

**DESARROLLO DE UN SISTEMA BIOMÉTRICO PARA EL
CONTROL DE TIEMPOS Y ASISTENCIA AL LABORATORIO
ER103 DE INGENIERÍA ELECTRÓNICA DE LA UNIVERSIDAD
DE PAMPLONA**

(Autor)

ROBLES GONZALEZ JESÚS MANUEL

**INGENIERÍA ELECTRÓNICA
DEPARTAMENTO DE ELECTRÓNICA, ELÉCTRICA, SISTEMAS
Y TELECOMUNICACIONES
FACULTAD DE INGENIERÍAS Y ARQUITECTURA**



**UNIVERSIDAD DE PAMPLONA
PAMPLONA, mayo 25 de 2018**

**DESARROLLO DE UN SISTEMA BIOMÉTRICO PARA EL
CONTROL DE TIEMPOS Y ASISTENCIA AL LABORATORIO
ER103 DE INGENIERÍA ELECTRÓNICA DE LA UNIVERSIDAD
DE PAMPLONA**

(Autor)
ROBLES GONZALEZ JESÚS MANUEL

**Trabajo de grado presentado como requisito para optar al título de
INGENIERO ELECTRÓNICO**

Director: LUIS ALBERTO MUÑOZ BEDOYA
MSc(c). En Ingeniería Electrónica
[Email](#)

**PROGRAMA DE INGENIERIA MECANICA
DEPARTAMENTO DE INGENIERIA MECANICA,
MECATRONICA E INDUSTRIAL
FACULTAD DE INGENIERIAS Y ARQUITECTURAS
UNIVERSIDAD DE PAMPLONA
PAMPLONA, mayo 25 de 2018**

DEDICATORIA

Lleno de regocijo, de amor y esperanza, dedico este proyecto, a cada uno de mis seres queridos, quienes han sido mis pilares para seguir adelante. Es para mí una gran satisfacción poder dedicarles a ellos, que con mucho esfuerzo, esmero y trabajo me lo he ganado.

A mis padres Alejo Robles y Cecilia Gonzalez, porque ellos son la motivación de mi vida mi orgullo de ser lo que seré.

A mis hermanas Liz Robles, Claudia Robles, Magda Robles y mi ahijada Kimberly Noriega Robles, porque son la razón de sentirme tan orgulloso de culminar mi meta, gracias a ellos por confiar siempre en mí.

Y sin dejar atrás a toda mi familia por confiar en mí, a mis abuelitos, tíos y primos, gracias por ser parte de mi vida y por permitirme ser parte de su orgullo.

AGRADECIMIENTOS

El amor recibido, la dedicación y la paciencia con la que cada día se preocupaban mis padres por mi avance y desarrollo de esta tesis, es simplemente único y se refleja en la vida de un hijo.

Gracias a mis padres por ser los principales promotores de mis sueños, gracias a ellos por cada día confiar y creer en mí y en mis expectativas; gracias por siempre desear y anhelar siempre lo mejor para mi vida, gracias por cada consejo y por cada una de sus palabras que me guiaron durante mi vida.

Gracias a Dios por la vida de mis padres, también porque cada día bendice mi vida con la hermosa oportunidad de estar y disfrutar al lado de las personas que sé que más me aman, y a las que yo sé que más amo en mi vida, gracias a Dios por permitirme amar a mis padres, gracias a mis padres por permitirme conocer de Dios y de su infinito amor.

Gracias a la vida por este nuevo triunfo, gracias a mi Luis Alberto Muños (director de tesis) y mi compañero Eduardo Luis Pinedo Jimenéz que me ayudaron compartiéndome sus conocimientos, gracias a todas las personas que me apoyaron y creyeron en la realización de esta tesis.

TABLA DE CONTENIDO

INTRODUCCION	8
JUSTIFICACION	9
OBJETIVOS	11
1.1 Objetivo general.....	11
1.2 Objetivos específicos	11
CAPITULO 1. IMPLEMENTACION DE SISTEMAS BIOMETRICOS	12
1. Sistemas biométricos	12
1.1. Técnicas biométricas.	12
1.2. Huellas dactilares	15
1.2.1. Adquisición de huellas dactilares.	15
1.2.2. Codificación de huellas dactilares	16
Sensores de huellas digitales	19
1.3. Sistemas embebidos.....	20
1.3.1. Usos de los sistemas embebidos.	21
1.3.2. Características de los sistemas embebidos	22
1.3.3. Ventajas de un sistema embebido	23
1.4. Frameworks para desarrollo de web.....	23
CAPITULO 2. <i>PROCESO DE ADQUISICIÓN BIOMÉTRICA DEL LECTOR DE HUELLAS Y EL REGISTRO DE USUARIOS EN UN SISTEMA EMBEBIDO.</i>	27
CAPÍTULO 3. <i>DISEÑO DE BASE DE DATOS EN TIEMPO REAL Y EL FRONTEND DE VISUALIZACIÓN WEB PARA EL REGISTRO DE LA INFORMACIÓN DEL SISTEMA DE CONTROL DE ASISTENCIA.</i>	30
CAPÍTULO 4. VALIDACIÓN DEL FUNCIONAMIENTO DEL PROTOTIPO FINAL DEL SISTEMA DE CONTROL DE ASISTENCIA EN LOS LABORATORIOS DE INGENIERÍA ELECTRÓNICA.....	52
TABLA COMPARATIVA	59
DISCUSIÓN Y RESULTADOS	60
CONCLUSIONES	62
RECOMENDACIONES Y TRABAJOS FUTUROS	63
REFERENCIAS BIBLIOGRÁFICAS.....	64

TABLA DE ILUSTRACIONES

Ilustración 1: Minucias.....	14
Ilustración 2: correlación.....	14
Ilustración 3: Dos ejemplos de área patrón de una huella dactilar.....	17
Ilustración 4: Clases de huella.....	18
Ilustración 5: Disposición de núcleos y deltas en las diferentes clases.....	19
Ilustración 6. características de los sistemas embebidos.....	22
Ilustración 7: Raspberry pi 3	28
Ilustración 8: Programa Thonny.....	29
Ilustración 9 Diagrama de flujo del sistema de reconocimiento biométrico.....	31
Ilustración 10 proceso prototipó biometrico	32
Ilustración 11 Pagina Firebase	32
Ilustración 12: Programa Visual Studio code.....	33
Ilustración 13 Programación del Login en Visual Studio code.....	33
Ilustración 14: Frontend del login en Angular	34
Ilustración 15: Programación de la interfaz en Visual Studio code	34
Ilustración 16: Visualización de la interfaz en web frontend.....	35
Ilustración 17: Programación de las card del cuerpo docente en Visual Studio code.....	36
Ilustración 18: Visualización de las card del cuerpo docente	36
Ilustración 19: Programación del Registro de docentes en Visual Studio code.....	37
Ilustración 20: Registro de docentes en la web frontend de angular.....	37
Ilustración 21: Programación de los botones de la Selección de laboratorios en Visual Studio code	38
Ilustración 22: Visualización de la selección de laboratorio de frontend en angular.....	38
Ilustración 23: Programación de las card de los laboratorios en Visual Studio code	39
Ilustración 24: Visualización frontend del registro de materias.....	39
Ilustración 25: cierre de sesión.....	40
Ilustración 26 Diagrama de flujo del inicio del programa.	41
Ilustración 27 Pantalla de inicio del programa con las opciones de uso.....	41
Ilustración 28 Diagrama de flujo de la comprobación de la huella.....	42
Ilustración 29 Pantalla de Nuevo Registro.....	43
Ilustración 30 Diagrama de flujo del Nuevo Registro.	43
Ilustración 31 Diagrama de flujo de la sección <i>Buscar</i> de la aplicación.	44
Ilustración 32 Ventana de mensajes a lo largo de la aplicación.....	45
Ilustración 33 Ventana con teclado numérico para ingreso de numero de documento.....	46
Ilustración 34 Diagrama de flujo de la primera parte del programa de verificación.	47
Ilustración 35 Mensaje para colocar la huella.....	48
Ilustración 36 Diagrama de flujo de la Comprobación de la Huella para el caso de la verificación.....	48
Ilustración 37 Visualización de Información Docente Verificación.....	49
Ilustración 38 Diagrama de flujo del programa cuando se queda en espera del cierre de sesión.	50

Ilustración 39 Ejemplo de los mensaje que se irán mostrando en el transcurso del programa.	50
Ilustración 40 Diagrama de flujo del programa de verificación completo, las líneas rojas implican un sub-proceso.	51
Ilustración 41: Sistema de control biométrico	53
Ilustración 42: Raspberry PI 3 Model B	54
Ilustración 43: Base ergonómica con lector de huella empotrado	55
Ilustración 44 base de dato firebase	56
Ilustración 45 validación de registro de asistencia.....	58

INTRODUCCION

Ésta propuesta de trabajo de grado pretende desarrollar un sistema biométrico para el control de tiempos y asistencia al laboratorio ER103 de Ingeniería Electrónica de la Universidad de Pamplona, esto será posible usando:

- 1) Sistema Embebido
- 2) Sensor Biométrico de Huella
- 3) Base de datos en tiempo real
- 4) Visualización WEB

La creación de usuarios en el sistema se realizará sobre el desarrollo web, éste estará conectado a una base de datos y al sistema embebido con el lector de huellas. A cada usuario se le asignará un cronograma semestral con horario y recurso físico (laboratorio). En cada clase, el usuario debe registrar el inicio y fin con la huella. El sistema automáticamente registrará las horas laboradas, así como también la asistencia. Al finalizar el semestre o en cualquier momento, se generarán los formatos en digital de asistencias y tiempos de cada usuario aplicando la política de gestión integrada de calidad de cero papeles.

JUSTIFICACION

El uso recurrente de formatos en papel va en contra de normativas vigentes de buenas prácticas para la reducción del consumo de papel en identidades públicas. Según el Ministerio de Tecnologías de la Información y las Comunicaciones, basándose en la Directiva Presidencial N° 04 del 3 abril del 2012, la importancia de reducir el consumo de papel mediante la utilización de tecnologías digitales es de carácter obligatorio para las entidades públicas en Colombia.

La iniciativa de esta propuesta de trabajo busca abarcar los siguientes aspectos:

1. Contribuir a la disminución del uso del papel, implementando un método alternativo para el registro de información en formatos repetitivos que se usan para el control de asistencia y monitoreo del uso de los recursos físicos en los Laboratorios de Ingeniería Electrónica de la Universidad de Pamplona.
2. Tener un registro certero de la asistencia y el tiempo laborado del personal que hace uso de los laboratorios en mención.
3. Optar por el título de Ingeniero Electrónico al desarrollar un sistema que aporte múltiples beneficios a los servicios que brindan los laboratorios de Ingeniería Electrónica que podrá ser replicado en las diferentes dependencias de la Universidad.

Consecución de la propuesta:

Mediante el desarrollo de un novedoso sistema de control de tiempo y asistencias, sobre un sistema embebido y un lector biométrico de huella. Todo lo anterior interactuando con una base de datos en tiempo real y un sitio web frontend para el monitoreo, registro de usuarios y generación de reportes.

Beneficios Institucionales:

- Uso óptimo de los recursos
- Buenas prácticas en gestión documental

- Mayor control y seguridad en el manejo de la información
- Eliminar duplicidad en documentos
- Disminuir los tiempos de localización de los archivos
- Reducir la necesidad de espacios de almacenamiento

Beneficios ambientales:

- Disminución de consumo de recursos naturales (árboles, agua y energía)
- Disminuir la contaminación producida por los productos blanqueadores de papel
- Reducir los residuos contaminantes como tóner y cartuchos de tinta.
- Contribuir al desarrollo sostenible, el consumo responsable de los recursos que no comprometa el desarrollo social y ambiental de las generaciones futuras.

OBJETIVOS

1.1 OBJETIVO GENERAL

Desarrollar un sistema biométrico para el control de tiempos y asistencia al laboratorio ER103 de ingeniería electrónica de la universidad de pamplona.

1.2 OBJETIVOS ESPECÍFICOS

1. Establecer el proceso de adquisición biométrica del lector de huellas.
2. Realizar el registro de usuarios en un sistema embebido.
3. Diseñar la base de datos en tiempo real y el frontend de visualización web para el registro de la información del sistema de control de asistencia.
4. Validar el funcionamiento del prototipo final del sistema de control de asistencia en los laboratorios de Ingeniería Electrónica.

CAPITULO 1. IMPLEMENTACION DE SISTEMAS BIOMETRICOS

1. Sistemas biométricos

Podemos decir que sistemas de reconocimiento biométrico están basados en una estructura singular que contiene: la huella dactilar, la huella palmar (palma de Ja mano), la geometría de la mano/dedos, la cara, el iris o la retina.

Otras características fisiológicas, que se utilizan con menos secuencias pueden ser: la forma de la oreja, la termo grama del cuerpo (o partes de él), la estructura de las venas, la estructura de poros (en alguna zona del cuerpo como, por ejemplo, la cara o la yema del dedo) o el olor corporal.

Por otro lado, encontramos que existen comportamientos a tener en cuenta como la voz, la escritura, la firma escrita, el modo de pulsar un teclado o el modo de andar.

1.1. Técnicas biométricas.

El problema común que se plantea en todas las técnicas biométricas es el de la definición del conjunto de características que identifican al individuo y que definen su patrón biométrico, la extracción de dichas características y el reconocimiento de los patrones cuando la identidad de un individuo es solicitada.¹

En los apartados que siguen a continuación se hace una revisión de las técnicas de reconocimiento biométrico basadas en los rasgos actualmente más utilizados.

¹ Instituto Nacional de Ciberseguridad. Tecnologías biométricas aplicadas a la ciberseguridad. Una guía de aproximación para el empresario.2016. Recuperado de: https://www.incibe.es/sites/default/files/contenidos/guias/doc/guia_tecnologias_biometricas_aplicadas_ciberseguridad_metad.pdf

Cara.

Las técnicas de tratamiento automático de cara pueden clasificarse según los siguientes campos de aplicación:

- Reconocimiento de cara en sistemas de seguridad y de aplicación forense.
- Indexación y extracción de personas a partir de imágenes estáticas e imágenes en movimiento.
- Codificación de cara en comunicaciones visuales.
- Reconocimiento de expresiones faciales, lectura de labios, estimación del género y de la edad.
- Síntesis de caras y animación.²

La cara es el rasgo biométrico más utilizado por la mayoría de las personas para reconocer a un individuo. Aunque es un rasgo cuya característica de unicidad es menor que la de la huella dactilar o el iris, su gran aceptabilidad y universalidad han contribuido significativamente a la investigación y desarrollo, desde comienzos de los años 60, de las técnicas de reconocimiento facial automático.³

Huella dactilar.

Es uno de los rasgos más utilizados, puesto que cada individuo lo tiene, desde su nacimiento, y permanece constante en su vida. La huella dactilar permite, además, discriminar perfectamente a los diferentes individuos y su grado de aceptabilidad es relativamente alto. Sin embargo requiere cuidado al momento

² Restrepo Jhony. Revisión de las técnicas básicas para el reconocimiento de rostros. Universidad Tecnológica de Pereira. Risaralda. 2012. Recuperado de: <http://repositorio.utp.edu.co/dspace/bitstream/handle/11059/6600/0053682R436.pdf?sequence=1>

³ Zorita Danilo Simón. Reconocimiento automático mediante patrones biométricos de huella dactilar. Universidad Politécnica de Madrid. España. 2003

de su captura con el objeto de perfeccionarse de la etapa de reconocimiento de patrones, y ser de utilidad real.⁴

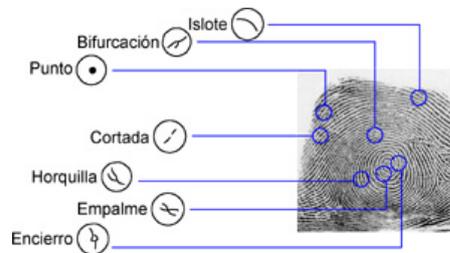


Ilustración 1: Minucias



Ilustración 2: correlación

Iris.

El patrón biométrico obtenido del iris es un patrón que se mantiene invariable a lo largo de toda la vida. Al igual que la huella dactilar o la retina es un patrón único para cada individuo, en la actualidad existen patrones muy adelantados y de excelente precisión que desarrollan nuevos algoritmos y la implementación de dispositivos de adquisición, hacen del reconocimiento basado en el iris, una técnica importante y con mucha calidad. Muestra el inconveniente de que la adquisición de la imagen del iris se lleva a cabo proyectando un haz de luz sobre los ojos, situación que generalmente no es aceptada por los usuarios.⁵

⁴Restrepo Jhony. Revisión de las técnicas básicas para el reconocimiento de rostros. Universidad Tecnológica de Pereira. Risaralda. 2012. Recuperado de: <http://repositorio.utp.edu.co/dspace/bitstream/handle/11059/6600/0053682R436.pdf?sequence=1>

⁵ Ibid

Voz.

La señal de voz es una señal muy compleja, cuya forma de onda se produce por la conjunción de diversos factores, como son: los fisiológicos del tracto vocal, la articulación del lenguaje, la semántica de la oración y los aspectos lingüísticos del idioma. Las diferencias apreciables que aparecen en la señal de voz de diferentes locutores, son debidas a las diferencias anatómicas propias del tracto vocal y a los hábitos adquiridos en el habla. En el reconocimiento de locutores, que es el ámbito de aplicación aquí tratado, todas esas diferencias pueden usarse para diferenciar un individuo de otro.⁶

1.2. Huellas dactilares

La huella dactilar en su estructura como método de identificación Está formada por crestas papilares y surcos interpapilares están localizados en la piel los cuales forman los dibujos en la falange distal de los dedos de las manos, lo que hace que se formen puntos característicos, un núcleo y uno o varios deltas, con el fin de llevar a cabo la verificación de identidad.⁷.

1.2.1. Adquisición de huellas dactilares

La forma en la que se realiza el proceso de adquisición de huellas dactilares es muy diferente, dependiendo del tipo de aplicación biométrica en la que se van a procesar las imágenes obtenidas. El método tradicional de adquisición de huellas dactilares tintadas ha tenido siempre, y sigue teniendo lugar, en el ámbito judicial y forense.⁸

⁶ Ibid

⁷ Maya Adriana. (2013) Sistema biométrico de reconocimiento de huella dactilar en control de acceso de entrada y salida. Universidad Militar Nueva Granada. Bogotá D.C

⁸ Zorita Danilo Simón. Reconocimiento automático mediante patrones biométricos de huella dactilar. Universidad Politécnica de Madrid. España. 2003

El esquema general de estos dispositivos comprende las siguientes partes: *sensor* de lectura, encargado de capturar la imagen de la huella; *conversor analógico/digital*, encargado de convertir la imagen analógica entregada por el sensor en una señal digital; *interfaz de comunicaciones* con otros dispositivos extremo (por ejemplo, un ordenador personal). Dependiendo del principio físico de funcionamiento del sensor, se definen los diferentes tipos: óptico, de estado sólido y ultrasónico.⁹

1.2.2. Codificación de huellas dactilares

La clasificación de huellas dactilares se realiza normalmente en aquellos sistemas de identificación que manejan grandes bases de datos, con el fin de limitar en lo posible el campo de búsqueda de las huellas que se desean. Dicha codificación se fundamenta en la específica distribución de las crestas y valles de la huella dactilar en la zona central de la imagen. Las configuraciones características que ahí aparecen, presentan cierta variabilidad dentro de la misma clase; sin embargo, tales variaciones son lo suficientemente pequeñas como para permitir establecer reglas sistemáticas de clasificación.¹⁰

1.2.2.1. Puntos únicos de la estructura de crestas de una huella.

En la clasificación de patrones de huella dactilar, solamente resulta de interés una zona de la imagen, que es la que se toma como patrón. El *área patrón* de una huella está formada por todas aquellas crestas y valles circunscritas entre dos crestas llamadas crestas de referencia. Estas crestas se definen como las dos crestas divergentes más internas de la estructura de la imagen que circunscriben la zona central de la huella (Figura 3).¹¹

⁹ Aguilar G, Sánchez G, Toscano K, Nakano M, Pérez H. (2008) Reconocimiento de Huellas Dactilares Usando Características Locales. Revista Facultad de Ingeniería Universidad de Antioquia N.º 46 pp. 101-109. Diciembre, 2008

¹⁰ Zorita Simon, (2003) Reconocimiento automático mediante patrones biométricos de huella dactilar. Escuela Técnica

¹¹ Ibid

- *Delta*. El punto delta, también conocido como punto singular extremo, se define como el punto más próximo al centro geométrico en el que tiene lugar la divergencia de las dos crestas de referencia.
- *Núcleo*. El núcleo o punto singular interno, se define como el punto situado sobre las crestas curvas más internas de la estructura..¹²
- *Cómputo de crestas*: Es, un parámetro importante para establecer la clasificación automática de huellas dactilares es el cómputo de crestas entre puntos singulares, resulta difícil establecer un criterio para realizar el cómputo preciso de crestas.



Ilustración 3: Dos ejemplos de área patrón de una huella dactilar

Las diferentes clases o tipos de huellas entre las que debe diferenciar un clasificador, se establecen a partir de la información estructural de las crestas que determinan los puntos singulares. Las categorías más encontradas son: *arco sencillo*, *arco tensado*, *lazo derecho*, *lazo izquierdo*, *rizo* y *doble rizo*.

Podemos definir las clases de huellas así:

- *Lazo*. Es aquella en la que las crestas entran por un lado de la imagen, fluyen hacia arriba rodeando a un núcleo con curvatura muy pronunciada, cortan la línea imaginaria que une el núcleo con una delta, y tienden a volver hacia el

¹² Ibid

lado de la imagen por el que entraron. Aproximadamente, entre un 60% y un 65% de las huellas pertenecen a esta clase¹³.

- *Arco*. El arco es una clase de huella muy particular, ya que a ella pertenecen menos del 5% de las huellas. Pueden darse dos tipos diferentes de arco: el *arco sencillo* y el *arco tensado*. En un arco sencillo las crestas fluyen de un lado al otro de la imagen, alzándose y curvándose ligeramente en el centro de la misma. La diferencia entre una clase y la otra se establece midiendo los ángulos que forman las crestas al cortar la línea imaginaria que une núcleo y delta. En un arco tensado dichos ángulos valen prácticamente cero.¹⁴
- *Rizo*. Una huella pertenece a la clase rizo cuando la estructura de crestas presenta al menos dos puntos delta, frente a los cuales fluyen las crestas curvándose alrededor de un núcleo. Aproximadamente, entre un 30% y un 35% de las huellas pertenecen a esta clase. (Figura 4)¹⁵



Ilustración 4: Clases de huella

¹³ Aguilar G, Sánchez G, Toscano K, Nakano M, Pérez H. Reconocimiento de Huellas Dactilares Usando Características Locales. Revista Facultad de Ingeniería Universidad de Antioquia N.º 46 pp. 101-109. Diciembre, 2008

¹⁴ Ibid

¹⁵ Ibid

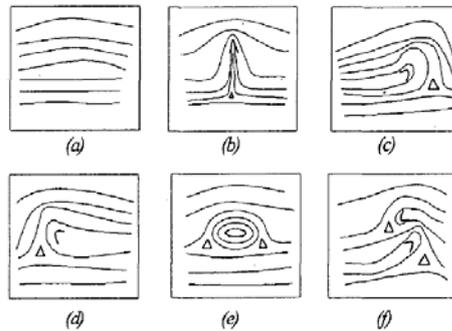


Ilustración 5: Disposición de núcleos y deltas en las diferentes clases

Sensores de huellas digitales

Tradicionalmente las huellas digitales se adquieren al transferir una impresión de tinta de una huella digital al papel. Este proceso es denominado adquisición off-line.¹⁶

Los sistemas de autenticación existentes están basados en dispositivos que realizan la adquisición de la imagen de la huella dactilar en tiempo real, este proceso es denominado live-scan. Los dispositivos para realizar live-scan suelen estar basados en uno de los siguientes esquemas de sensores.

1. Sensores Ópticos: Es la tecnología más vieja y más usada. En la mayoría de los dispositivos, en la mayoría de los casos poseen un dispositivo charged couple device (CCD), que convierte la imagen de la huella digital, con oscuras crestas y valles luminosos, en una señal digital.
2. Sensores Capacitivos: Un sensor de silicona actual como un extremo del capacitor y el dedo a extraer la huella digital como el otro extremo.
3. Sensores de Ultrasonido: Ultrasonido es quizás la tecnología con mayor nivel de certeza dentro de las tecnologías de censado de huellas dactilares. utiliza ondas de ultrasonido para medir la distancia, basándose en la impedancia de los dedos.

¹⁶ Pusiol Guido. (2007) Sistema de verificación de huellas digitales.

4. Sensores Térmicos: Estos sensores están erigidos por materiales pyro-electricos cuyas propiedades cambian con la temperatura. Dado que la piel logra equilibrio térmico en poco tiempo una vez apoyado el dedo en el scanner.¹⁷

El scanner solo puede adquirir pequeñas zonas de la huella por cada proceso de adquisición, obtener una imagen de tamaño completo de la huella digital es complejo debido a que tras la adquisición de todas las zonas de la huella hay que proceder con la restauración de la imagen uniendo las zonas logradas.¹⁸

1.3. Sistemas embebidos

Los sistemas embebidos ajustan interfaces de entrada, un procesador interno y un periférico de salida, poseen gran cantidad de recursos físicos y están diseñados para realizar diferentes funciones como lo son las repetitivas, frecuentemente en un sistema de computación en tiempo real, están diseñados para cubrir un amplio rango de necesidades. Por lo general los sistemas embebidos se pueden programar directamente en el lenguaje ensamblador del microcontrolador o microprocesador incorporado sobre el mismo, o también, utilizando los compiladores específicos.¹⁹ Los productos inteligentes se han convertido en algo común en casi todos los mercados de consumo industriales.²⁰

¹⁷ Crespo Claudia Marcela y Fernández Ernesto. Diseño e implementación de un scanner 3D para prototipado y modelado de objetos. Universidad autónoma de Barranquilla. 2014.

¹⁸ Ibid

¹⁹ Román Juliana. & Gonzales Kelly (2013) Sistemas embebidos y Hardware libre. Escuela de Ingeniería de Sistemas e Informática Universidad Industrial de Santander Bucaramanga, Santander. Disponible en: <http://wiki.sc3.uis.edu.co/images/e/e1/GR7.pdf>

²⁰ Duque José Alejandro. Metodología integral para el emprendimiento basado en sistemas embebidos digitales en Colombia. Universidad Nacional de Colombia. Bogotá D.C., Colombia. 2017

El software embebido puede ejecutar funciones específicas, como por ejemplo el control de las teclas de un horno de microondas, o suministrar una función significativas.²¹

La primordial regla del software embebido es interactuar con el mundo físico, mientras que en el progreso de aplicaciones empresariales, se busca olvidarse del mundo físico y enfocarse en abstracciones como entidades de información y proceso de negocios, en el desarrollo del software embebido sucede todo lo contrario.²²

1.3.1. Usos de los sistemas embebidos.

Los lugares donde se pueden encontrar los sistemas embebidos son numerosos y de varias naturalezas, como son en un una fábrica, para controlar un proceso de montaje o producción, en puntos de servicio o venta (POS, point of sale), en Puntos de información al ciudadano, en decodificadores y set-top boxes para la recepción de televisión..²³

También se pueden encontrar aplicaciones de sistemas embebidos con microcontroladores: los aparatos electrónicos, los sistemas de comunicación, los sistemas automotores, el equipo militar, las aplicaciones empresariales, los dispositivos médicos .²⁴

Por otro lado, Sistemas mecatrónicos: La mecatrónica es una sinergia de sistemas mecánicos y eléctricos controlados por un sistema embebido, Redes inalámbricas de sensores, ya que coexisten varias tecnologías propuestas para resolver el problema de comunicación inalámbrica en este tipo de redes, entre ellas se

²¹ Ibid

²² Ibd

²³ Ibid

²⁴ Hernández Vega, José Isidro (2010) El Software Embebido y los Retos que Implica su Desarrollo Conciencia Tecnológica, núm. 40, julio-diciembre, 2010, pp. 42-45 Instituto Tecnológico de Aguascalientes Aguascalientes, México

encuentra Zigbee.²⁵ También es posible conocer aplicaciones en FPGA (*Field Programmable Gate Array*), esta contiene bloques de lógica cuya interconexión y funcionalidad se puede programar.²⁶

1.3.2. Características de los sistemas embebidos

Los tres atributos que típicamente tienen consideraciones especiales en el desarrollo de software embebido son: confiabilidad, limitaciones en recursos de hardware y respuesta en tiempo real.²⁷

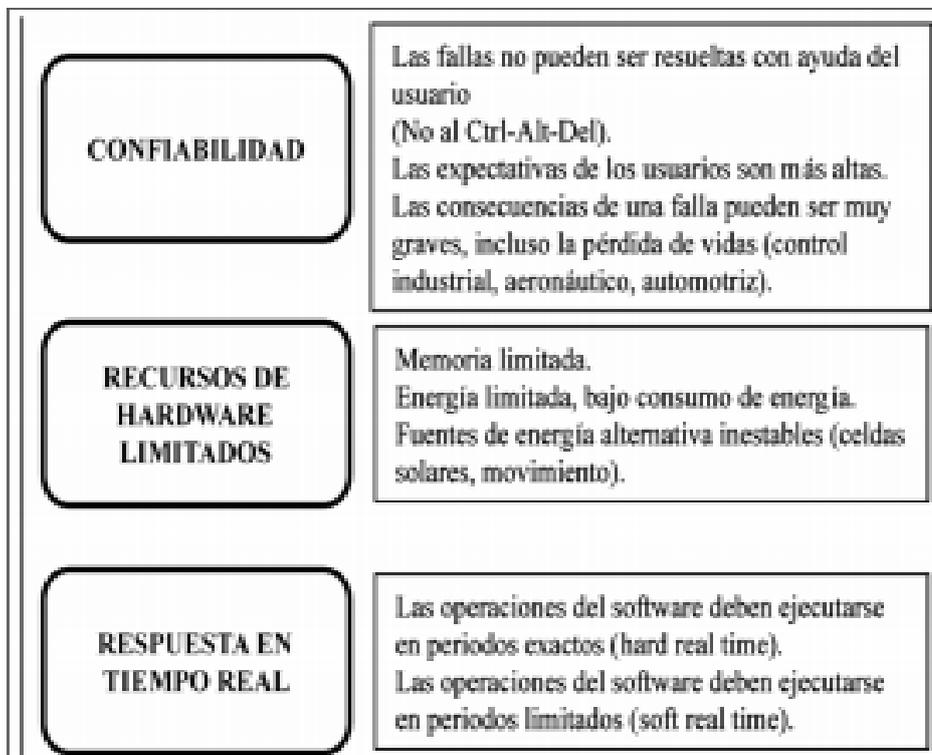


Ilustración 6. características de los sistemas embebidos

²⁵ Ibid

²⁶ Ibid

²⁷ Hernández Vega, José Isidro (2010) El Software Embebido y los Retos que Implica su Desarrollo *Conciencia Tecnológica*, núm. 40, julio-diciembre, 2010, pp. 42-45 Instituto Tecnológico de Aguascalientes Aguascalientes, México

1.3.3. Ventajas de un sistema embebido

Los sistemas embebidos tienen ventajas importantes que permiten mejor utilización de estas herramientas y más efectividad en los procedimientos, por ejemplo se tiene la posibilidad de utilización de sistemas operativos potentes que ya realizan numerosas tareas: comunicaciones por redes de datos, soporte gráfico, mejor utilización concurrencia con lanzamiento de *threads*, etc; por otro lado, al utilizar los sistemas embebidos, se pueden encontrar fácilmente herramientas de desarrollo de software potentes, así como numerosos programadores que las dominan, dada la extensión mundial de las aplicaciones para computadoras compatibles y además, la reducción en el precio de los componentes hardware y software debido a la gran cantidad de computadoras en todo el mundo.²⁸

1.4. Frameworks para desarrollo de web

Hablar de los mejores Framework JavaScript el día de hoy, es hablar del crecimiento que ha tenido JavaScript en los últimos tiempos, la popularidad que ha tenido y sigue teniendo va en aumento. Sólo en el año 2016 ha sido testigo de grandes cambios como por ejemplo AngularJS con actualización completa y la aparición de un nuevo Angular denominado asociado Angular.²⁹

Quién no sabe hablar de jQuery y su dominio en casi el 95% de los sitios web que hoy se implementan. Apariciones de otras tecnologías basadas en JavaScript Cómo reaccionar o Node.JS, y ahora Vue.JS que aparece como una nueva alternativa que está ganando mucho terreno en el desarrollo FrontEnd, JavaScript también tiene parte en el desarrollo BackEnd gracias a NodeJS.

²⁸http://www.semanticwebbuilder.org.mx/en_mx/swb/Sistemas_Embebidos_Innovando_hacia_los_Sistemas_Inteligentes_

²⁹ <https://www.beeva.com/beeva-view/tecnologia/la-evolucion-del-framework-angularjs/>

Es importante centrarnos en algunos de los marcos de JavaScript, que han tenido gran crecimiento en los últimos años. Entre estos tenemos: El ReactJS más que un Framework es una biblioteca. Sabemos que se encuentra detrás de las interfaces de usuario de Facebook e Instagram, mostrando su eficiencia dentro de las aplicaciones dinámicas de alto tráfico. Debido al uso de un DOM virtual viene proporcionando un aumento considerable del rendimiento de las aplicaciones en comparación con las versiones de AngularJS, los componentes que se crean con ReactJS pueden ser reutilizados entre diferentes aplicaciones.³⁰

Por otro lado, el EmberJS fue considerado como uno de los mejores JavaScript Frameworks para la web, dejando atrás React y AngularJS. Hoy en día cuenta con una enorme comunidad en línea, actualizaciones regulares y amplia aplicación de mejores prácticas en JavaScript para garantizar la mejor experiencia final. Además, cuenta con enlace de datos bidireccional, como Angular, manteniendo la vista y el modelo sincronizados todo el tiempo. Al aplicar el módulo Fastboot.js, se garantiza la representación puntual del DOM en el lado del servidor, mejorando el rendimiento de las interfaces de usuario complejas.³¹

Otro muy importante es el MeteorJS es considerado también uno de los mejores frameworks JavaScript más populares, viene bien equipado con toneladas de características para el desarrollo de back-end, renderizado de frontend, gestión de bases de datos y lógica de negocio. Desde su lanzamiento en 2012, su ecosistema ha crecido drásticamente a un ritmo muy acelerado. Esta plataforma permite un rápido desarrollo de aplicaciones web y móviles de extremo a extremo en JavaScript puro. Debido a la estructura modular todos los paquetes y bibliotecas se pueden reutilizar. En términos de rendimiento, todos los cambios en la base de datos son inmediatamente transmitidos a la interfaz de usuario y, por el contrario, sin pérdidas

³⁰ Krall Cesar. (2006-2029) Programación Webapp Javascript. Librerías, Frameworks. Jquery, Angularjs. Ventajas. Diferencias. (Cu01194e) APRENDERAPROGRAMAR.COM. Descargado de: http://www.aprenderaprogramar.com/index.php?option=com_attachments&task=download&id=803

³¹ Ibid

de tiempo evidentes causadas por diferentes idiomas o el tiempo de respuesta del servidor.³²

También se puede citar, el VueJS en su versión 2.0 también fue introducido en el año 2016 y tomó lo mejor de Ember, React y Angular, poniendo todo eso en un paquete práctico, esto ha permitido que hoy en día sea considerado como uno de los mejores Frameworks JavaScript en crecimiento para este 2017. Se ha demostrado ser más rápido y más delgado, en comparación con React y Angular. Internamente, Vue.js ofrece enlace bidireccional de datos (visto en AngularJS), renderizado al lado del servidor (como en Angular2 y ReactJS), Vue-cli (herramienta para el inicio rápido) y soporte opcional JSX. Su fundador afirma que VueJS 2 es uno de los Frameworks más rápidos en general.³³

Uno de los más conocidos es el Angular.js, fundamenta su trabajo como un marco MVW (Model-View-Whatever, Modelo-Vista-Loquesea) y entre los principales beneficios se consideran: la producción rápida de código, la prueba fácil de cualquier parte de la aplicación y datos bidireccionales (los cambios en el backend se reflejan inmediatamente en la interfaz de usuario). Angular2, 3 y 4 viene con una larga lista de características que permiten construir todo, que van desde la web a escritorio y móviles.³⁴

El Framework está construido con TypeScript de Microsoft con el objetivo de hacer que JavaScript sea más ágil y atractivo para las grandes empresas. Ng2 cuenta con una arquitectura basada en componentes, DI mejorada (inyección de dependencia), servicio de registro eficiente, comunicaciones entre componentes y mucho más.³⁵

³² Krall Cesar. (2006-2029) Programación Webapp Javascript. Librerías, Frameworks. Jquery, Angularjs. Ventajas. Diferencias. (Cu01194e) APRENDERAPROGRAMAR.COM. Descargado de: http://www.aprenderaprogramar.com/index.php?option=com_attachments&task=download&id=803

³³ Krall Cesar. (2006-2029) Programación Webapp Javascript. Librerías, Frameworks. Jquery, Angularjs. Ventajas. Diferencias. (Cu01194e) APRENDERAPROGRAMAR.COM. Descargado de: http://www.aprenderaprogramar.com/index.php?option=com_attachments&task=download&id=803

³⁴ Aviles López Javier. Desarrollo de una aplicación web para compartir medio de transporte con AngularJS. Universidad Politécnica de Cartagena. 2014.

³⁵ Ibid

Tanto AngularJS como Angular son una mejor opción para las aplicaciones basadas en la empresa o para entornos de programación estrictos con altos estándares de legibilidad del código.

Además, ahora con la aparición de React, AngularJS ha tenido que incorporar la visión de páginas basadas en componentes, desarrollando los suyos propios. Angular crea una buena primera impresión debido a la implementación de la lógica a través de etiquetas que se actualizan por sí solas, sin embargo, se han encontrado varios defectos. Una de las reglas fundamentales en programación es que siempre se prefiere la forma explícita a la implícita como ocurre con la invocación de los manejadores de eventos de AngularJS.³⁶

En cuanto a la elección entre Angular JS y jQuery, siempre depende de la naturaleza de la aplicación. Si estamos hablando de una aplicación compleja, entonces, personalmente, optaría por AngularJS, de lo contrario prueba con jQuery.³⁷

Una de las victorias de AngularJs fue cuando Google compró la compañía DoubleClick, reescribiendo gran parte del código con este *framework*. Debido al éxito generado, tuvo gran crecimiento, tanto a nivel interno de la compañía como externo.³⁸

³⁶ Revilla Danel (2016) Evaluación del uso de librerías modelo-vista-controlador para el desarrollo de servicios web. caso de estudio: REACT. Universidad Politecnica de Madrid. Descargado de: http://oa.upm.es/40491/1/PFC_DANIEL_REVILLA_TWOSE_2016.pdf

³⁷http://programacion.net/articulo/angularjs_vs_jquery-_cuales_son_sus_mayores_diferencias_1610

³⁸ibid

CAPITULO 2. PROCESO DE ADQUISICIÓN BIOMÉTRICA DEL LECTOR DE HUELLAS Y EL REGISTRO DE USUARIOS EN UN SISTEMA EMBEBIDO.

Se implementó un sistema de acceso por biométrica, donde se sensa la característica física única que cada persona posee, que no es otra que la huella dactilar esta peculiaridad en funciona como reconocimiento y autenticación para tener acceso (entrada y salida) a un servicio o sitio en específico, en este caso al registro de asistencia a laboratorios en la bases de datos.

El procesamiento de reconocimiento de huellas tiene diferentes etapas: Adquisición: En esta etapa se toma una muestra de la huella dactilar de un sujeto. Pre-procesado: La imagen obtenida en la etapa de adquisición se trata en función de las necesidades de los algoritmos de extracción y comparación que se vayan a utilizar. Extracción de características: Se extrae la información relevante de las huellas para su posterior comparación.

Comparación: Se comparan las características extraídas para determinar si las dos muestras pertenecen al mismo individuo. Existen algoritmos que no necesitan extracción de características y por tanto realizan directamente la comparación. Para esto se utilizo ademas del lector de huellas

Autenticación de huella

La autenticación de huella está compuesta por el módulo DIGITALPERSONA U.are.U 4500, que incluye un sensor de huella dactilar óptico, un chip DSP, y una Uart de comunicación serial.

Este también cuenta con dos buffers en los cuales se pueden leer y escribir, por medio de instrucciones, la información de estos buffers no se almacenan después

de que el modulo se apaga. Se escogió este módulo por su facilidad de conexión con cualquier micro controlador o sistema con TTL en serie, y su disponibilidad de adquirir en el mercado local.

Se utilizó la librería “fingerprint” para Python, la cual permite la conexión, transmisión y recepción de los datos. El módulo DIGITALPERSONA U.are.U 4500, se comunica por el protocolo de comunicación RS232, maneja por defecto una velocidad de 57600 bps, con un formato de trama de 10 bits, un bit de inicio, datos de 8 bits y un bit de parada, sin paridad. La Raspberry pi 3 modelo B solo le da instrucciones al módulo, y el modulo respuestas a la Raspberry, por medio de paquetes hexadecimales.

La Raspberry pi 3

Consiste de un computador de placa reducida, capaz de correr distintos tipos de sistemas operativos incluyendo sistemas operativos embebidos; ofrece herramientas de gran utilidad para este tipo de proyectos. El software es Open Source, siendo su sistema operativo oficial una versión adaptada de Debian, denominada Raspbian.

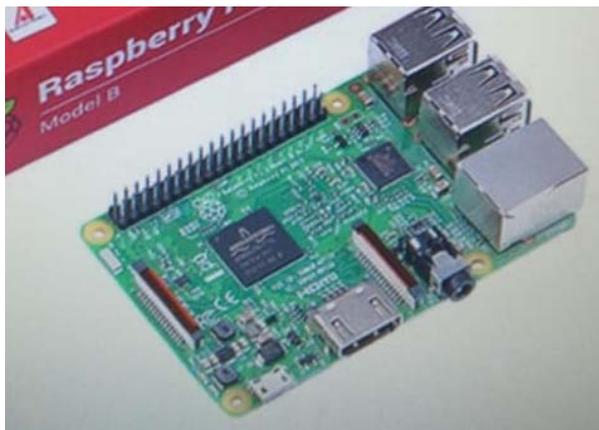
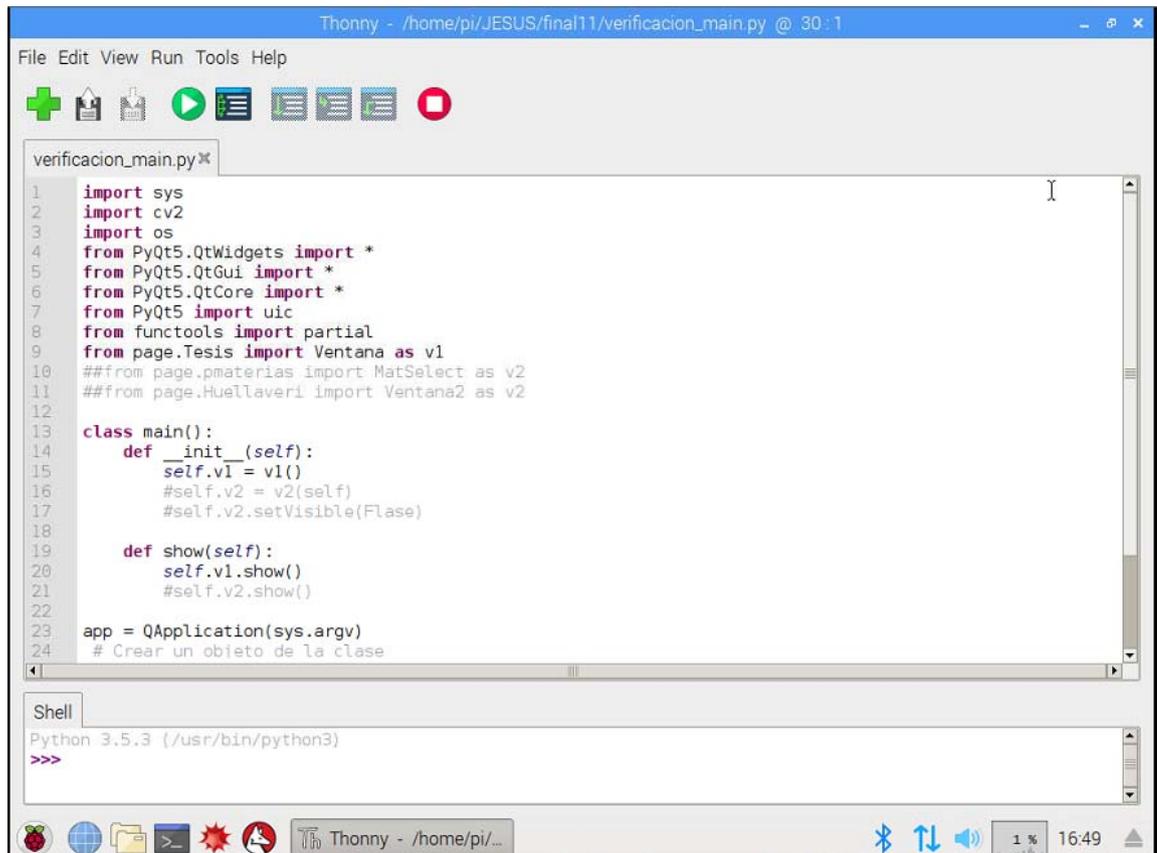


Ilustración 7: Raspberry pi 3



The image shows a screenshot of the Thonny Python IDE. The window title is "Thonny - /home/pi/JESUS/final11/verificacion_main.py @ 30:1". The menu bar includes "File", "Edit", "View", "Run", "Tools", and "Help". The toolbar contains icons for file operations and execution. The main editor displays the following Python code:

```
1 import sys
2 import cv2
3 import os
4 from PyQt5.QtWidgets import *
5 from PyQt5.QtGui import *
6 from PyQt5.QtCore import *
7 from PyQt5 import uic
8 from functools import partial
9 from page.Tesis import Ventana as v1
10 ##from page.pmaterias import MatSelect as v2
11 ##from page.Huellaveri import Ventana2 as v2
12
13 class main():
14     def __init__(self):
15         self.v1 = v1()
16         #self.v2 = v2(self)
17         #self.v2.setVisible(False)
18
19     def show(self):
20         self.v1.show()
21         #self.v2.show()
22
23 app = QApplication(sys.argv)
24 # Crear un objeto de la clase
```

Below the editor is a "Shell" terminal window showing "Python 3.5.3 (/usr/bin/python3)" and a prompt ">>>". The system tray at the bottom shows various icons, including the Raspberry Pi logo, and the time "16:49".

Ilustración 8: Programa Thonny

En el Programa Thonny que trae el sistema operativo de Raspbian se descarga un aplicativo para el lector de huella que verifica el funcionamiento en el sistema embebido (Raspberry), permite ingresar nueva data, verificar e identificar. Además de instalarse el driver del lector de huella y las librerías del lector de huella (Digital persona U.are.U 4500). para hacer el reconocimiento en el programa.

CAPÍTULO 3. DISEÑO DE BASE DE DATOS EN TIEMPO REAL Y EL FRONTEND DE VISUALIZACIÓN WEB PARA EL REGISTRO DE LA INFORMACIÓN DEL SISTEMA DE CONTROL DE ASISTENCIA.

El proyecto se enfocó en un sistema de control de acceso y control biométrico debido a que este posee un grado de vulneración muy bajo, ya que el carné o llave de acceso la tenemos incorporada dentro de nosotros o es propia de cada uno, este sistema tiene un costo razonable teniendo en cuenta el nivel de seguridad que se presta.

El sistema de reconocimiento biométrico está basado en el procesamiento digital de una imagen de una huella dactilar de 256 x 300 píxeles, la cual se toma por medio de un sensor capacitivo, y luego se transmite a un computador (PC) por medio de una interfaz de comunicación. En este trabajo se presenta el diseño de un sistema donde se configura el sensor y se obtiene la imagen para comunicarla al PC. El procesamiento de la huella por medio de algoritmos de filtrado de imágenes propios; el primero de los filtros resalta los valles y crestas de la huella, y se aplica a la imagen por regiones, el siguiente algoritmo se encarga de filtrar la imagen para extraer las características más importantes, luego se lleva la imagen a sólo dos niveles de gris, las crestas a un ancho de un solo píxel (esqueletización), y por último el reconocimiento, se hace hallando la ubicación de las características o minucias de cada huella y las distancias entre ellas, generando vectores de información.

El proceso sería el siguiente:

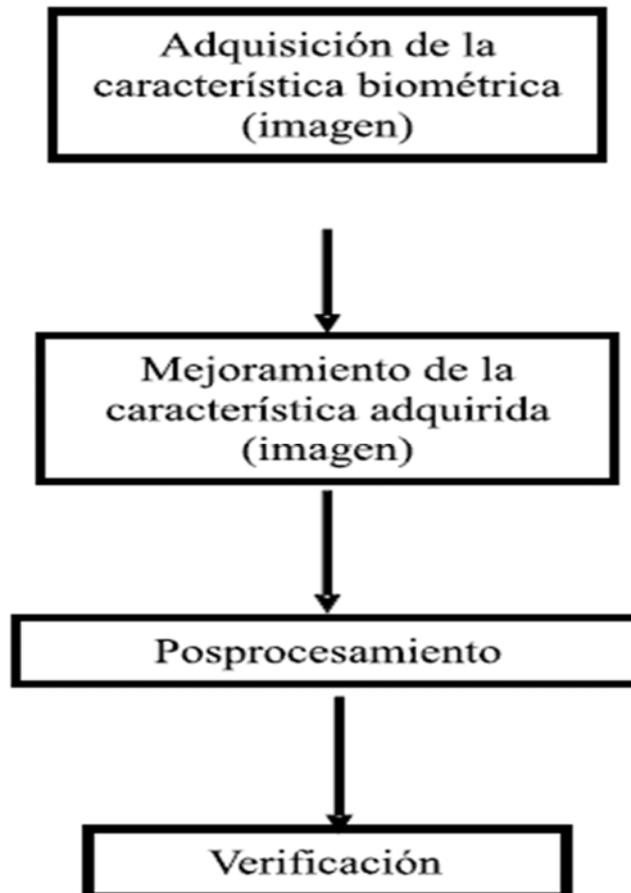


Ilustración 9Diagrama de flujo del sistema de reconocimiento biométrico

Mediante el siguiente diagrama de flujo se estructuran los eventos que permiten la visualización de la base de datos y control de la asistencia a los docentes:

A continuación, se describe el proceso mediante las imágenes reales de la base de datos en la página web y el funcionamiento de la misma.

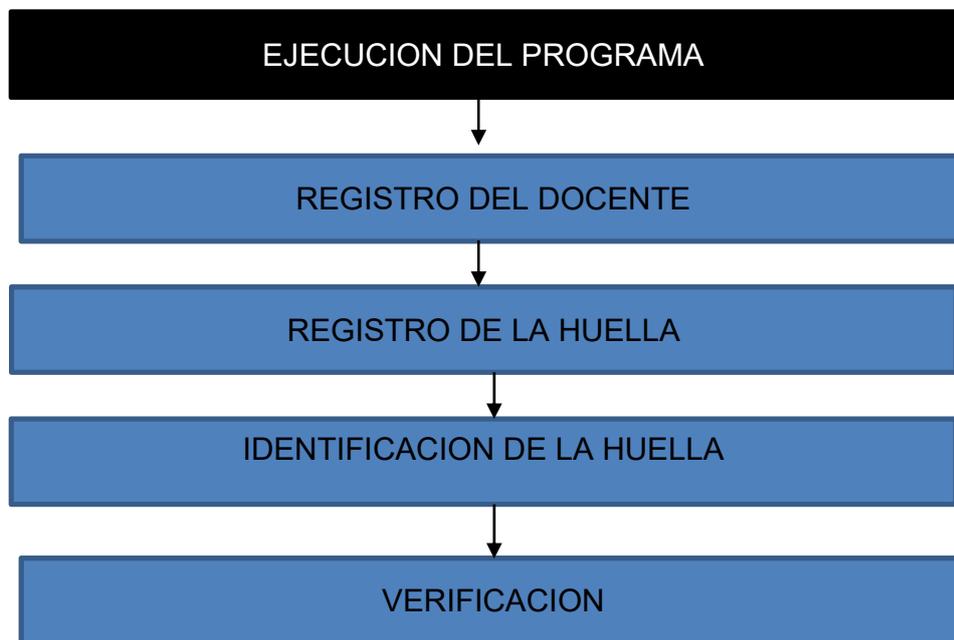


Ilustración 10 proceso prototipó biometrico

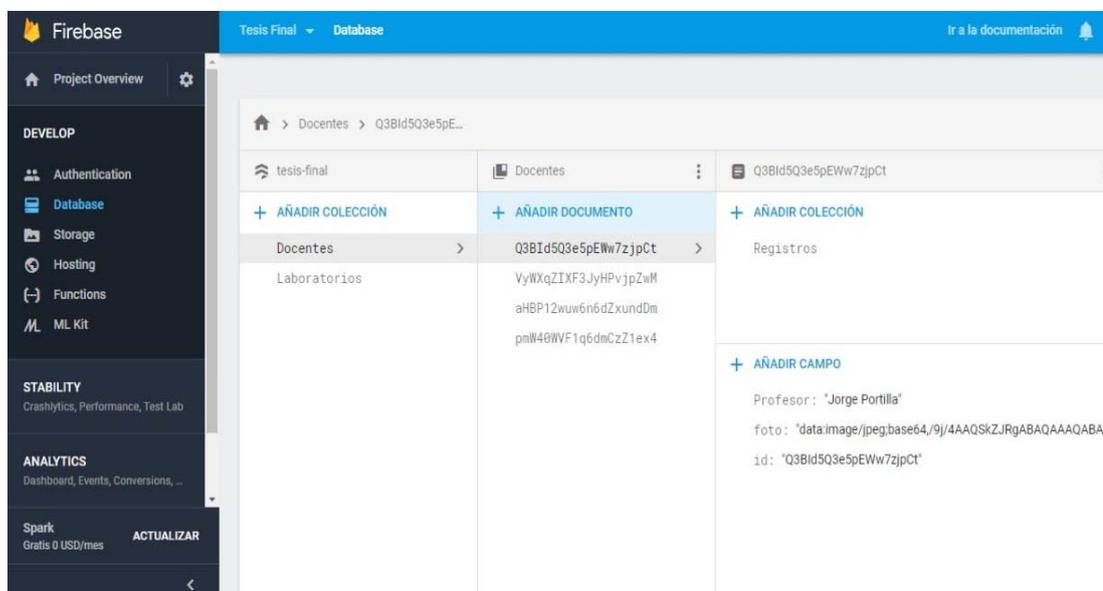


Ilustración 11 Pagina Firebase

Se utiliza la página Firebase que permite tener una base de datos en internet en tiempo real para realizar la comunicación con el Programa Thonny, para modificarla en tiempo real, la hace visible en la página web diseñada,



Ilustración 12: Programa Visual Studio code

Se instaló un programa Visual Studio code, permite programar la página web frontend con Angular.

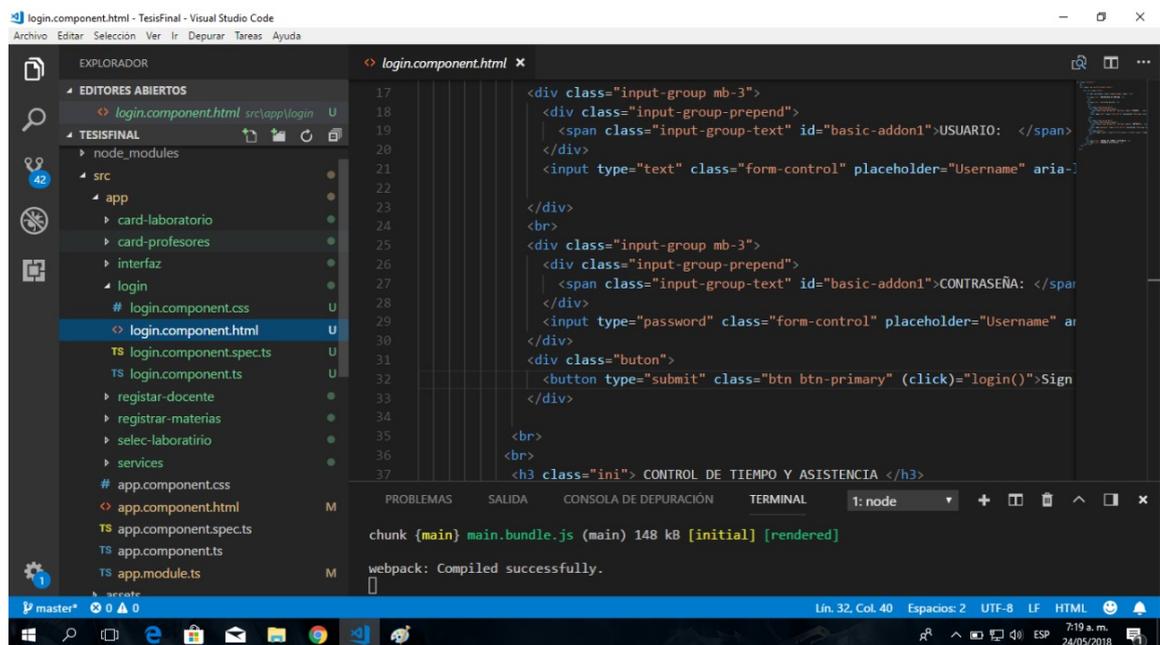


Ilustración 13 Programación del Login en Visual Studio code



Ilustración 14: Frontend del login en Angular

En principio se solicita usuario y contraseña como toda página web personalizada, se diseñó el frontend que permita que el usuario que registra en el sistema biométrico de control sea solo una persona. (Encargado de la página web)

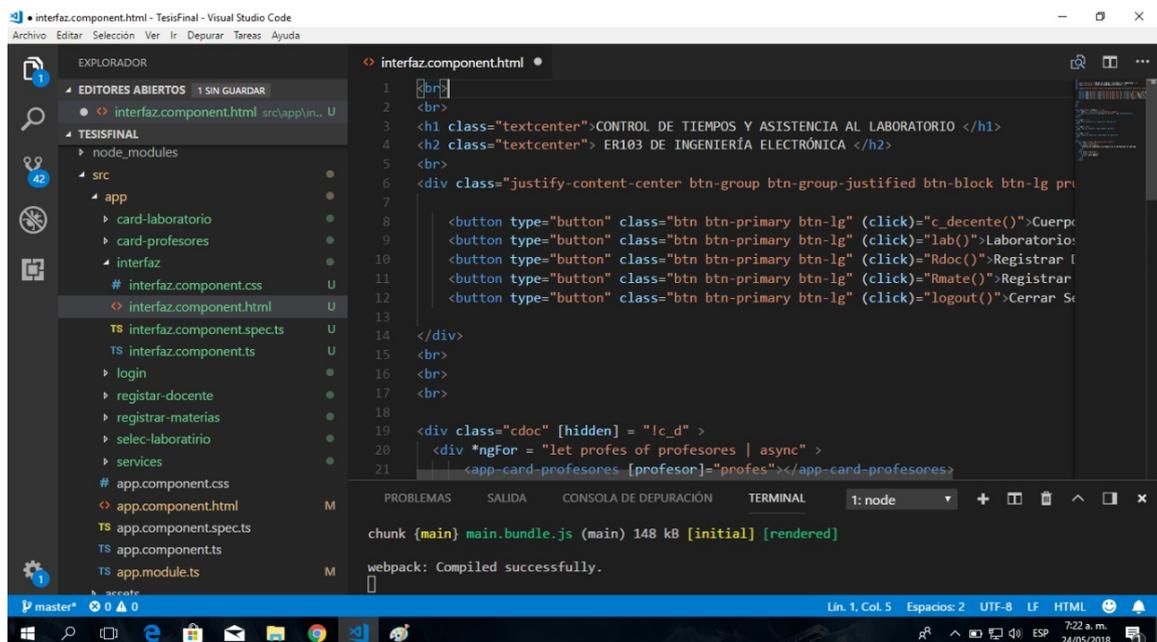


Ilustración 15: Programación de la interfaz en Visual Studio code

Se selecciona la programación del interfaz donde se visualiza las opciones para trabajar.

CONTROL DE TIEMPOS Y ASISTENCIA AL LABORATORIO ER103 DE INGENIERÍA ELECTRÓNICA



Ilustración 16: Visualización de la interfaz en web frontend

Visualización de la interfaz en web Frontend

Visualización de la interfaz de la página web Frontend: Interfaz donde se observan las ventanas con las distintas opciones como Cuerpo docente, Laboratorios Registrar docente, registrar materia y el cierre de sesión.

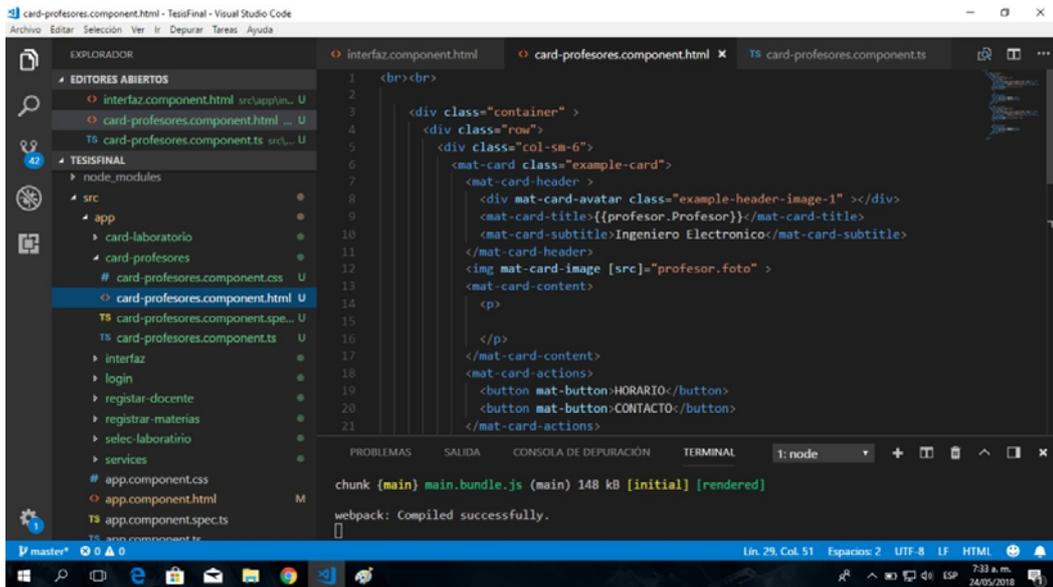


Ilustración 17: Programación de las card del cuerpo docente en Visual Studio code

La web frontend Contiene las funciones, así: (cuerpo docente) visualiza desde la card las fotos y nombres de los docentes, para ir directamente al horario o los contactos.



Ilustración 18: Visualización de las card del cuerpo docente

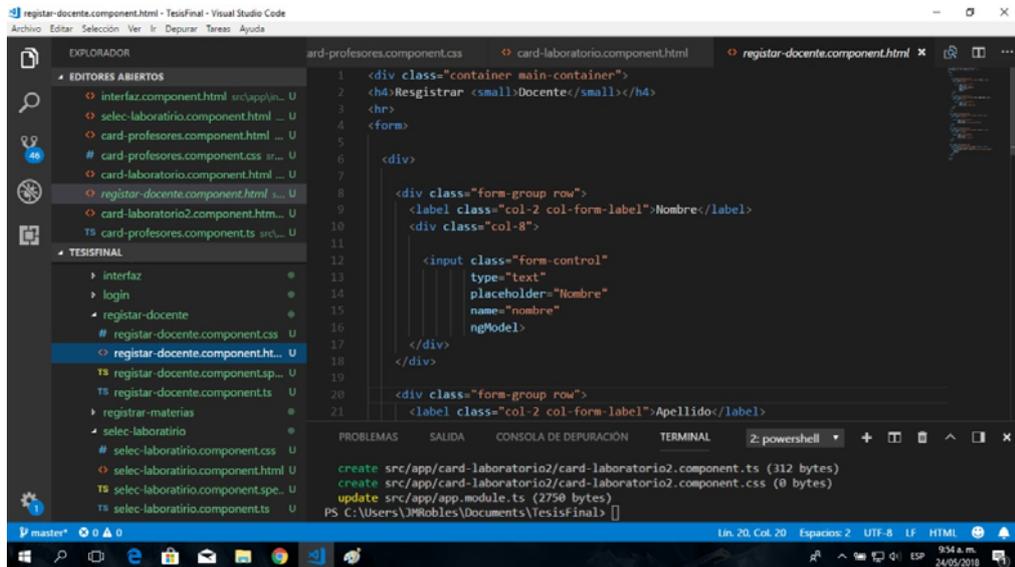


Ilustración 19: Programación del Registro de docentes en Visual Studio code

Programación del Código para la visualización del registro de los docentes.

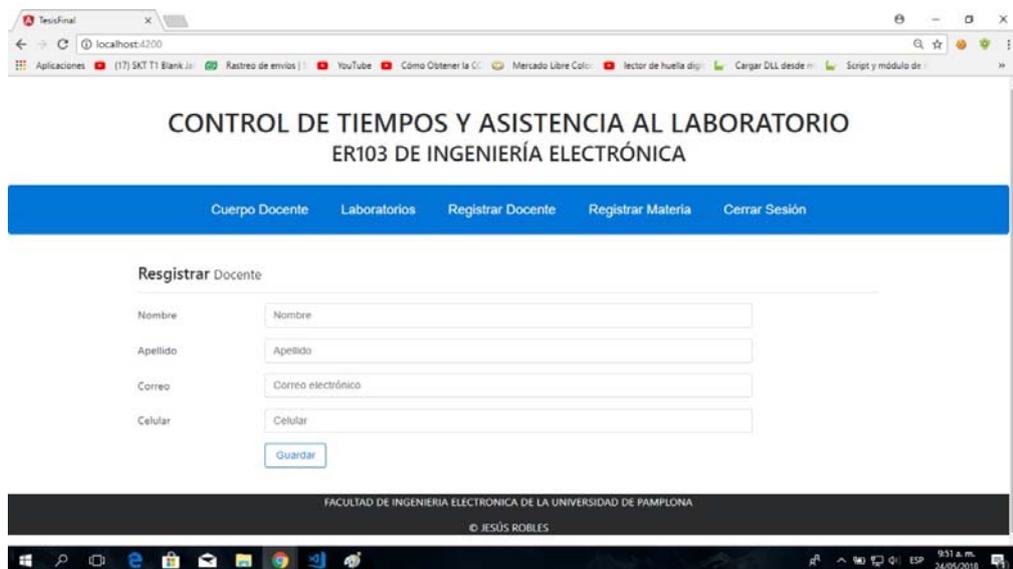


Ilustración 20: Registro de docentes en la web frontend de angular

Visualización frontend del registro de los docentes para realizar consultas sobre la base de datos (celular).

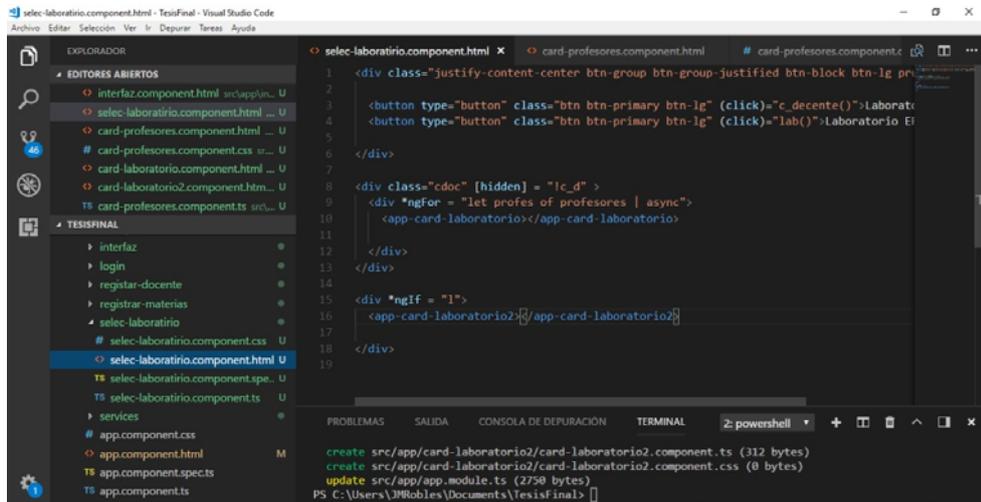


Ilustración 21: Programación de los botones de la Selección de laboratorios en Visual Studio code

De la misma manera al seleccionar opción laboratorio o cualquier otra de las funciones.

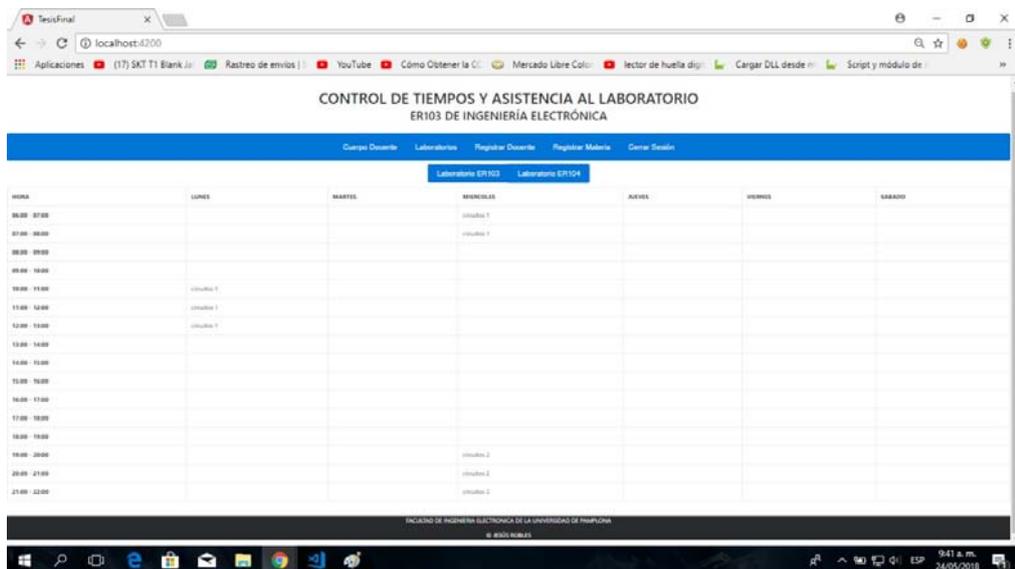


Ilustración 22: Visualización de la selección de laboratorio de frontend en angular

Visualización frontend.

Aparecen los dos laboratorios en la selección de la base de datos, para obtener la actualización en tiempo real de cada uno de ellos. (Ejemplo Laboratorio ER103 de ingeniería electrónica) Horas y fechas asignadas

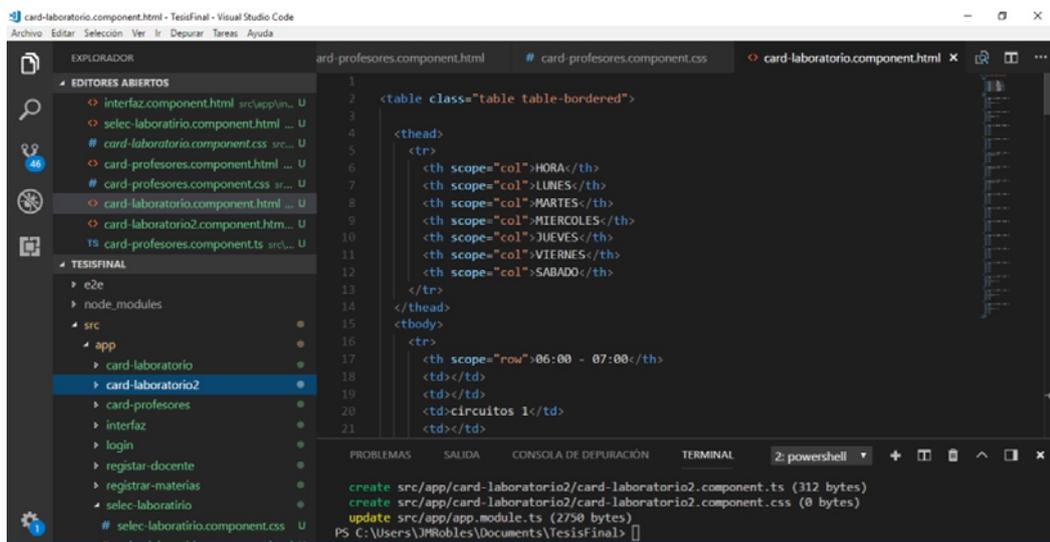


Ilustración 23: Programación de las card de los laboratorios en Visual Studio code

Ventanas separadas de cada uno de los laboratorios para programar horarios y fechas.

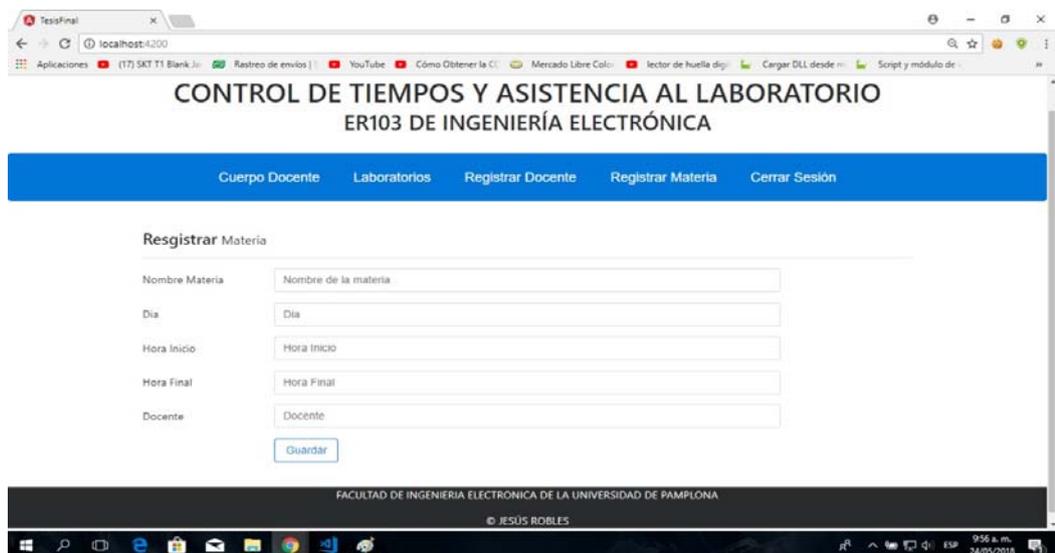


Ilustración 24: Visualización frontend del registro de materias

Visualización frontend del registro de materias. Contiene calendario, docentes, hora de inicio y hora final.



Ilustración 25: cierre de sesión

Y por último el cierre de sesión para mantener seguros los datos.

Aplicación para registro de profesores en el prototipo.

El dispositivo de control de acceso, es programado desde un dispositivo embebido (Raspberry), en lenguaje Python. Debido a la versatilidad y el gran auge que este tiene. El registro deberá ser usado por un “Super Usuario” (Super Us) el cual se configura desde el código fuente, por razones de seguridad.

La aplicación tiene tres funciones principales: Nuevo Registro, el cual va a ser el encargado de registrar a un nuevo usuario; Buscar, que se encargara de buscar en la base de datos al profesor; y Salir, el cual cierra la aplicación. Para todos los casos, previamente se tiene que acceder con permiso de Super Us, como se muestra en el diagrama del flujo del programa (*Ilustración 26*).



Ilustración 26 Diagrama de flujo del inicio del programa.

En caso de que no tenga permisos de Super Us, el usuario no podrá acceder y siempre se visualizará la pantalla de la *Ilustración 27*.



Ilustración 27 Pantalla de inicio del programa con las opciones de uso.

Para comprobar el Super Us, el programa desplegará un teclado donde se ingresa el documento del usuario, si corresponde el dato con el de Super Us, el programa pedirá la comprobación por huella dactilar, el proceso de esta parte se muestra en la *Ilustración 28*.

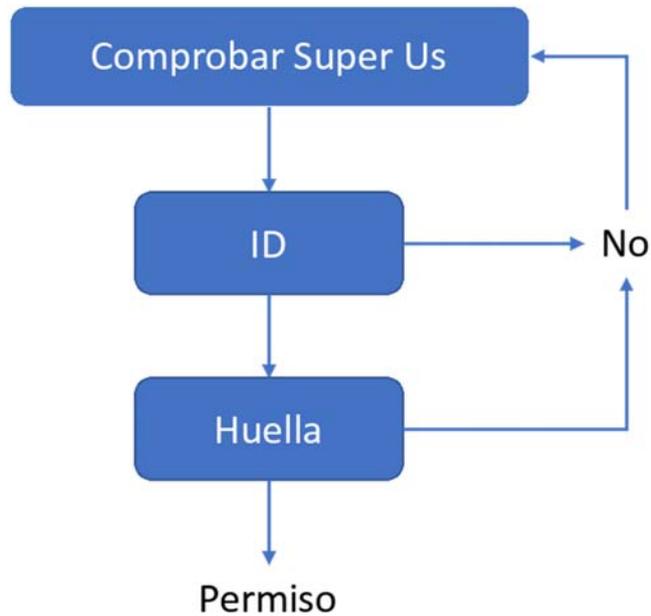


Ilustración 28 Diagrama de flujo de la comprobación de la huella.

Una vez se accede con permisos de Super Us, el usuario podrá ingresar hacer uso de sus opciones. En caso de que la opción elegida sea la de *Nuevo Registro*, aparecerá la pantalla de la *Ilustración 29*. Esta fase de la aplicación permitirá llenar los campos de los datos del nuevo usuario, así como cargar una foto y registrar la respectiva huella del nuevo usuario, la cual se tendrá que ingresar 5 veces, una vez suceda esto se podrán registrar los datos en la base de datos, tanto a nivel local, como en la nube en una base de datos en FireBase; las imágenes y otra información de relevancia será cargada a una cuenta de GoogleDrive que permita cargar y descargar esta información.



Ilustración 29 Pantalla de Nuevo Registro.

El proceso de esta sección del programa se encuentra descrito en la *Ilustración 30*.



Ilustración 30 Diagrama de flujo del Nuevo Registro.

La aplicación no hace el registro en caso de que no hallan datos, y en caso de que no se cargue una foto y no se haga el registro de la huella; para esta última tendrá

que existir datos y foto para que se pueda activar la opción. El botón “Atrás” regresara a la pantalla de inicio.

Para la opción *Buscar*, se despliega un teclado en pantalla en el cual el Super Us ingresará el Documento de Identidad del usuario a buscar y el programa busca entre los usuarios ya registrados si hay coincidencia, en ese caso extraerá los datos del sujeto y los visualizará en pantalla, como se muestra en la *Ilustración 31*. En caso de que no haya coincidencia el programa arroja un mensaje advirtiendo de esto, un ejemplo de los mensajes se puede ver en la *Ilustración 32*.

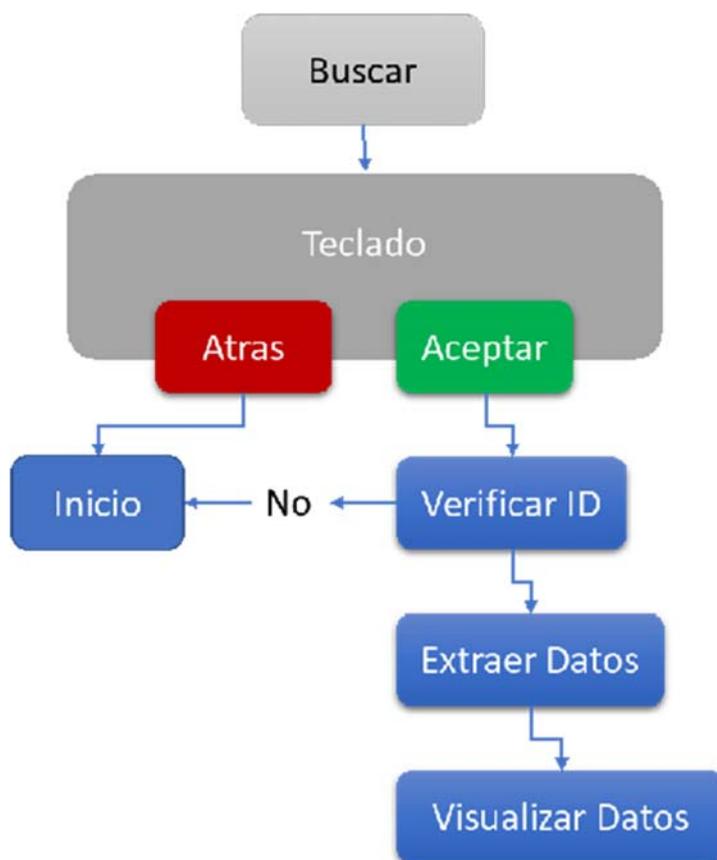


Ilustración 31 Diagrama de flujo de la sección *Buscar* de la aplicación.



Ilustración 32 Ventana de mensajes a lo largo de la aplicación.

Aplicación para verificación de profesores.

Para la verificación de los profesores dispondremos de un programa distinto al de registro. Ya que no se necesita hacer registro nuevo, este programa solo se encarga de hacer la verificación de entrada y salida de los profesores. En este sentido la aplicación cuenta con una interfaz bastante sencilla que permite intuir rápidamente lo que se debe hacer con ella. El programa requiere que previamente se haga el registro del docente si se quiere que la verificación sea exitosa.

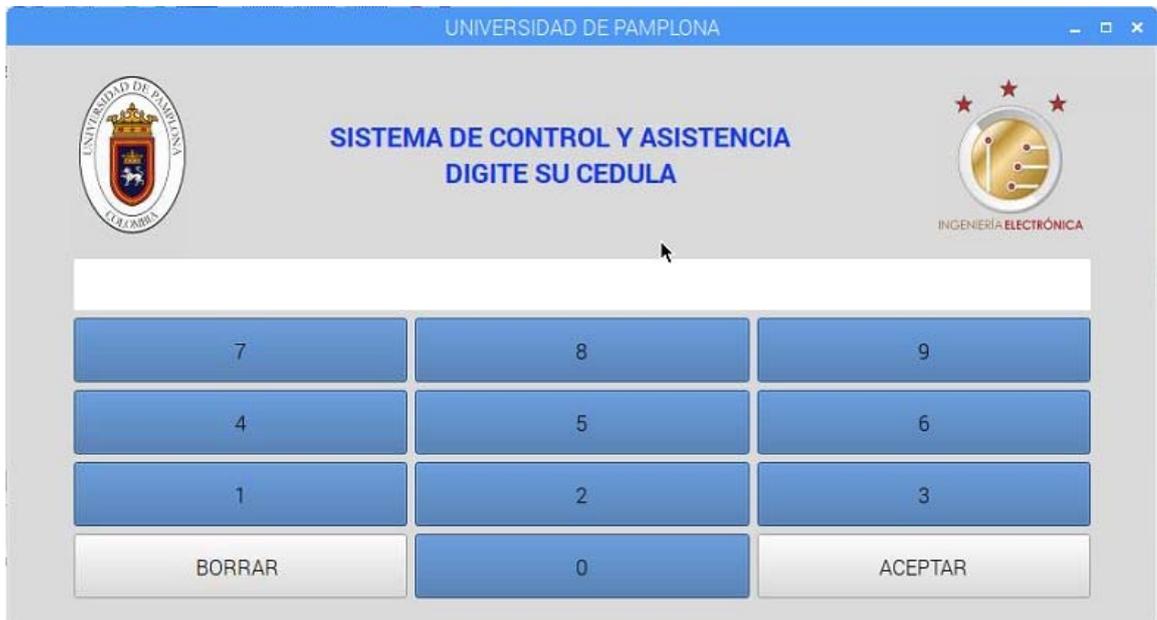


Ilustración 33 Ventana con teclado numérico para ingreso de numero de documento.

Al iniciar el programa, esta lanza una ventana con un teclado numérico donde el docente deberá ingresar su número de documento (Ver *Ilustración 33*). Una vez ingresado el número de documento y pulsar *Aceptar* el programa pasará a verificar si la ID se encuentra en la base de datos, en caso de que no exista volverá al inicio. En caso de que exista entonces pasara a hacer la comprobación de la huella, como se describe en el diagrama de la *Ilustración 34*.



Ilustración 34 Diagrama de flujo de la primera parte del programa de verificación.

Para hacer la comprobación de la huella se le ofrecerán al usuario 5 intentos para que haya coincidencia y se le avisará para que coloque la huella, como se muestra en la *Ilustración 35*. Si no ocurre esto en ninguna de las oportunidades entonces el programa regresa al inicio y cancela el acceso. En caso de que haya coincidencia entonces se otorga el permiso y se puede continuar con la extracción y la visualización de los datos, el comportamiento de esta parte se puede mirar en la *Ilustración 36*.



Ilustración 35 Mensaje para colocar la huella.

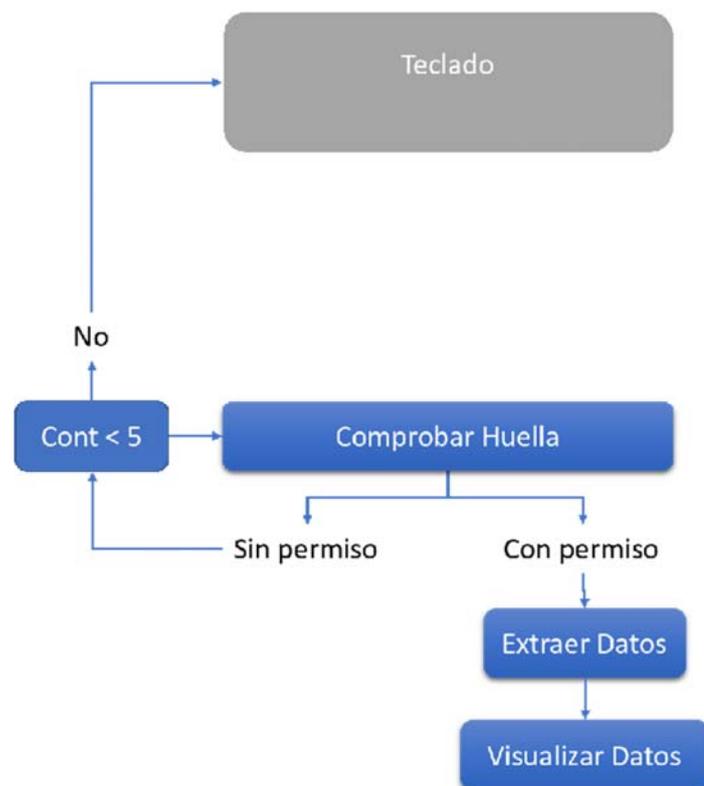


Ilustración 36 Diagrama de flujo de la Comprobación de la Huella para el caso de la verificación.

En el nodo “Extraer Datos” del diagrama de la *Ilustración 36*, la extracción de los datos se hacen desde una base de datos local y remota, que será actualizada inmediatamente en Google Drive.



Ilustración 37 Visualización de Información Docente Verificación

Una vez hecho el ingreso del profesor, permanecerá el sistema abierto sin que se pueda cerrar hasta que el profesor cierre la sesión. El botón “Salir” hará nuevamente la comprobación con las mismas condiciones de la comprobación inicial, como se muestra en el diagrama de la *Ilustración 37*. Luego de esto se registran los datos de ingreso y salida del profesor, así como el salón al que ingreso y en qué horas estuvo. Todo esto se guarda en un archivo personalizado de cada uno de los profesores. A lo largo del programa se visualizarán mensajes de aviso del estado de los procesos como se muestra en la *Ilustración 38*.

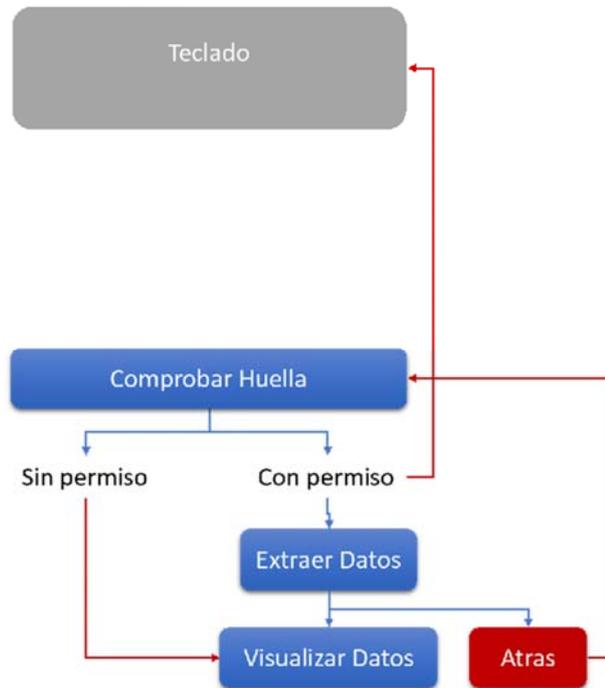


Ilustración 38 Diagrama de flujo del programa cuando se queda en espera del cierre de sesión.

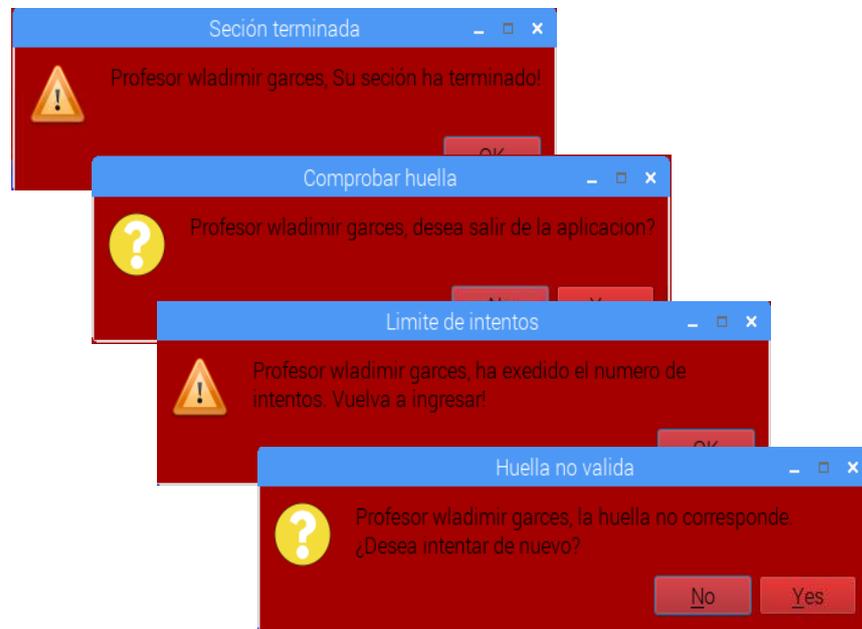


Ilustración 39 Ejemplo de los mensaje que se irán mostrando en el transcurso del programa.

Por último, el diagrama de la *Ilustración 39* muestra el proceso completo de nuestra aplicación.

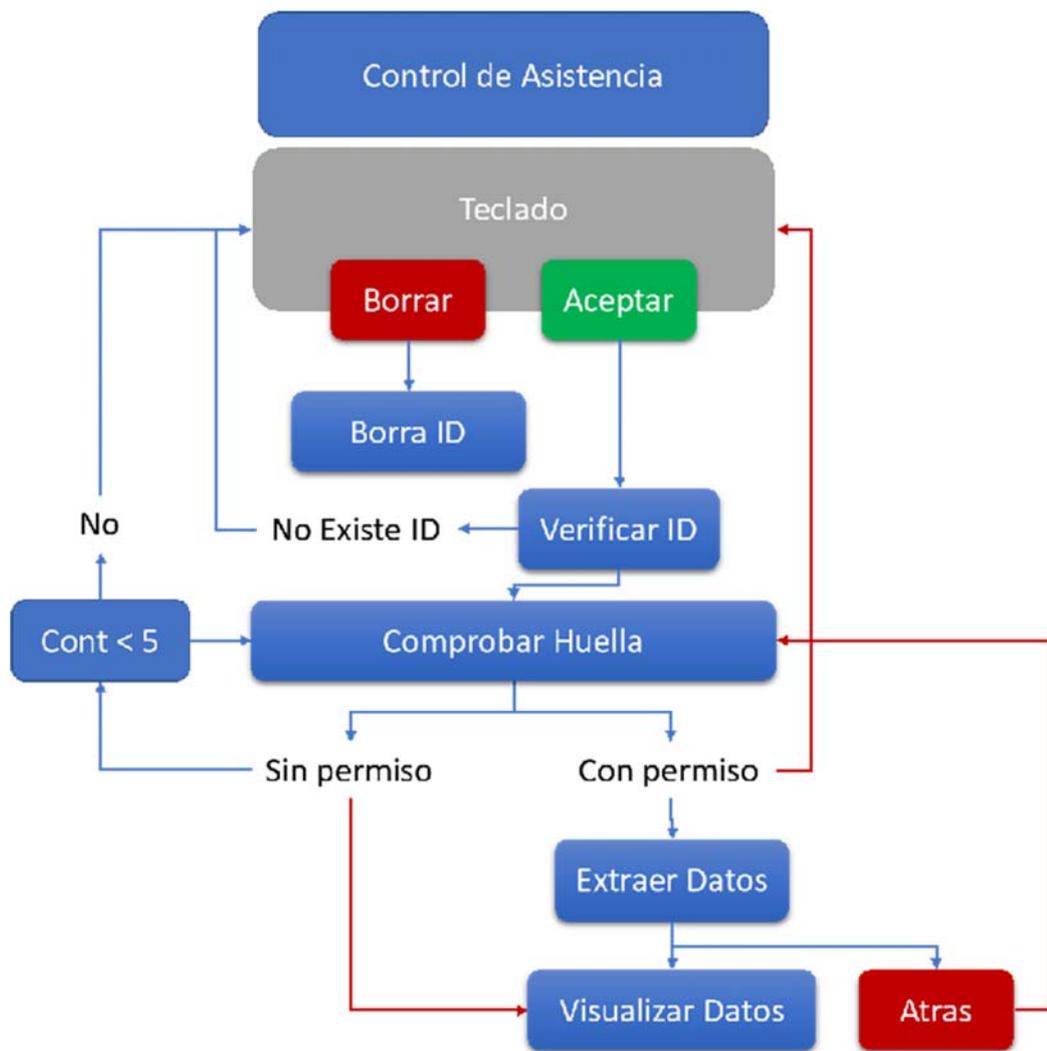


Ilustración 40 Diagrama de flujo del programa de verificación completo, las líneas rojas implican un sub-proceso.

CAPÍTULO 4. VALIDACIÓN DEL FUNCIONAMIENTO DEL PROTOTIPO FINAL DEL SISTEMA DE CONTROL DE ASISTENCIA EN LOS LABORATORIOS DE INGENIERÍA ELECTRÓNICA

El prototipo ofrece grandes ventajas como el reconocimiento de los patrones de huella en un lapso de tiempo corto, su tamaño es reducido y permite facilidad de instalación en recintos, realiza un escaneo óptico y un switcheo rápido entre usuarios

Las pruebas de validación constan de la toma de huella a diferentes individuos, los cuales realizan el proceso de guardar su huella en el módulo y luego autenticarse, se debe observar qué factores externo influyen en el reconocimiento de la huella por el módulo, (tamaño del dedo, que dedo autenticar para permitir una posición más cómoda al usuario al momento de acercar el dedo al módulo y se puede realizar una detección del 100%). Es aconsejable que la posición del dedo cubra toda la ventana de detección de la imagen del módulo, ya que si se encuentra ladeado o no cubre toda la superficie y además no está limpio y seco posiblemente el sensor no podrá leer bien la imagen del dedo o simplemente obtendrá una imagen diferente a la huella original.

Debe tenerse en cuenta el tiempo de respuesta de cerca de 5 segundos, en la que se demora tomando la imagen, realizar los debidos procesos, y enviado la información pertinente a la Raspberry Pi 3, Model B.

Prototipo final

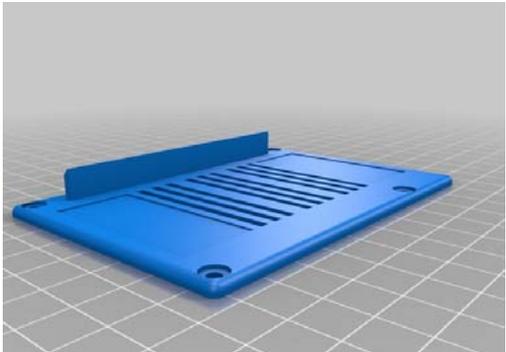
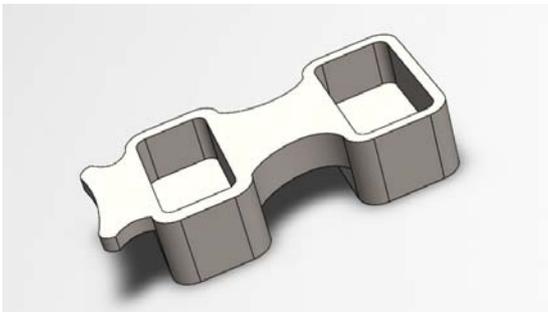
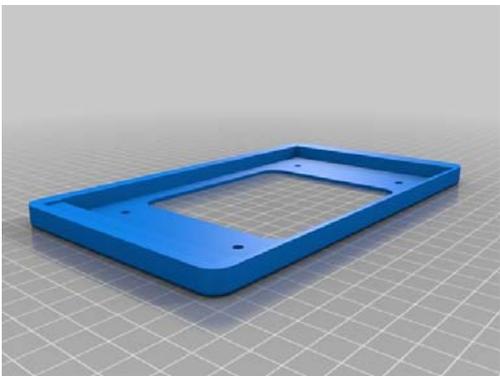
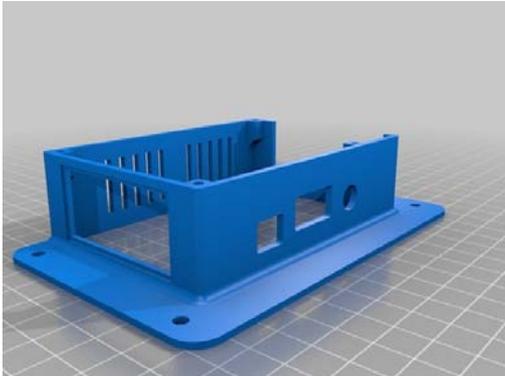


Ilustración 41: Sistema de control biométrico



Ilustración 42: Raspberry PI 3 Model B

El dispositivo se conforma de la Raspberry Pi, el cual tiene su propio software (raspbian),

Las Partes de este sistema embebido:

1. BSP (*Board Support Package*), que contiene el kernel de Linux, el bootloader, con los drivers de los dispositivos físicos de la placa a usar.
2. Aplicaciones
3. Sistema de integración: que une las dos anteriores, las librerías y las aplicaciones al entorno de trabajo del sistema.

Se diseñó una Base ergonómica con lector de huella empotrado, con el objeto de mantener firme el lector de huella dactilar y además no causar problemas de túnel carpiano entre otros a los usuarios de la base de datos, en este caso los docentes quienes posiblemente deban colocar su dedo más de una vez al día, dependiendo de su horario.



Ilustración 43: Base ergonómica con lector de huella empotrado

Como se dijo antes el Lector de Huella, se instala en un espacio ergonómico, La identidad única de una huella digital ofrece una seguridad absoluta, mientras que proporciona facilidad de uso a través de su simplicidad innata.

Con su diseño innovador, robusto y fácil de usar, el lector de huellas digital biométrico, se programa a través de una simple configuración con driver de instalación para que el software de la Raspberry lo pueda reconocer y recibir la información que proporciona.

Este biométrico funciona a través de un dispositivo CCD o *Charged Coupled Device*, el cual cuenta con un arreglo de diodos sensible a la luz que transmiten una señal eléctrica en respuesta a fotones de luz. Cada diodo es capaz de grabar un pixel, un pequeño punto que representa la luz que le es reflejada.

En conjunto, la luz y perfiles oscuros forman una imagen de la huella leída. En si, el proceso de lectura comienza cuando se presiona el dedo sobre el lector, el cual ya trae incorporado su propia fuente de iluminación, normalmente un juego de LED's para iluminar las vallas de la huella digital.

El lector posteriormente se encarga de asegurar que haya registrado claramente la huella, por medio de la luz apropiada, para continuar con la oportuna comparación

de la información guardada y se verifica en la Raspberry la obtención de los datos de la huella y el usuario.

Validación entre firebase y la página web frontend

Es una base de datos en tiempo real, para que la información este de forma síncrona en todos los dispositivos que están conectados a ella, Las bases de datos convencionales SQL no proporcionan esta funcionalidad, Adicional es una base de datos offline y online, Si el internet llega a faltar puede realizar registros y cuando vuelva el internet actualiza toda la información, Proporciona sistemas de autenticación que hacen que la seguridad de la información no sea vulnerada, Adicional firebase proporciona múltiples servicios con los cuales se pueden a futuro expandir las funcionalidades del prototipo.

Se realiza la comunicación entre la firebase y la visualización de la página web frontend, cuando haya una modificación en la visualización de la página web, podemos visualizar en la página de firebase que se hicieron los cambios pertinentes y viceversa.

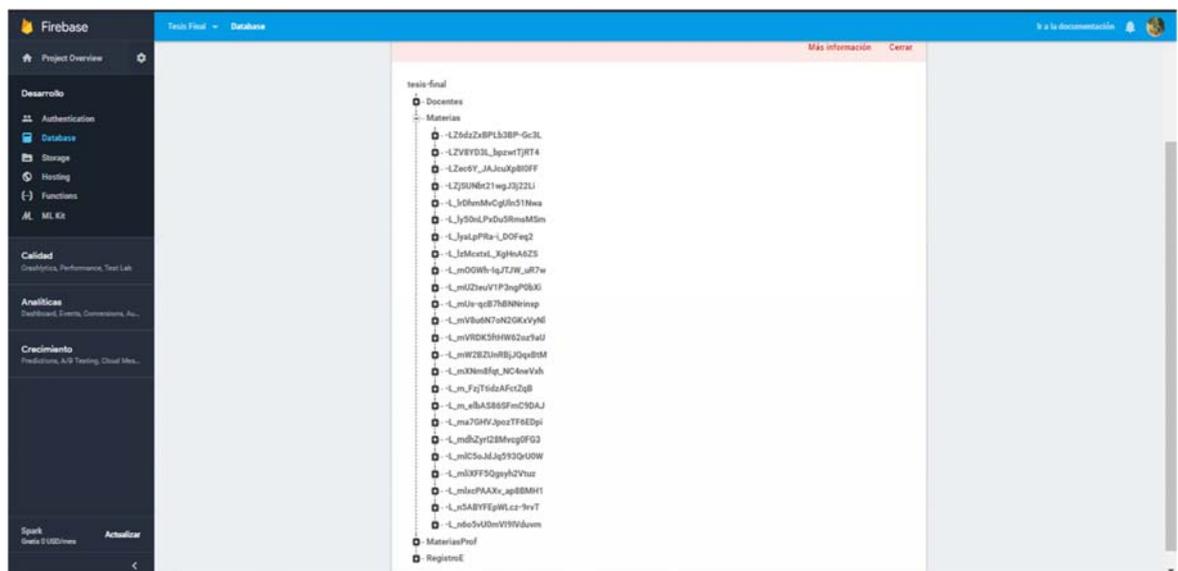


Ilustración 44 base de dato firebase

Validación entre sistema embebido (Raspberry) y firebase

En la siguiente tabla se muestra una comparativa de las características más importantes tanto de la Raspberry bi3 B como las de sus primeras rivales, todas con sistemas operativos base Unix.

	RASPBERRY PI3 B	OLINUXINO A20	CUBIEBOARD2	ODROID XU3	BEAGLE BOARD BLACK
FRECUENCIA PROCESADOR	1200 MHz	1000 MHz	1000 MHz	2000 MHz	1024 MHz
RAM	1024 GB	512 MB	1024 MB	2048 MB	512 MB
N° NÚCLEOS	4	2	2	8	2
N° USB	4	2	2	5	1
WI-FI BLUETOOTH	Si/Si	No/No	No/No	No/No	No/No
SATA	No	Si	Si	No	No
GPIO	40	160	67	30	96
PRECIO	140.000	555,501	350.990	506.900	332.800

La elección de la Raspberry Pi3 Modelo B se debe a su menor precio, cantidad de puertos USB y conexión a WIFI, ya que son las características que se requiere para la realización del proyecto.

En el programa de Python que trae incorporado el software de Raspbian donde se encuentra la programación del lector de huella, se verifica que cuando exista el

registro de una primera huella se genere también el registro en firebase del tiempo de iniciación y cuando nuevamente registre la misma huella también registre el tiempo de finalización.

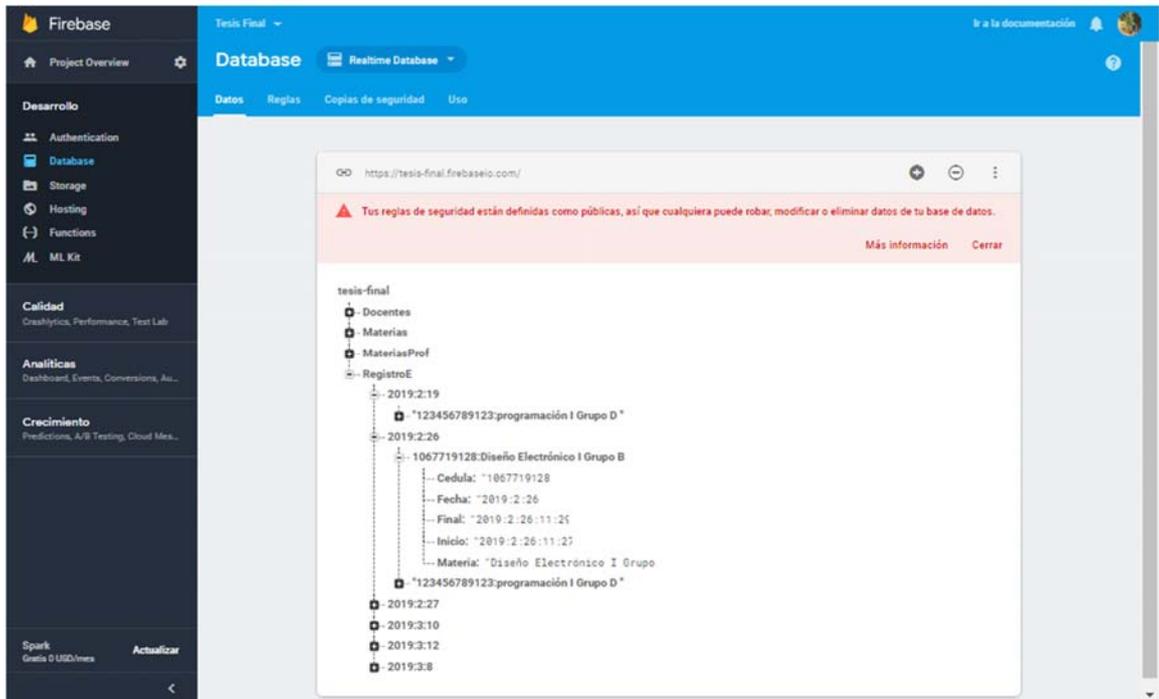


Ilustración 45 validación de registro de asistencia

Cotización prototipo:

N°	material	Precios
1	LCD Touch de 7 Pulgadas	315.000
2	Raspberry bi3 B	140.000
3	digital persona u.are.u 4500	150.000
4	impresión 3D	95.000
5	Mano de obra	200.000
	Precio total	900.000

TABLA COMPARATIVA

En la siguiente tabla se muestra una comparación del prototipo final con respecto a otros sistemas de control de asistencias comerciales.

			
huellas	indefinido	2000	1500
USB	4	1	1
PANTALLA	LCD Touch de 7 Pulgadas	LCD	LCD Touch de 2.8 Pulgadas
COMUNICACIÓN	TCP/IP, WIFI	TCP/IP	TCP/IP
PRECION	900.000	344.000	539.000
FUENTE DE ALIMENTACIÓN	DC 5V 2.5A	USB/5V DC	DC 12V 3A

VENTAJAS:

- Pantalla más grande.
- Conexión wifi.
- Mayor cantidad de puerto USB.
- Registro indefinido de usuarios.
- Sincronización con múltiples puntos.

DESVENTAJAS:

- Mayor precio.
- Conexión internet obligatoria.

DISCUSIÓN Y RESULTADOS

Los sistemas embebidos están tomando cada vez más importancia en nuestra vida cotidiana, pues contribuyen a elevar la productividad y competitividad en un mundo cada vez más globalizado.

La reducción en los costos de los componentes electrónicos ha aumentado el número de plataformas de hardware que soportan el sistema operativo, las cuales son muy comunes en productos comerciales; una de ellas es Raspberry Pi 3 Modelo B.

El análisis de distintos lectores de huella dactilar que pueden ser implementados posteriormente en el desarrollo de sistemas de bajo costo basados en este indicador biométrico utilizando una Raspberry P es importante para la puesta en práctica de proyectos como este.

Por otro lado, la precisión de un sistema biométrico está determinado por una serie de pruebas, que están divididas en tres categorías: tecnología, escenario y operacional y para su evaluación se consideran varios conceptos que se pueden generalizar en la probabilidad de que alguien autorizado sea rechazado y la probabilidad de que alguien no autorizado sea aceptado,

Factibilidad en otro campo

La relevancia de este trabajo puede consolidarse en los siguientes aspectos:

Tecnológico: El proyecto de grado contribuye a la mejora del sistema académico de la Universidad de Pamplona, aplicando la huella digital como método de control de asistencia en la base de datos de los laboratorios

Operativo: Permitirá contar con la información de cada usuario (docente), para saber que la fecha, hora de entrada y hora de salida a los laboratorios, la puntualidad y el cumplimiento de las funciones en los docentes.

Social: Tanto los estudiantes, como los docentes y administrativos de la universidad de Pamplona podrán estar beneficiados con un servicio de control de clases en laboratorio, mejorando así la calidad de la educación por medio del cumplimiento de las funciones de su personal.

CONCLUSIONES

Después de finalizado el prototipo y realizadas las pruebas necesarias de todo el sistema, se puede concluir que:

Durante el desarrollo del proyecto, al extraer desde el sensor la imagen digital de la huella y vectorizarla, podría llegarse a comunicar dos prototipos que compartan una base de datos de registro de huellas.

Según los resultados obtenidos, se puede decir que el proyecto cuenta con una alta confiabilidad en la autenticación de los usuarios, pero un poco demorado para el usuario que no puede completar exitosamente el proceso de registro y autenticación a la primera vez, esto se ve afecto por situaciones externa, principalmente por el estado de la red o por falta de energía para el sistema embebido.

En cualquier equipo con acceso a internet se puede visualizar la página web del sistema de control y asistencia.

La comprobación de los sistemas utilizados muestra que tienen un funcionamiento correcto tanto en el dispositivo de reconocimiento de la huella como en el sistema de programación.

RECOMENDACIONES Y TRABAJOS FUTUROS

Las recomendaciones de la parte práctica sobre aspectos por mejorar si alguien decide en un futuro realizar un avance, comparación, rectificación o rebatimiento al tema estudiado, así como una propuesta de los posibles trabajos futuros que complementarían o profundizarían este desarrollo serían:

- Crear un sistema que permita tener energizado el sistema cuanto se presente un fallo de energía en los laboratorios.
- Diseñar la base de datos para los laboratorios faltantes.
- Incorporar un sistema de control de acceso a cada laboratorio sincronizados entre ellos.

Es de vital importancia tener una conexión constante y segura a internet que permita hacer la subida de la base de datos a la interfaz gráfica web frontend; de lo contrario no se podrá visualizar la información actualizada en la página web frontend en tiempo real.

REFERENCIAS BIBLIOGRÁFICAS

Aviles López Javier. Desarrollo de una aplicación web para compartir medio de transporte con AngularJS. Universidad Politécnica de Cartagena. 2014.

Crespo Claudia Marcela y Fernández Ernesto. Diseño e implementación de un scanner 3D para prototipado y modelado de objetos. Universidad autónoma de Barranquilla. 2014.

Restrepo Jhony. Revisión de las técnicas básicas para el reconocimiento de rostros. Universidad Tecnológica de Pereira. Risaralda. 2012. Recuperado de: <http://repositorio.utp.edu.co/dspace/bitstream/handle/11059/6600/0053682R436.pdf?sequence=1>

Maya Adriana. (2013) Sistema biométrico de reconocimiento de huella dactilar en control de acceso de entrada y salida. Universidad Militar Nueva Granada. . Bogotá D.C

Tolosa Cesar & Giz Álvaro. (2008) Sistemas Biométricos. Disponible en: https://www.dsi.uclm.es/personal/MiguelFGraciani/mikicurri/Docencia/Bioinformatica/web_BIO/Documentacion/Trabajos/Biometria/Trabajo%20Biometria.pdf

Aguilar G, Sánchez G, Toscano K, Nakano M, Pérez H. (2008) Reconocimiento de Huellas Dactilares Usando Características Locales. Revista Facultad de. Ingeniería Universidad de Antioquia N.º 46 pp. 101-109. Diciembre, 2008

Pusiol Guido. (2007) Sistema de verificación de huellas digitales.

Román Juliana. & Gonzales Kelly (2013) Sistemas embebidos y Hardware libre. Escuela de Ingeniera de Sistemas e Informtica Universidad Industrial de Santander Bucaramanga, Santander. Disponible en: <http://wiki.sc3.uis.edu.co/images/e/e1/GR7.pdf>

Hernández Vega, José Isidro. (2010) El Software Embebido y los Retos que Implica su Desarrollo Conciencia Tecnológica, núm. 40, julio-diciembre, 2010, pp. 42-45 Instituto Tecnológico de Aguascalientes Aguascalientes, México

Diaz Suarez Daniel. 2015. ReactJS.

<https://www.adictosaltrabajo.com/tutoriales/introduccion-react/>

Krall Cesar. (2006-2029) Programación Webapp Javascript. Librerías, Frameworks. JQuery, Angularjs. Ventajas. Diferencias.

(Cu01194e) APRENDERAPROGRAMAR.COM. Descargado de: http://www.aprenderaprogramar.com/index.php?option=com_attachments&task=download&id=803

Revilla Danel (2016) Evaluación del uso de librerías modelo-vista-controlador para el desarrollo de servicios web. caso de estudio: REACT. Universidad Politecnica de Madrid. Descargado de: http://oa.upm.es/40491/1/PFC_DANIEL_REVILLA_TWOSE_2016.pdf

http://www.semanticwebbuilder.org.mx/en_mx/swb/Sistemas_Embebidos_Innovando_hacia_los_Sistemas_Inteligentes

<https://www.beeva.com/beevea-view/tecnologia/la-evolucion-del-framework-angularjs/>

http://programacion.net/articulo/angularjs_vs_jquery-cuales_son_sus_mayores_diferencias_1610

Zorita Danilo Simón. Reconocimiento automático mediante patrones biométricos de huella dactilar. Universidad Politécnica de Madrid. España. 2003