón en la interfaz gráfica de la página web.

Sprint 2.

Sprint 2			
Id	Nombre de la historia de usuario	Prioridad	Estimación
HU03	Comprobación de fuga de gas	Alta	8
HU04	Control de luces	Alta	8

Tabla 7: Sprint 2.

Historia de usuario 3.

Historia de Usuario	
Id: HU03	Usuario: Cliente
Nombre de Historia: Comprobación de fuga de gas	
Responsable: David López	Estimación: 3
Descripción: Como cliente, quiero ingresar a la página web, para poder verificar si existen fugas de gas en el sector de la cocina para evitar accidentes.	
Tareas: <ul style="list-style-type: none"> • Crear la interfaz para visualizar los datos del sensor de gas • Crear un botón para poder comprobar fugas de gas en tiempo real • Codificar las sentencias necesarias en Python para obtener los datos emitidos por el sensor de gas hacia el dispositivo Raspberry • Enviar los datos obtenidos hacia la página web mediante programación en PHP 	

Tabla 8: Historia de usuario 3.

Historia de usuario 4.

Historia de Usuario	
Id: HU04	
Usuario: Cliente	
Nombre de Historia: Control de luces	
Responsable: David López	Estimación: 8
Descripción: Como cliente, quiero ingresar a la página web, para poder controlar las luces de la entrada principal de la vivienda.	

Tareas:

- Crear la interfaz para visualizar el control de las luces
- Crear un botón para el encendido de las luces
- Crear un botón para el apagado de las luces
- Codificar las sentencias necesarias en un script para controlar las entradas y salidas del dispositivo Raspberry hacia el módulo de relés conectado
 - Codificar las sentencias necesarias en Python para obtener las señales emitidas por los relés
 - Enviar los datos obtenidos hacia la página web mediante programación en PHP

Tabla 9: Historia de usuario 3.

El bosquejo se observa que el resultado del sensor es normal o existencia de gas, esto sucede cuando se presiona el botón de la interfaz.

La interfaz gráfica del control de luces muestra los estados de encendido o apagado.

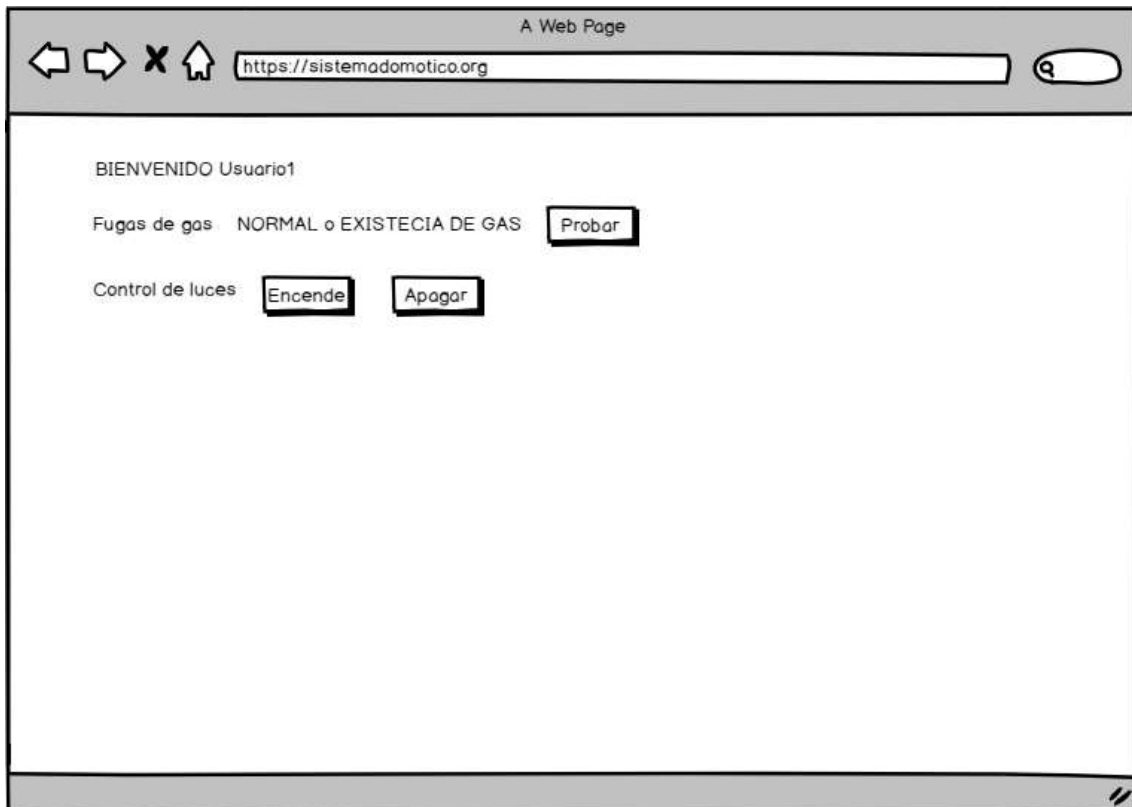


Figura 17: Bosquejo de fugas de gas y control de luces.

4.4.1 Riesgos

El desarrollo del segundo Sprint requiere de un conocimiento básico de electrónica, Python y del framework kumbiaphp, ya que todas estas herramientas serán necesarias para el desarrollo de las dos historias de usuario.

Es necesario que los dos sensores funciones completamente y sean fáciles de usar para el cliente.

4.4.2 Estado del Proyecto

Al finalizar el Sprint 2 se presenta el entregable, de este modo se observa el sistema de fugas de gas con el sensor respectivo y el control de luces con su actuador.

4.4.3 Resultados

La tercera historia de usuario consta de un sensor funcional programado en Python, el cual puede medir si existe existencia de gas, dicha función se ejecutará al presionar el botón comprobar fugas que se encuentra en una página web alojada en un servidor apache en el raspberry.



Figura 18. Comprobar fugas de gas.

La cuarta historia de usuario consta de control de luces a través de un módulo relé conectado a un raspberry, el actuador será accionado por la pagina web, para encender se presiona el botón (encender) y para apagar el botón (apagar), cada acción muestra un mensaje en el estado que se encuentra.

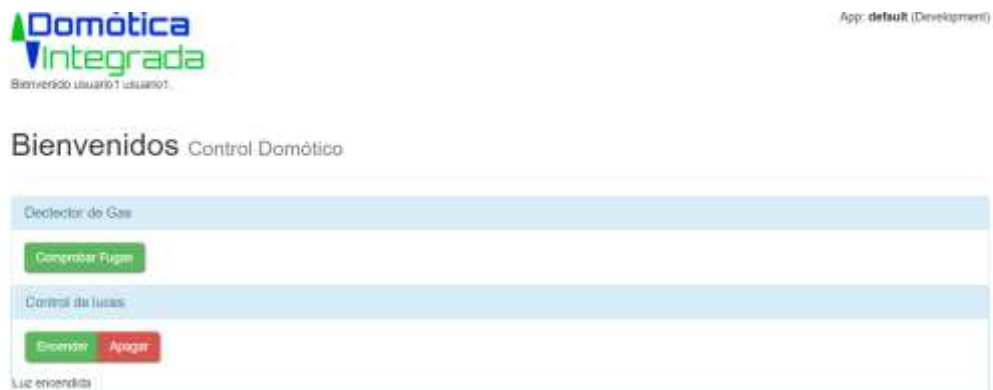


Figura 19. Control de luces.

Ya terminadas las tareas se procede a la actualización de las tareas en trello, pasando de en proceso ha finalizado.

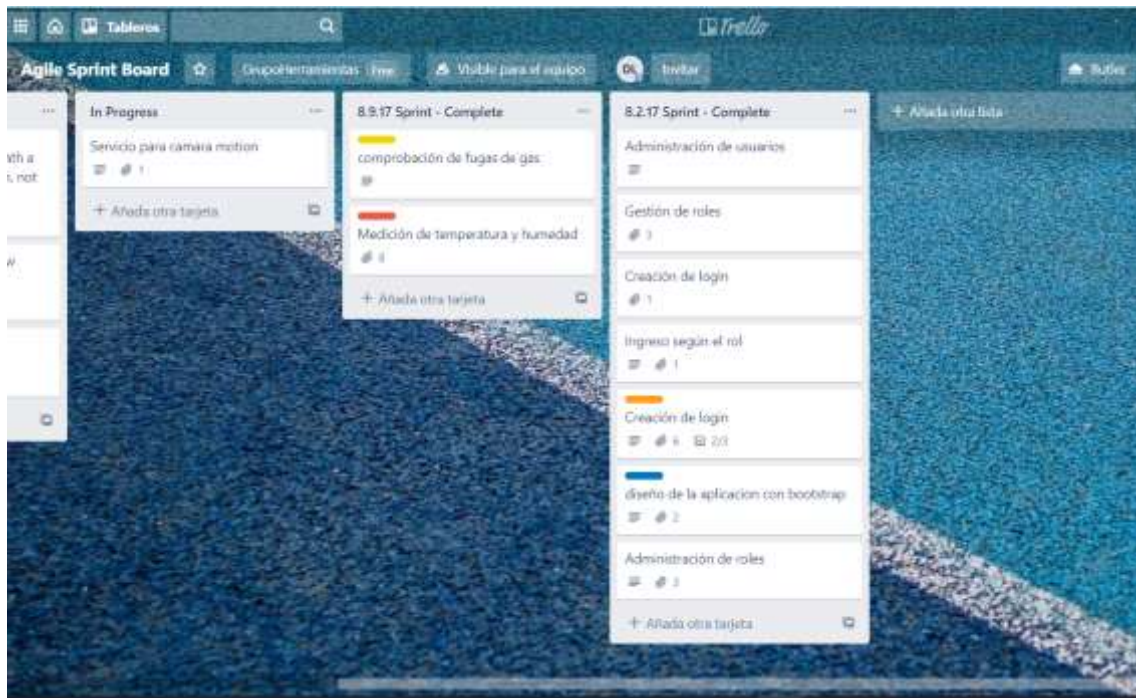


Figura 20. Actualización de trello Sprint 2

Se procede a la actualización del sprint Backlog 2.

Sprint Backlog 2.

Numero	Historia de usuario	Sprint	Esfuerzo	Puntos Ganados
3	Comprobación de fugas de gas.	2	3	1
4	Medición de temperatura y humedad.	2	3	1

Tabla 10. Sprint backlog 2.

Velocidad de desarrollo

El tiempo que duro fue un poco mas de lo esperado ya que se tuvo que montar un servidor apache y configurar archivos de entorno para ejecutar desde la página web.

4.4.4 Retrospectiva

El desarrollo del Sprint 2 duro mas tiempo de lo pensado sin embargo se logro completar el Sprint y se tuvo experiencia con comandos de sistema Linux, como también en las herramientas de Python para accionar o medir los sensores o actuadores.

Que funcionó bien

La aplicación funciono de manera optima ya que el caso de la segunda historia de usuario de utilizo comandos Shell.

Que se debe mejorar

Se puede guardar los datos del sensor de gas para tener valores continuamente cuando se requiera.

Que se aprendió

Se aprendió a tomar datos de una plataforma a otra que en este caso fue de Python a php.

Que problemas impiden el progreso

Los problemas fueron en la configuración de los archivos de sistema para ejecución de los comandos en la aplicación web.

4.5 Sprint 3

Para el Sprint 3 se ha seleccionado dos historias de usuario, la prioridad es alta ya que se trata de un sistema para asegurar un accidente y el control de apertura de cerradura para el ingreso al hogar.

Sprint 3.

Sprint 3			
Id	Nombre de la historia de usuario	Prioridad	Estimación
HU05	Comprobación de temperatura y humedad	Alta	5
HU06	Control apertura de cerradura	Alta	3

Tabla 11: Sprint 3.

Historia de usuario 5.

Historia de Usuario
Id: HU05
Usuario: Cliente
Nombre de Historia: Comprobación de temperatura y humedad
Responsable: David López
Estimación: 5
Descripción: Como cliente, quiero ingresar a la página web, para poder visualizar la temperatura y humedad en la cocina y prevenir accidentes.
Tareas: <ul style="list-style-type: none">• Crear la interfaz para visualizar los datos del sensor de temperatura y humedad• Crear un botón para comprobar el estado de la temperatura y humedad• Descargar la librería adecuada para obtener los datos emitidos por el sensor de temperatura y humedad en el dispositivo Raspberry• Codificar las sentencias necesarias en Python para obtener la información emitida por el sensor• Enviar los datos obtenidos hacia la página web mediante programación en PHP

Tabla 12: Historia de usuario 5.

Historia de usuario 6.

Historia de Usuario
Id: HU06
Usuario: Cliente
Nombre de Historia: Control apertura de cerradura.

Responsable: David López
Estimación: 13
Descripción: Como cliente, quiero ingresar a la página web, para poder abrir la cerradura de la puerta principal de la vivienda.
Tareas: <ul style="list-style-type: none"> • Crear la interfaz para visualizar el control que permitirá la apertura de la cerradura de la puerta principal. • Crear un botón para abrir la puerta. • En la interfaz permitirá ver si la puerta se abrió. • Crear y mostrar el estado de la puerta.

Tabla 13: Historia de usuario 6.

El bosquejo se muestra de la siguiente manera.

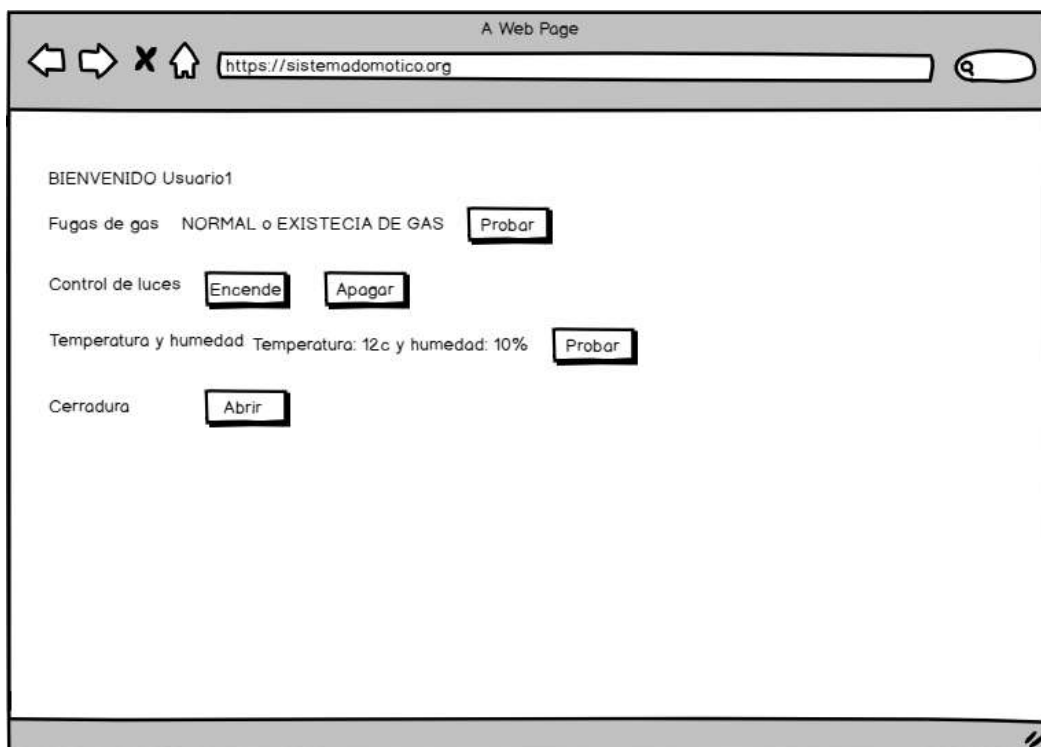


Figura 21: Bosquejo de Temperatura y humedad con apertura de cerradura.

4.5.1 Riesgos

Para el desarrollo del tercer sprint se puede tener varios riesgos, como en caso del sensor de temperatura y humedad que necesita de una librería para interactuar, sin embargo, para la apertura de puerta se reutilizara el código del control de luces.

4.5.2 Estado del proyecto

Ya finalizado el Sprint3 se presenta el entregable en el cual se observa la temperatura y humedad al presionar el botón respectivo y se puede accionar la cerradura eléctrica de igual manera al presionar el botón en la pagina web.

4.5.3 Resultados

En la quinta historia de usuario se mide la temperatura y humedad de la vivienda, todo esto es posible con un sensor de temperatura y humedad, como también con una librería que permite convertir datos análogos en porcentaje de temperatura y humedad, de la misma manera esto se puede ejecutar directamente de la página web.

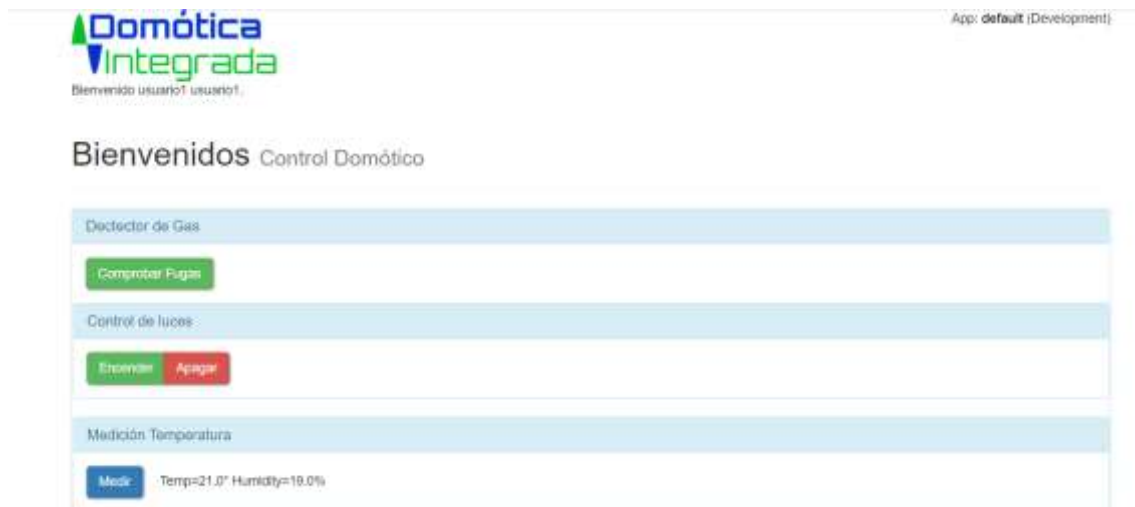


Figura 22. Medición de temperatura y humedad.

La sexta historia de usuario se trata de un sistema de apertura de una cerradura eléctrica que se abrirá desde la aplicación web presionando el botón abrir y mostrará el estado de esta.

Bienvenidos Control Domótico

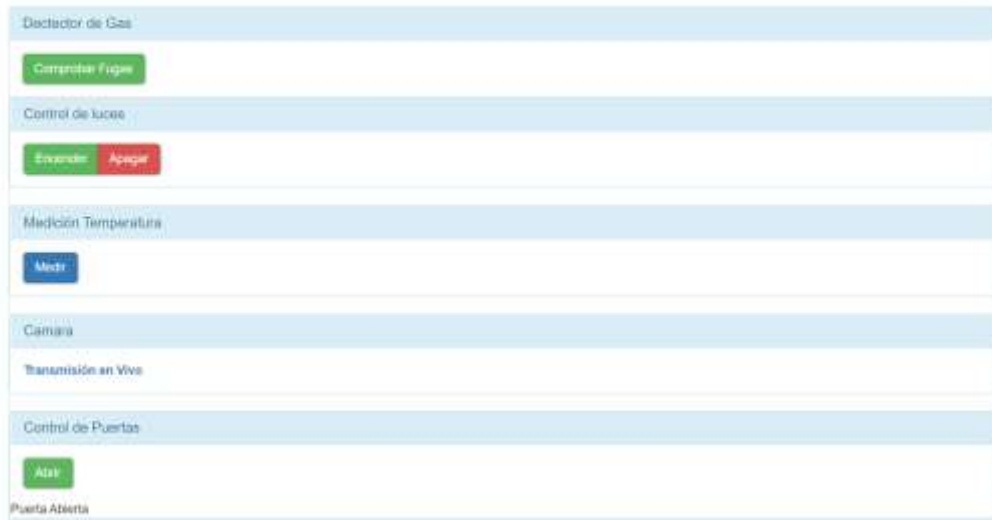


Figura 23. Apertura de cerradura.

Ya completadas las tareas del Sprint3 se procede actualizar en la herramienta trello.

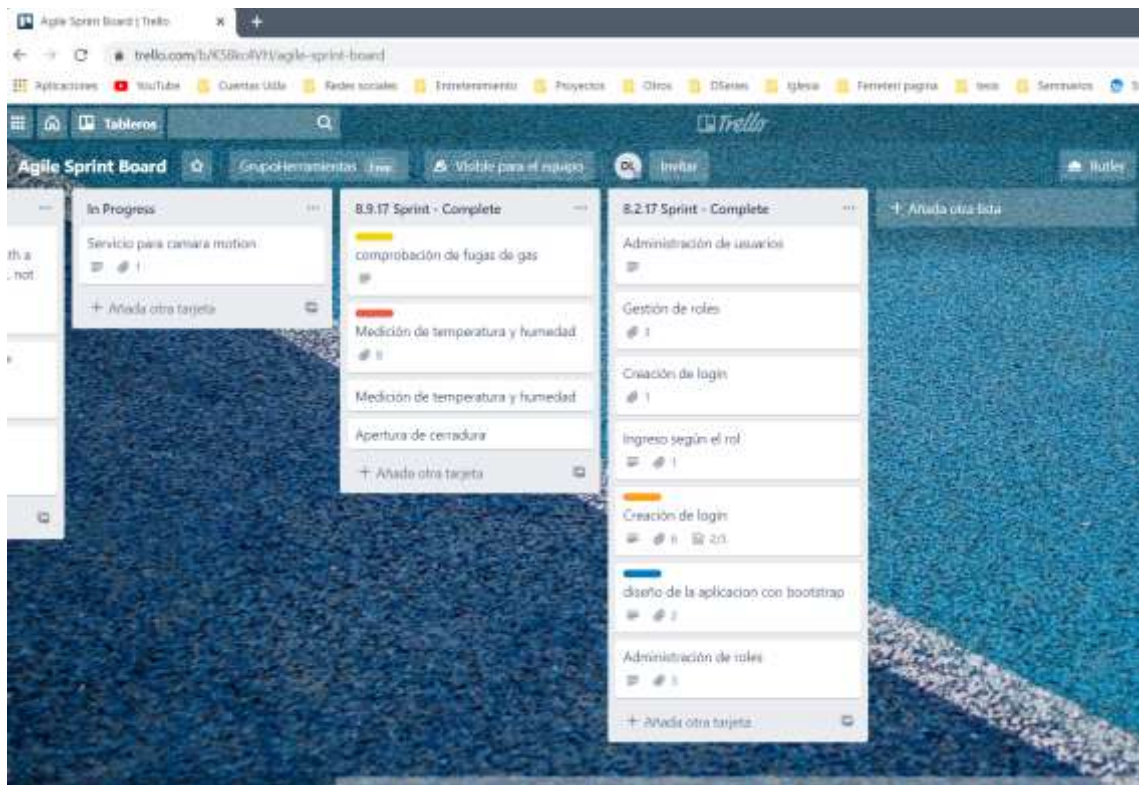


Figura 24. Actualización de trello Sprint 3.

Ya actualizado en la herramienta trello se procede a actualizar los puntos obtenidos en el Sprint Backlog 3.

Sprint Backlog 3.

Numero	Historia de usuario	Sprint	Esfuerzo	Puntos Ganados
5	Medición de temperatura y humedad.	3	3	1
6	Apertura de cerradura.	3	1	1

Tabla 14. Sprint backlog 3

Velocidad de desarrollo

El desarrollo del tercer sprint no se tuvo inconvenientes, ya que se reutilizo código para la apertura de puerta y para la medición de temperatura y humedad se utiliza una librería, de modo que se terminó antes de lo determinado.

4.5.4 Retrospectiva

El desarrollo del tercer Sprint no tomo mucho tiempo ya que la librería y la reutilización de código ayudo a facilitar la programación.

Que funciono bien

Tanto el sistema para medir la temperatura y humedad como el sistema de apertura de puerta funcionaron eficientemente.

Que se debe mejorar

El sensor de temperatura y humedad puede utilizar base de datos y mejorar su funcionalidad midiendo en todo tiempo que se le requiera.

Que se aprendió

Al ejecutar una librería para obtener datos de un sensor.

Que problemas impiden el progreso

No se identificó un problema en el desarrollo de este sprint.

4.6 Sprint 4

Ya que el sensor de gas no contiene ninguna alerta se ha implementado una alarma cuando exista una fuga de gas, por otro lado, las cámaras de vigilancia mantienen un ambiente seguro y tiene la posibilidad de visualizar quien entra y sale.

Sprint 4.

Sprint 4			
Id	Nombre de la historia de usuario	Prioridad	Estimación
HU07	Integración de alarma para el sensor de gas	Alta	5
HU08	Visualización de cámara de vigilancia	Alta	8

Tabla 15: Sprint 4.

Historia de usuario 7.

Historia de Usuario
Id: HU07 Usuario: Cliente
Nombre de Historia: Integración de alarma para el sensor del gas
Responsable: David López
Estimación: 8
Descripción: Como cliente, quiero ingresar a la página web y poder escuchar la alarma emitida por la sirena al detectar presencia de gas en niveles altos.
Tareas: <ul style="list-style-type: none">• Codificar la sentencia condicional que permita detectar la presencia de gas• Codificar las sentencias para determinar los niveles altos de gas y emitir una alarma o alerta.

Tabla 16: Historia de usuario 7.

Historia de usuario 8.

Historia de Usuario	
Id: HU08	
Usuario: Cliente	
Nombre de Historia: Visualización de cámara de vigilancia	
Responsable: David	
López	Estimación: 8
Descripción: Como cliente, quiero ingresar a la página web, para poder visualizar la cámara web.	
Tareas: <ul style="list-style-type: none">• Crear la interfaz para visualizar la cámara hacia los sensores• Instalar paquete motion en el sistema para realizar la conexión entre la cámara y el Raspberry• Configurar las sentencias necesarias para visualizar la cámara• Enviar la transmisión de la cámara hacia la página web	

Tabla 17: Historia de usuario 8.

El bosquejo muestra como se ve la cámara en vivo, en este caso no existe un bosquejo de la integración de la alarma.

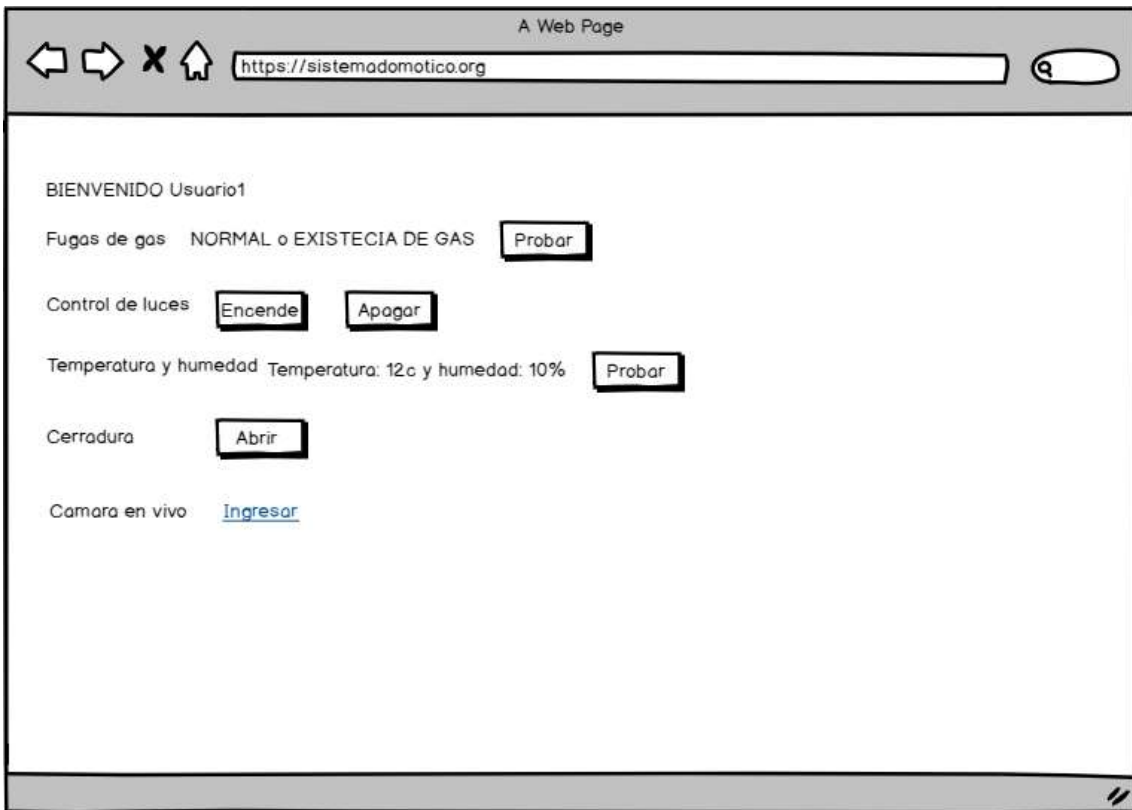


Figura 25: Bosquejo de cámara en vivo.



Figura 26: Bosquejo de cámara en vivo segunda pantalla.

4.6.1 Riesgos

En esta parte del proyecto se debe conocer los comandos por consola Linux que se requieran para la instalación del motion que es servidor para cámaras que estén conectadas al raspberry, este programa puede no instalarse si no se actualizan todas las librerías del sistema operativo Linux.

Por lo tanto, se actualizará todos los paquetes y librerías de Linux.

4.6.2 Estado del proyecto

Se presenta el respectivo entregable en el cual se muestra que cuando existe fuga de gas comienza a sonar la alarma, por otro lado en caso que se requiera ver a los visitantes de la vivienda poder hacer un clic en la página web y visualizar.

4.6.3 Resultados

Para la séptima historia de usuario se puede visualizar imagen a través de una cámara usb conectada al raspberry, lo cual permitirá vigilar la entrada a la vivienda, de este modo tener mayor seguridad, en la página web se encuentra un link que lleva a visualizar en tiempo real.

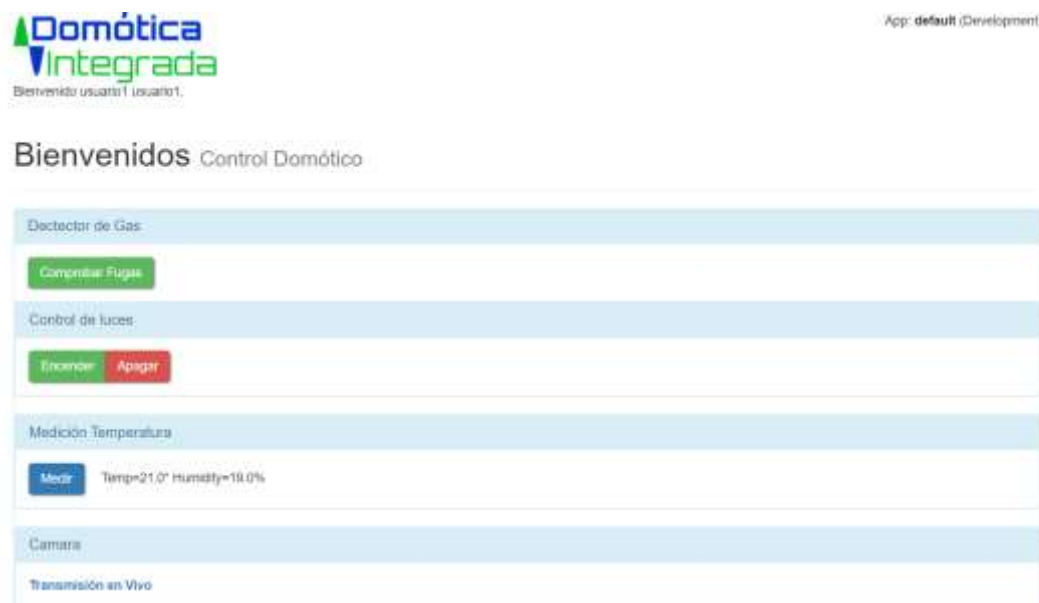


Figura 27. Enlace para transmisión en vivo.

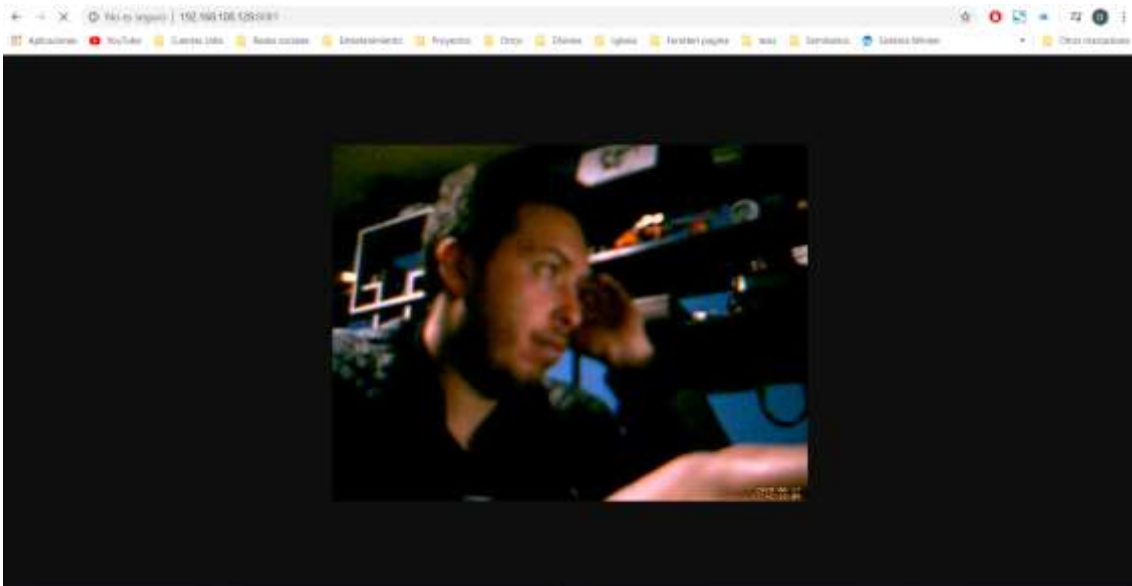


Figura 28. Visualización e Cámara.

Luego de completar las tareas se procede a actualizar en la herramienta trello, colocando en estado hecho.

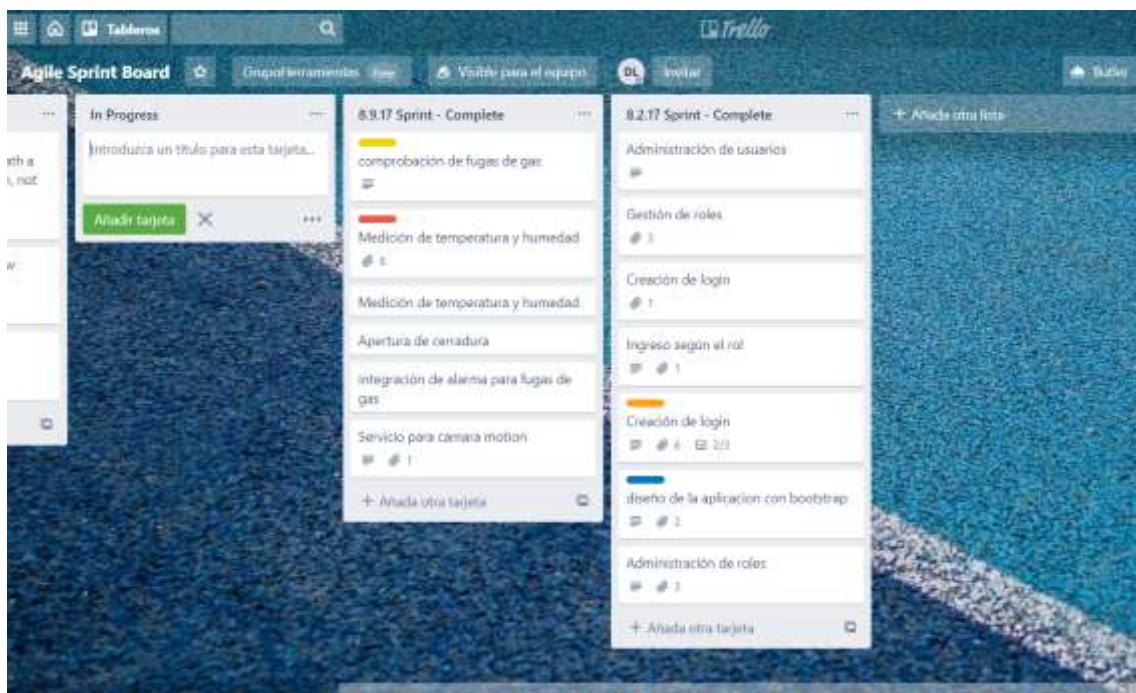


Figura 29. Actualización de trello sprint 4.

Luego se procede a actualizar los puntos ganados en el producto backlog 4.
Sprint Backlog 4.

Numero	Historia de usuario	Sprint	Esfuerzo	Puntos Ganados
9	Integración de alarma para sensor de gas	4	1	1
10	Visualización a través de cámaras de vigilancia.	4	3	1

Tabla 18. Sprint backlog 4

Velocidad de desarrollo

La velocidad fue considerable y se obtuvo los resultados antes del tiempo establecido.

4.6.4 Retrospectiva

El desarrollo de este cuarto Sprint fue tomo poco tiempo y fue muy sencillo de realizar ya que solo se necesitaba instalar por consola en el caso de la Cámara y de un bucle para el caso de la alarma.

Que funcione bien

Las dos historias de usuario fueron exitosas y se agregaron a la correctamente en la página web.

Que se mejora

En el caso de la alarma seria una buena practica agregar un desbloqueo por clave, como también implementar una alerta por correo.

Que se aprendió

Como utilizar comandos en Linux para instalación de librerías y paquetes.

Que impide el progreso

No hubo ninguna acción que impida el progreso, ya que son tareas muy simples.

4.7 Sprint 5

Finalmente, para el ultimo Sprint se elige las partes fundamentales para el sistema que son el reconocimiento facial para reconocer quien ingresa al domicilio.

Sprint 5.

Sprint 5			
Id	Nombre de la historia de usuario	Prioridad	Estimación
HU10	Reconocimiento facial.	Alta	9

Tabla 19: Sprint 5.

Historia de usuario 9.

Historia de Usuario
Id: HU11
Usuario: Cliente
Nombre de Historia: Reconocimiento facial.
Responsable: David López
Estimación: 3
Descripción: Como cliente, quiero tener seguridad en mi hogar por medio de reconocimiento facial para los dueños de la vivienda y en caso de que sean reconocidos puedan entrar.
Tareas: <ul style="list-style-type: none"> • Conexión necesaria para el módulo de facial. • Configurar la biblioteca para el reconocimiento facial. • Codificar la cámara para reconocimiento facial.

Tabla 20: Historia de usuario 9.

Para el bosquejo del reconocimiento facial se ha tomado imágenes de muestra de cómo funciona el sistema.

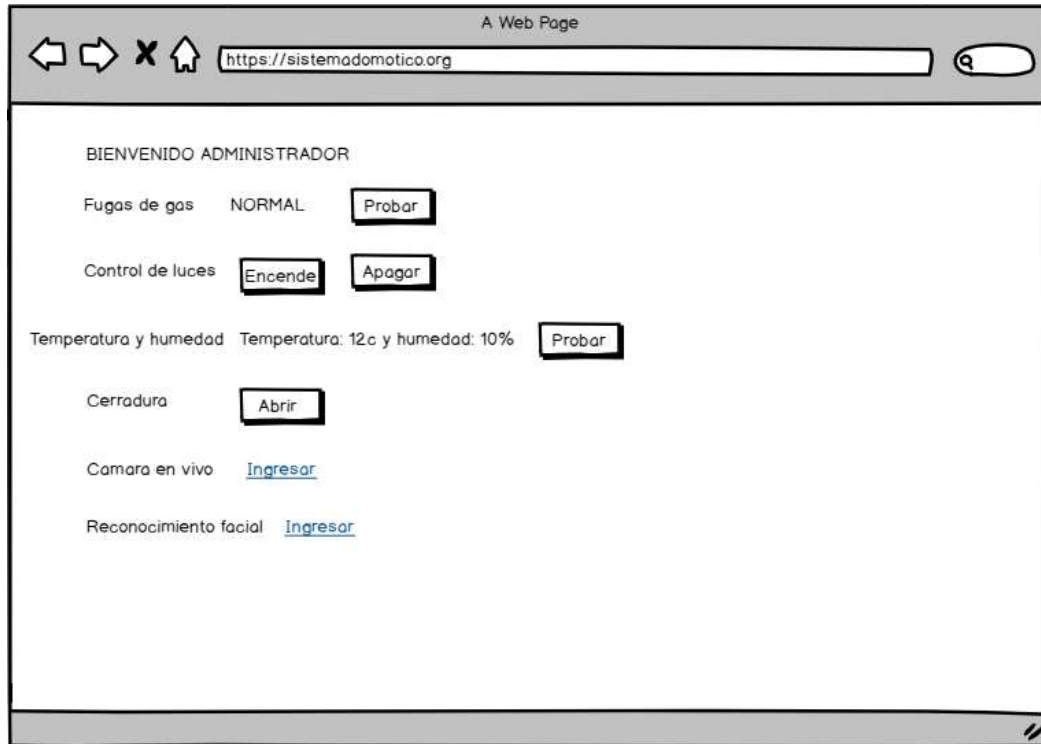


Figura 30: Bosquejo de reconocimiento facial pantalla principal.

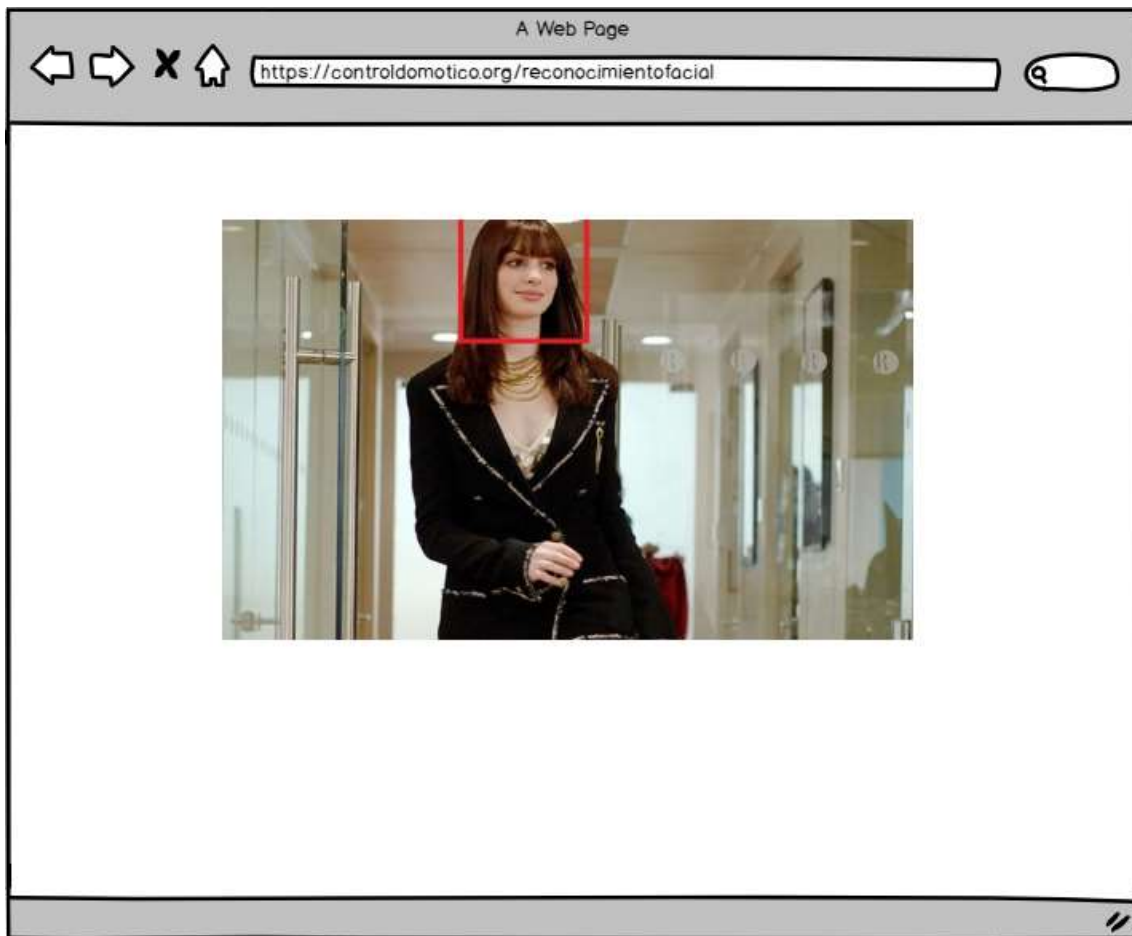


Figura 31: Bosquejo de reconocimiento facial segunda pantalla.

4.7.1 Riesgos

En el quinto Sprint no se tiene el conocimiento del tema y se realizó una investigación para poder entender como funciona y como se implementará, por lo tanto, el riesgo es un poco elevado ya que es un tema nuevo.

4.7.2 Estado del Proyecto

Ya finalizado el sprint 5 se presenta los entregables donde se obtuvieron los resultados del reconocimiento facial el cual muestra el nombre de la persona que se encuentra en la entrada de la vivienda y le da acceso a la misma.

4.7.3 Resultados

La séptima historia de usuario consta del reconocimiento facial, el cual además de reconocer el rostro mediante la Cámara del raspberry permitirá acceder al domicilio, para todo esto existe una librería llamada opencv que realiza algoritmos para reconocimiento, dicho esto hay tres etapas que son:

- La captura del rostro: La librería guarda 30 imágenes del rostro que se requiera localmente en el servidor.
- Entrenamiento del sistema: Para el reconocimiento la librería crea dos archivos los cuales tendrán datos de comparación.
- Reconocimiento y accionamiento: Mediante los archivos creados el sistema bota datos de parentesco y si son mayores a 70% accionara la puerta y permitirá el Ingreso.

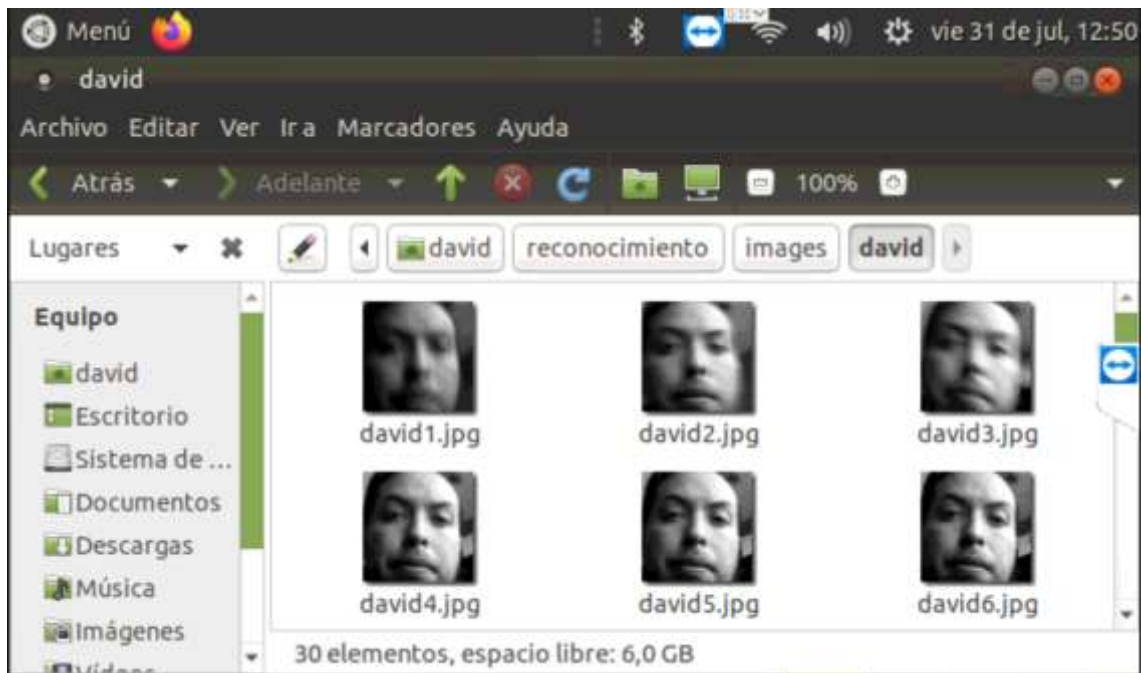


Figura 32. Capturas del rostro.

```
• david@david-desktop: ~/reconocimiento
Archivo Editar Ver Buscar Terminal Ayuda
david@david-desktop:~/reconocimiento$ sudo python3 facetrain.py
/home/david/reconocimiento/images ['david'] []
/home/david/reconocimiento/images/david [] ['david14.jpg', 'david2.jpg',
'david30.jpg', 'david26.jpg', 'david12.jpg', 'david28.jpg', 'david17.jpg',
'david22.jpg', 'david29.jpg', 'david19.jpg', 'david23.jpg', 'david11.jpg',
'david16.jpg', 'david18.jpg', 'david8.jpg', 'david13.jpg', 'david4.jpg',
'david21.jpg', 'david20.jpg', 'david24.jpg', 'david25.jpg', 'david5.jpg',
'david15.jpg', 'david3.jpg', 'david27.jpg', 'david1.jpg', 'david10.jpg',
'david7.jpg', 'david6.jpg', 'david9.jpg']
david14.jpg
david
{'david': 1}
david2.jpg
david
david30.jpg
david
```

Figura 33. Entrenamiento del sistema.



Figura 34. Creación de archivos de entrenamiento.

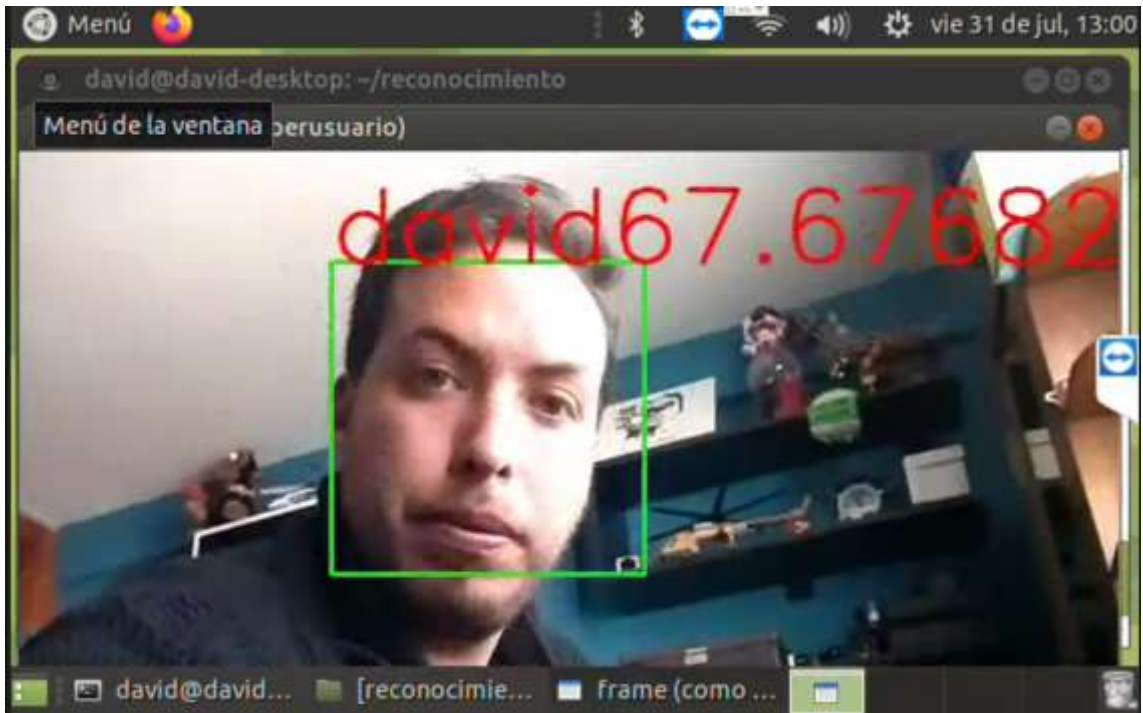


Figura 35. Reconocimiento facial.

Ya completado la tarea del Sprint se procede a colocar las tareas finalizadas en la herramienta trello.

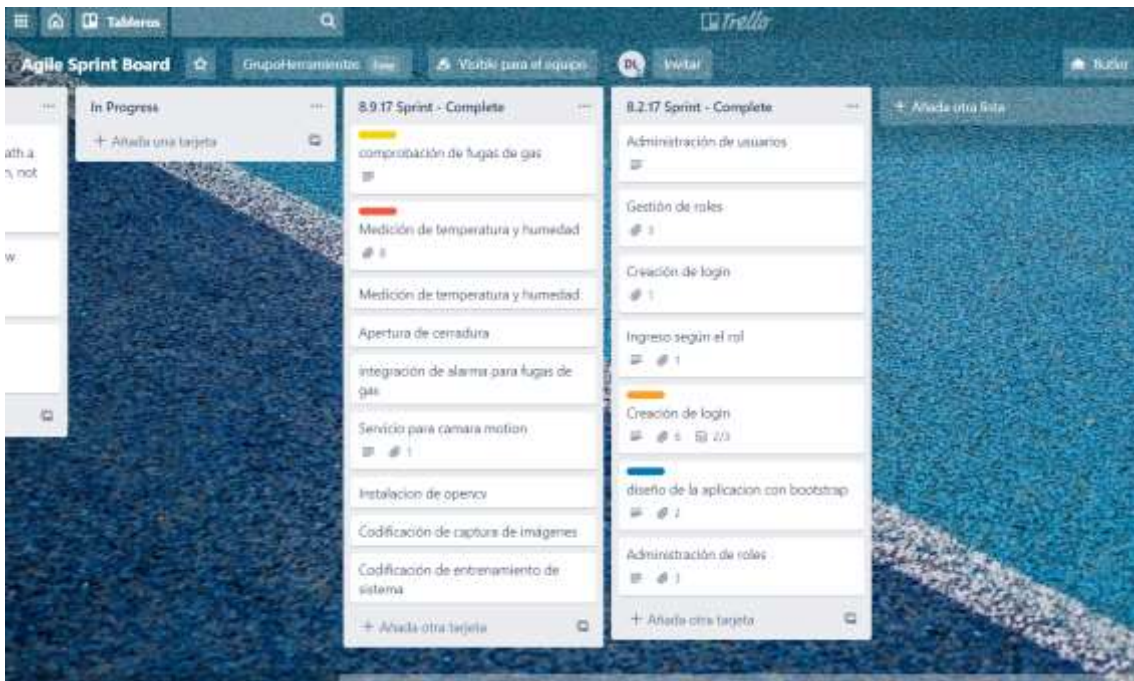


Figura 36. Actualización de trello sprint 5.

Ya actualizadas las tareas en trello se procede a actualizar los puntos en Sprint backlog 5.

Sprint Backlog 5.

Numero	Historia de usuario	Sprint	Esfuerzo	Puntos Ganados
11	Integración de alarma para sensor de gas	5	5	1

Tabla 21. Sprint backlog 5

Velocidad de desarrollo

La velocidad fue demorada sin embargo se pudo realizar cumpliendo con el objetivo propuesto.

4.7.4 Retrospectiva

El desarrollo del Sprint 5 como se tenía planeado fue el que más tiempo tomó en desarrollarse, ya que se tuvo que hacer una investigación profunda del tema, como también se tuvo varios conflictos en la instalación de las librerías de Python debido a su incompatibilidad con el sistema.

Que funcionó bien

El reconocimiento facial funcionó perfectamente desde la consola, tanto la toma de capturas de imágenes como también el entrenamiento del sistema para finalmente pueda reconocer el rostro y permitir el acceso automáticamente.

Que se debe mejorar

Se debe incluir el reconocimiento en la página web para poder ejecutarla desde esta.

Que se aprendió

Existen herramientas más completas que evitan hacer un código en PHP y recogen los datos directamente desde Python mejorando funcionalidades.

Que problemas impiden el progreso

Los códigos de Python no se pueden recoger en php a pesar de que las otras funcionalidades se pudieron realizar, en este caso el código es muy largo impidiendo recoger los datos en una variable.

CAPITULO V CASOS DE PRUEBA

Los casos de prueba es un conjunto de acciones para identificar posibles errores, como también mejorar una funcionalidad que se encuentre defectuosa para esto, se ha llevado a cabo una tabla de caso de prueba por cada historia de usuario.

5.1 Caso de prueba historia de usuario 1

Caso de prueba 1.

Caso de prueba	
Código: 1	HU: Gestión de Usuarios
Nombre: Gestion de Usuarios y login	
Descripción: Permite ingresar al sistema con una clave y contraseña y gestionar usuarios.	
Condiciones de ejecución: Esto se llevará a cabo cuando se ingrese el usuario y contraseña correcta, como también al hacer clic en cada link se puede realizar cada tarea específica.	
Entrada/ Pasos de ejecución:	Respuesta del sistema:
Ingresar al dominio local	Ingresar al dominio correctamente
Ingresar a la aplicación	Se despliega la pantalla del login
Ingresar usuario	Se observa los caracteres ingresados
Ingresar contraseña	Se observa los caracteres ingresados de manera oculta
Clic en el botón ingresar	Se despliega la administración de usuarios

Clic en botón “Crear registro”	Se despliega una página en la cual se Ingresar el nombre y apellido de usuario, nombre para Ingreso, Perfil y contraseña.
Clic en el link “ver”	Se despliega una página en la cual se observa la información del usuario.
Clic en el link “Editar”	Se despliega una página en la cual edita los datos del usuario.
Clic en el link” Cambiar password”	Se despliega una página en la cual se puede modificar la clave.
Clic en el link “Eliminar”	Se despliega mensaje “estas seguro” y muestra opción de cancelar y aceptar, en caso de seleccionar aceptar se elimina automáticamente.

Tabla 22: Caso de prueba 1.

En la pantalla de inicio solo se muestra dos textbox para ingresar usuario, contraseña y un botón para el ingreso.



Figura 37: Pantalla de Inicio de Login.

En la pantalla de inicio para administración se muestra un nombre para login el cual es necesario para el inicio de sesión.

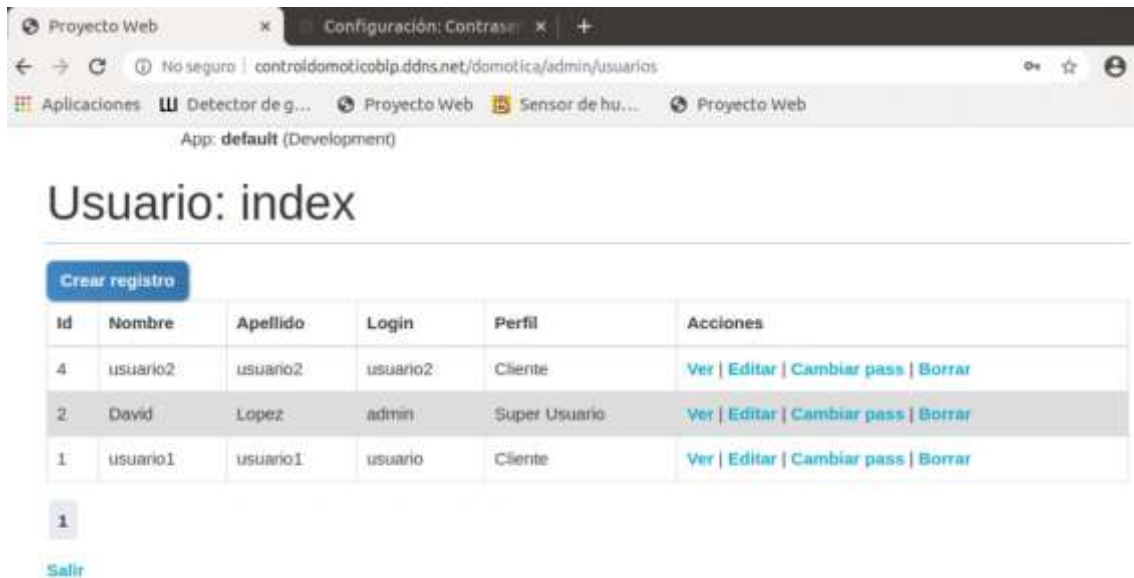


Figura 38: Listado de usuarios y administración.



Figura 39: Creación de usuarios

Caso de prueba 2.

Caso de prueba	
Código: 2	HU: Autenticación al sistema como administrador y usuario.
Nombre: Ingreso como administrador y usuario.	
Descripción: Permitirá que el usuario o administrador ingrese respectivamente a su página.	
Condiciones de ejecución: Esto se llevará a cabo cuando se ingrese con usuario y contraseña.	
Entrada/ Pasos de ejecución:	Respuesta del sistema:
Ingresar a la aplicación	Se despliega la pantalla de login
Ingresar como administrador	Se despliega la página para gestión de usuarios.
Ingresar como usuarios	Se despliega la página para manejar los sensores, actuadores etc.

Tabla 23: Caso de prueba 2.

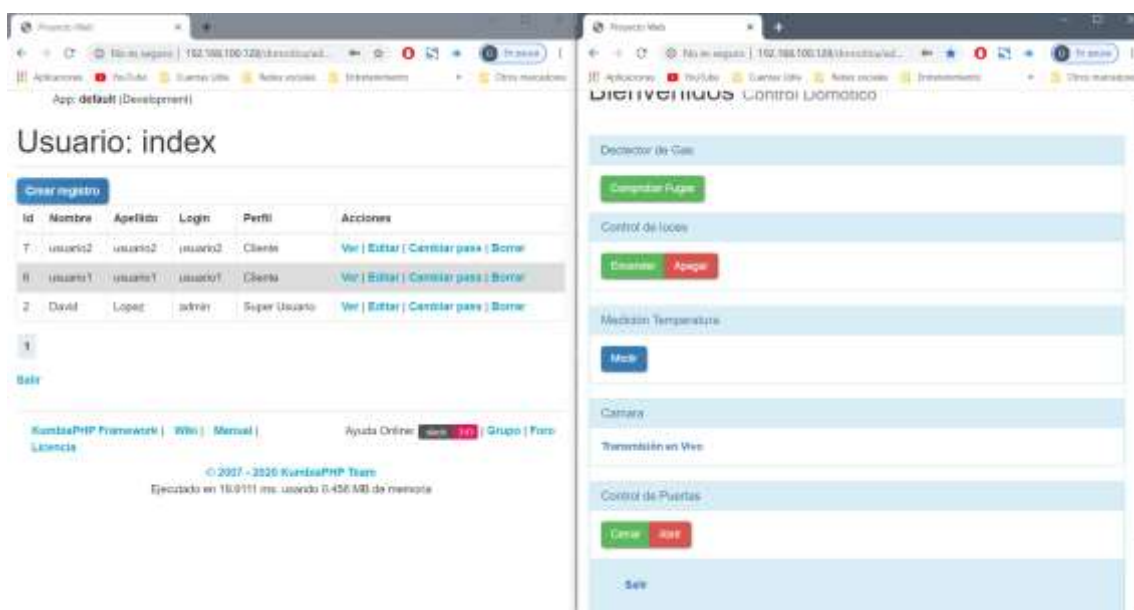


Figura 40. Ingreso respectivo como usuario y como administrador.

Caso de prueba 3.

Caso de prueba	
Código: 3	HU: Comprobación de fuga de gas
Nombre: Sensor de Gas	
Descripción: Permitirá que el sensor de gas detecte en tiempo real si existe alguna fuga	
Condiciones de ejecución: Esto se llevará a cabo cuando se da clic en el botón de “comprobar fugas”.	
Entrada/ Pasos de ejecución:	Respuesta del sistema:
Ingresar con contraseña	Ingreso correcto
Dar clic en el botón “comprobar fugas”	Se observa el estado actual del sensor (normal o existencia de gas).

Tabla 24: Caso de prueba 3.



Figura 41: comprobación de sensor de fugas de gas estado normal.

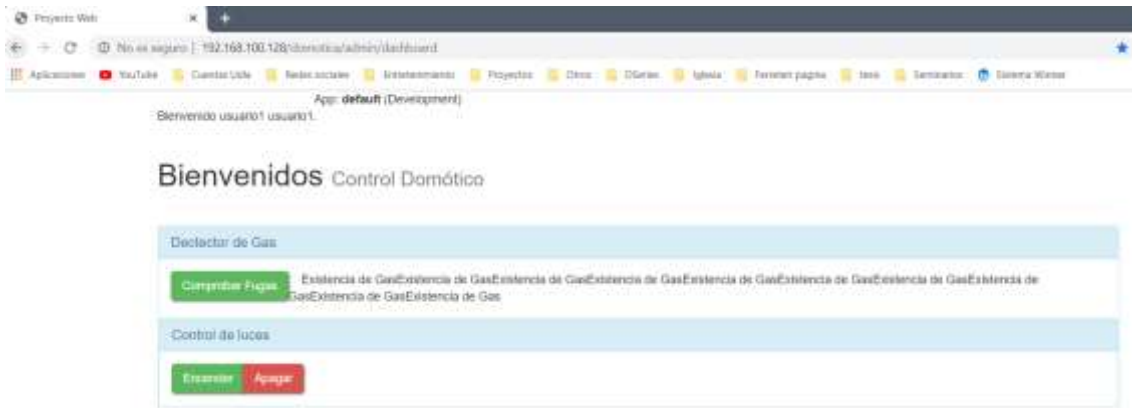


Figura 43: comprobación de encendido de luces.

Caso de prueba 5.

Caso de prueba	
Código: 5	HU: Comprobación de temperatura y humedad
Nombre: Sensor Temperatura y Humedad	
Descripción: La comprobación de la temperatura y humedad no se llevará a cabo hasta hacer clic en el botón de "Medir".	
Condiciones de ejecución: Esto se llevará a cabo cuando se da clic en el botón de "Medir".	
Entrada/ Pasos de ejecución:	Respuesta del sistema:
La comprobación de temperatura y humedad no se llevará a cabo hasta dar clic en el botón "Medir"	No se muestra la información relacionada con la temperatura y humedad.
Dar clic en el botón "Medir"	Se muestra la información capturada por el sensor relacionada con la temperatura y humedad.

Tabla 24: Caso de prueba 5.

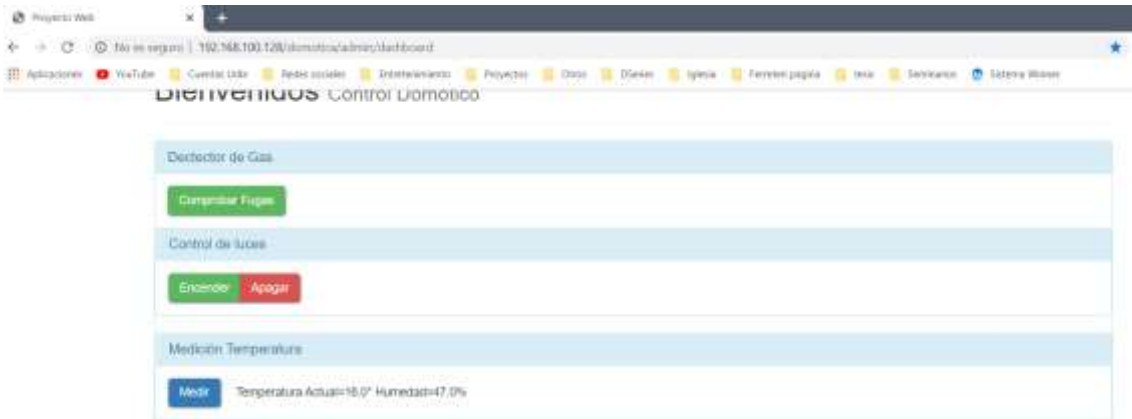


Figura 44. comprobación del sensor de temperatura y humedad.

Aplicando un poco de calor se observa que en la segunda prueba la temperatura varia y humedad variaron.

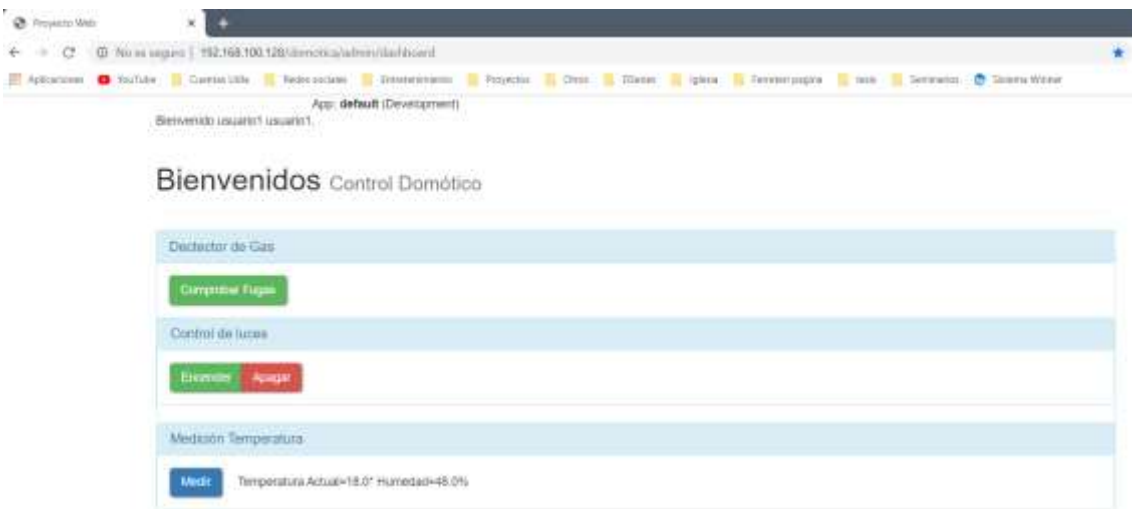


Figura 45: comprobación del sensor de temperatura y humedad.

Caso de prueba 6.

Caso de prueba	
Código: 6	HU: Control apertura de puerta
Nombre: Control de cerradura	
Descripción: Permitirá que la cerradura se abra.	
Condiciones de ejecución: Esto se llevará a cabo cuando se da clic en el botón de "Abrir Puerta".	
Entrada/ Pasos de ejecución:	Respuesta del sistema:
Ingresar a la aplicación	Se despliega la pantalla del login
Ingresar usuario	Se observa los caracteres ingresados
Ingresar contraseña	Se observa los caracteres ingresados de manera oculta
Dar clic en el botón ingresar	Se despliega la pantalla de los sensores
Al dar clic en el botón "Abrir Puerta"	La cerradura será accionada.

Tabla 25. Caso de prueba 6.

Caso de prueba 7.

Caso de prueba	
Código: 7	HU: Visualización de cámara de vigilancia
Nombre: Cámara de vigilancia	
Descripción: Permitirá que la cámara transmita en vivo	
Condiciones de ejecución: Esto se llevará a cabo cuando se da clic en el botón de "transmitir en vivo"	
Entrada/ Pasos de ejecución:	Respuesta del sistema:

Dar clic en el botón “transmitir en vivo”	Se abre una pestaña con la imagen de la cámara
---	--

Tabla 26: Caso de prueba 7

Por medio de la aplicación motion se puede observar la imagen utilizando una webcam.

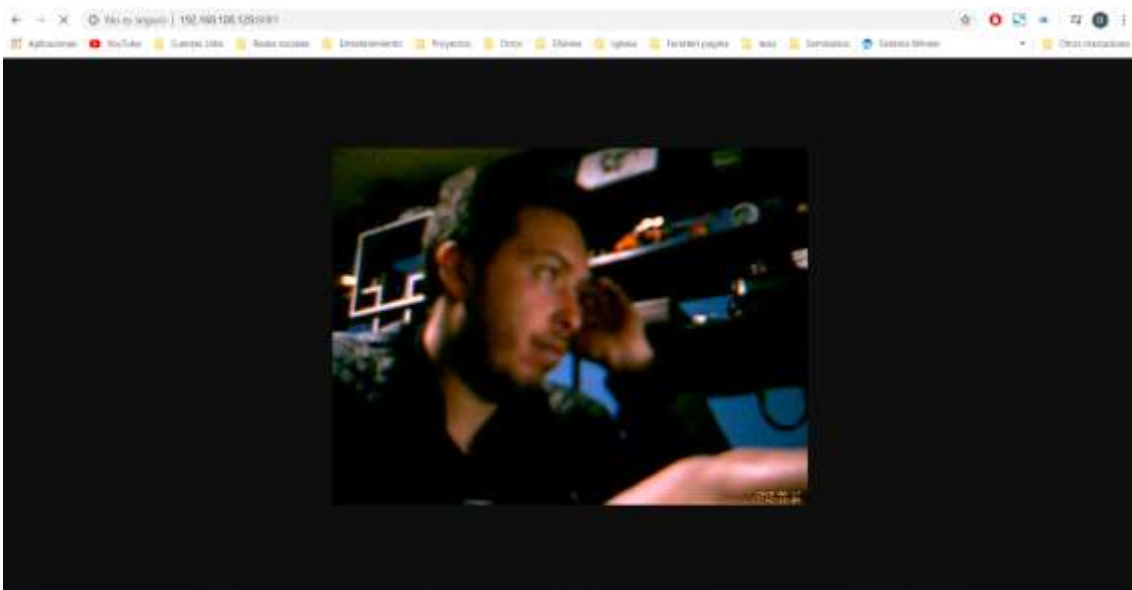


Figura 46: Funcionamiento de la cámara de vigilancia.

Caso de prueba 7.

Caso de prueba	
Código: 8	HU: Reconocimiento Facial
Nombre: Reconocimiento facial	
Descripción: Permitirá reconocer el rostro de la persona que se encuentre en la entrada de la casa.	
Condiciones de ejecución: Esto se llevará a cabo cuando se da clic en el botón reconcomiendo facial.	

Entrada/ Pasos de ejecución:	Respuesta del sistema:
Dar clic en el botón “Entrenar”	Se abre una pestaña con la imagen de la cámara
Dar clic en el botón “Reconocimiento facial”	Se abre una pantalla para detectar el rostro y muestra un cuadro con el nombre.

Tabla 27: Caso de prueba 8

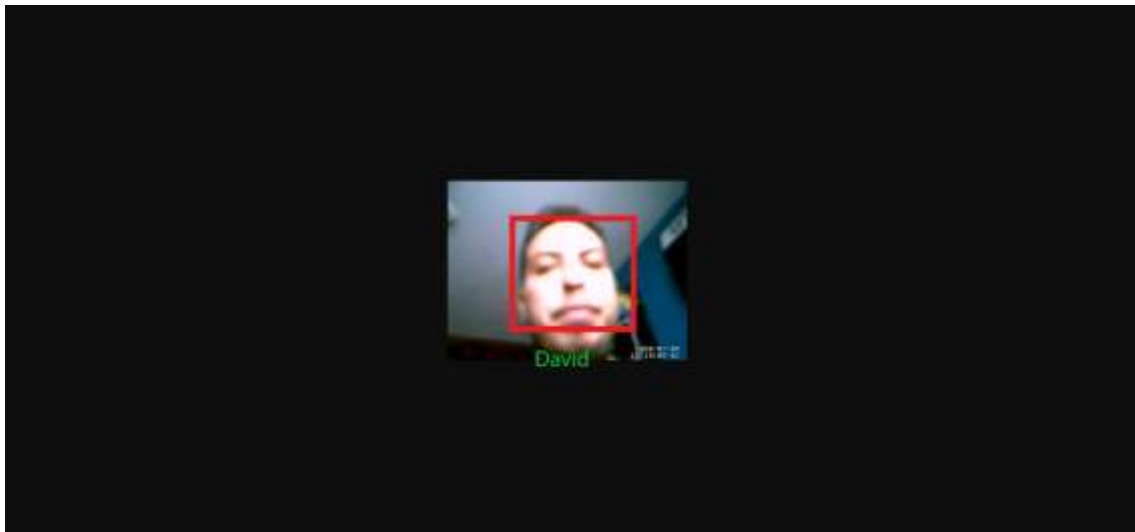


Figura 47. Funcionamiento de reconocimiento facial.

CAPITULO VI CONCLUSIONES Y RECOMENDACIONES

6.1 Conclusiones

- El análisis se llevó a cabo gracias a un esquema de flujo y arquitectura que se planteó logrando cumplir con todos los requerimientos propuestos.

- Se utiliza el raspberry pi 3 porque es un computador con capacidad de albergar un servidor, sensores y actuadores, como también manejar cantidades grandes de datos.
- El trabajar con diferentes lenguajes de programación ayudo a obtener mejores resultados, sin embargo, fue complejo trabajar con todos conjuntamente.
- La herramienta del diseño del hardware fue muy útil ya que muestra cómo serán las conexiones físicamente en el proyecto.
- El raspberry tiene la opción de ser escalable y permitir la conexión con más sensores.

6.2 Recomendaciones

- Se recomienda instalar un sistema operativo Ubuntu sin muchos programas, ya que ocupan mucho espacio y recursos innecesarios para el proyecto.
- Es necesario utilizar un sistema operativo actualizado, de la página oficial de raspberry, ya que muchas librerías de Python antiguas generan errores.
- Es importante instalar un framework que se adapte al sistema operativo Linux para que su manejo sea más fácil y consistente.
- En caso que se requiera aumentar funciones al sistema domótico, es necesario comprar el ultimo raspberry pi4, ya que cuenta hasta con 8GB de RAM y permite una mejor velocidad y procesamiento.

REFERENCIAS

Sáez Vacas, F. y Domínguez, H. (2006). Domótica: Un enfoque sociotécnico. Madrid, España: Editorial Fundación Rogelio Segovia para el Desarrollo de las Telecomunicaciones.

Unocero, (2018, 11 noviembre). Qué es SmartThings y por qué es la mejor opción para tener un hogar inteligente. Recuperado 13 de Abril de 2020, de: <https://www.unocero.com/gadgets/samsungsmarththings-telcel-hogar-inteligente/>

INNOTECH, (2017). Lutron Caseta Kit de iluminación inalámbrico. Recuperado 13 de Abril de 2020, de: <http://www.innotec.com.ec/productos-domotica.html>.

Rodríguez A. (2014). Diseño de un sistema domótico centralizado. Recuperado 3 de Julio de: <https://uvadoc.uva.es/bitstream/handle/10324/12922/TFGP206.pdf;jsessionid=60EB32BB1D0376249EBEA35389FBEF52?sequence=1>

Guinaldo M, Sanchez J, Dormido S (2017). Control en red basado en eventos: de lo centralizado a lo distribuido. Recuperado el 12 de Abril de 2020 de: https://www.researchgate.net/publication/312413050_Control_en_red_basado_en_eventos_de_lo_centralizado_a_lo_distribuido

SIMAD (2017). Ventajas de la administración web centralizada. Recuperado el 12 de Abril de 2020 de: <http://www.si-mad.com/ventajas-de-la-administracion-web-centralizada/>

Armijos K. (2019). Desarrollo de gestión de asistencia personal mediante metodología XP y lenguaje de programación java. Recuperado el 12 de Abril de 2020 de: <http://repositorio.utmachala.edu.ec/bitstream/48000/14551/1/ECFIC-2019-ISIS-DE00024.pdf>

IONOS (2020). Kanban. Recuperado el 12 de Abril de 2020 de: <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/que-es-kanban/>

Perez A (2019). Ventajas y desventajas de la metodología Scrum. Recuperado el 12 de Abril de 2020 de: <https://blog.wearedrew.co/ventajas-y-desventajas-de-la-metodologia-scrum>

Abellán E (2020). Metodología Scrum: qué es y cómo funciona. Recuperado el 12 de Abril de 2020 de: <https://www.wearemarketing.com/es/blog/metodologia-scrum-que-es-y-como-funciona.html>

Scrum. (2020). What is Scrum?. Recuperado el 12 de Abril de 2020 de: <https://www.scrum.org/resources/what-is-scrum>

Kyaw A, Phat H, Joseph (2017) J Low-Cost Computing Using Raspberry Pi 2 Model B. Recuperado el 12 de Abril de 2020 de: <http://www.jcomputers.us/vol13/jcp1303-05.pdf>

Rosebrock A (2018). Face recognition with OpenCV, Python, and deep learning. Recuperado el 13 de Abril de 2020 de: <https://www.pyimagesearch.com/2018/06/18/face-recognition-with-opencv-python-and-deep-learning/>

Arduino (2020), What is Arduino?. Recuperado el 12 de Abril de 2020 de: <https://www.arduino.cc/en/Guide/Introduction>

Raspberry pi foundation (2020). GPIO. Recuperado el 12 de Abril de 2020 de: <https://www.raspberrypi.org/documentation/usage/gpio/>

AOSONG (2018). Hoja de datos de dht11. Recuperado el 12 de Abril de 2020 de:

<https://www.geekfactory.mx/download/hoja-de-datos-o-datasheet-de-dht11-sensor-de-temperatura-y-humedad-relativa/>

Winsen (2014). Flammable Gas Sensor. Recuperado el 12 de Abril de 2020 de: <https://cdn.sparkfun.com/datasheets/Sensors/Biometric/MQ-6%20Ver1.3%20-%20Manual.pdf>

Kumbia (2020). ¿Qué es KumbiaPHP?. Recuperado el 18 de Mayo de 2020 de:
<https://kumbiaphp.com/blog/about/>

PHP (2020). ¿Qué es PHP?. Recuperado el 18 de Mayo de 2020 de:
<https://www.php.net/manual/es/intro-what-is.php>

PHP (2020). ¿Qué puede hacer PHP?. Recuperado el 18 de Mayo de 2020 de:
<https://www.php.net/manual/es/intro-what-cando.php>

Raspberry (2020). Python. Recuperado el 18 de Mayo de 2020 de:
<https://www.raspberrypi.org/documentation/usage/python/>

Luca (2020). ¿Qué es Python?. Recuperado el 18 de Mayo de 2020 de:
<https://luca-d3.com/es/data-speaks/diccionario-tecnologico/python-lenguaje>

Laurie W. (2010). Advances in Computers. Recuperado el 18 de Mayo de 2020 de: <https://www.sciencedirect.com/topics/computer-science/planning-poker/pdf>

