Diseño e implementación de un sistema de identificación de personas basado en servicios de reconocimiento facial en la nube

Obando Zapata, Saúl Marcelo

Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Ingeniería en Electrónica y Telecomunicaciones

Trabajo de titulación, previo a la obtención del título de Ingeniero en Electrónica y Telecomunicaciones

Ing. Alulema Flores, Darwin Omar PhD.

22 de diciembre del 2021

COPYLEAKS

Para_Antiplagio_Tesis_Saul_Obando.pdf

Scanned on: 15:37 December 22, 2021 UTC

DARWIN OMAR
ALULIMA
FLORES

3.6%	19	23990
Overall Similarity Score	Results Found	Total Words in Text

Identical Words	397
Words with Minor Changes	93
Paraphrased Words	377
Ommited Words	0

DEPARTAMENTO DE ELÉCTRICA, ELECTRÓNICA Y TELECOMUNICACIONES

CARRERA DE INGENIERÍA EN ELECTRÓNICA Y TELECOMUNICACIONES

CERTIFICACIÓN

Certifico que el trabajo de titulación, "**Diseño e implementación de un sistema de identificación de personas basado en servicios de reconocimiento facial en la nube**" fue realizado por el señor **Obando Zapata, Saúl Marcelo** el cual ha sido revisado y analizado en su totalidad por la herramienta de verificación de similitud de contenido; por lo tanto cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, razón por la cual me permito acreditar y autorizar para que lo sustente públicamente.

Sangolquí, 17 de diciembre del 2021

Firmado electrónicamente por:
DARWIN OMAR ALULEMA FLORES

.....

Ing. Alulema Flores, Darwin Omar, PhD

C. C. 1002493334

DEPARTAMENTO DE ELÉCTRICA, ELECTRÓNICA Y TELECOMUNICACIONES

CARRERA DE INGENIERÍA EN ELECTRÓNICA Y TELECOMUNICACIONES

RESPONSABILIDAD DE AUTORÍA

Yo, **Obando Zapata, Saúl Marcelo**, con cédula de ciudadanía n°1718976291, declaro que el contenido, ideas y criterios del trabajo de titulación**: Diseño e implementación de un sistema de identificación de personas basado en servicios de reconocimiento facial en la nube** es de mi autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Sangolquí,17 de diciembre de 2021

.....

Obando Zapata, Saúl Marcelo

C.C.: 1718976291

DEPARTAMENTO DE ELÉCTRICA, ELECTRÓNICA Y TELECOMUNICACIONES

CARRERA DE INGENIERÍA EN ELECTRÓNICA Y TELECOMUNICACIONES

AUTORIZACIÓN DE PUBLICACIÓN

Yo **Obando Zapata, Saúl Marcelo,** con cédula de ciudadanía n°1718976291, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: **Diseño e implementación de un sistema de identificación de personas basado en servicios de reconocimiento facial en la nube** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

Sangolquí,17 de diciembre del 2021

.....

Obando Zapata, Saúl Marcelo

C.C.: 1718976291

Dedicatoria

Este trabajo lo dedico primero a Dios por ser mi guía y fortaleza en cada instante de mi vida, a mis padres Ruth y Marcelo, a mis hermanos Erick y Alejandro, a mis angelitos Cesar, Elvia a mis abuelitos y a toda mi familia por ser mi soporte a través de su amor, apoyo, preocupación, consejos, enseñanzas, valores y aliento en mi vida que permitieron llegar a cumplir una meta más.

Saúl Marcelo Obando Zapata

Agradecimiento

En primer lugar, quiero agradecer a Dios por todo lo que me ha brindado a lo largo de mi vida, a mi Madre Ruth Zapata Tapia por todo su amor, consejos, su preocupación, su apoyo y por alentarme a seguir adelante y a quien le debo todo lo que soy, a mi Padre Marcelo Obando Changuán por su amor, ayuda y sacrificio para que nunca me falte nada y ser mi ejemplo de perseverancia en la vida.

A mis hermanos Erick y Alejandro por su ayuda, preocupación, por todos los buenos momentos y apoyo incondicional en los malos a lo largo de la vida y saber que puedo contar siempre con ellos.

A mis angelitos Cesar y Elvia por inculcarme su ejemplo, valores, el deseo de seguir adelante y ser una buena persona ante todo y que desde el cielo nos siguen guiando y bendiciendo junto a Dios.

A mis abuelitos Cristina, Hugo y a mi madrina Yolanda por sus consejos, su cariño y siempre estar al pendiente de mí.

A mis familiares, amigos y al futbol gracias por los buenos momentos ya que con ellos y su apoyo de una u otra manera me han permitido culminar este trabajo.

Finalmente agradecer a mi tutor del proyecto Ing. Darwin Alulema por toda su ayuda prestada para solucionar problemas y revisiones para el desarrollo del presente proyecto.

Saúl Marcelo Obando Zapata

Índice

Índice de Tablas

Índice de Figuras

RESUMEN

El Cloud Computing cada vez tiene mayor presencia en las TICs, por varios beneficios como son almacenamiento de datos, bajos costos, no requerir infraestructura física para la administración, simplificar la programabilidad y desarrollar aplicaciones a través de los servicios en la nube como machine learning, base de datos, almacenamiento y transferencia de archivos, procesamiento de datos entre otros. Por lo cual el presente trabajo consiste en utilizar los servicios de reconocimiento facial en la nube para la implementación de un sistema de identificación de personas debido a su utilidad para vigilar, mantener el orden y alertar de posibles agentes desconocidos en campos como la seguridad y la salud entre otros, para lo cual se propone una arquitectura en dos capas, la primera capa es el backend que integra sensores, servicios de almacenamiento y APIs de detección de rostros, identificación y agregación de personas utilizando la nube de Microsoft Azure a través de servicios REST, desarrollado en el framework de Node-RED, con base en JavaScript controlado mediante Rasberry Pi, mientras para el desarrollo de la etapa de frontend se utiliza el dashboard de Node-RED para desplegar la interfaz que pueda ser utilizada en cualquier dispositivo de manera local y a través del internet mediante ngrok y la plataforma de servicio en la nube FRED. Para comprobar el funcionamiento del sistema se evalúa el porcentaje de confiabilidad del reconocimiento de rostros en los periodos del día y noche, el rendimiento y usabilidad.

PALABRAS CLAVE:

- **CLOUD COMPUTING**
- **RECONOCIMIENTO FACIAL**
- **SERVICIOS EN LA NUBE**

ABSTRACT

Cloud Computing has an increasing presence in ICTs, due to several benefits such as data storage, low costs, they do not require physical infrastructure for administration, simplify programmability and develop applications through cloud services such as machine learning, database, file storage and transfer, data processing among others. Therefore, the present work consists of using facial recognition services in the cloud for the implementation of a person identification system due to its usefulness to monitor, maintain order and alert possible unknown agents in fields such as security and safety. health among others, for which a two-layer architecture is proposed, the first layer is the backend that integrates sensors, storage services and APIs for face detection, identification and aggregation of people using the Microsoft Azure cloud through services REST, developed in the Node-RED framework, based on Rasberry Pi controlled JavaScript, while for the development of the frontend stage the Node-RED board is used to display the interface that can be used locally on any device and via the internet through ngrok and the FRED cloud service platform. To check the operation of the system, the percentage of reliability of face recognition in the periods of day and night, performance and usability is evaluated.

KEYWORDS

- **CLOUD COMPUTING**
- **FACIAL RECOGNITION**
- **SERVICES IN THE CLOUD**

CAPÍTULO I

INTRODUCCIÓN

Antecedentes

En la actualidad, Ecuador cuenta con el servicio de videovigilancia de ECU 911, para monitorear las actividades que puedan generar situaciones de riesgo. A escala nacional existen alrededor de 4779 cámaras de videovigilancia instaladas y operativas, entre ellas 430 cuentan con reconocimiento facial y corporal. El sistema de videovigilancia del sistema ECU 911 permite la prevención del delito, reducir el crimen en zonas de vigilancia, además de visualizar las emergencias y eventualidades de riesgo en tiempo real o usar las grabaciones de estas como elementos de evidencia en procesos legales (Servicio integrado de seguridad ECU 911, 2019).

En China por ejemplo se utiliza la identificación facial en el metro de Pekín para separar grupos de personas en función de créditos de seguridad, para las personas de un feedback anormal, realizan la captura de los miles de rostros de pasajeros que serán sometidos a controles suplementarios. El uso del reconocimiento e identificación en China se está expandiendo en todos los centros donde existe gran cantidad de personas como supermercados, medios de transporte, con la finalidad de examinar gestos corporales o situaciones que sean consideradas de riesgo para la ciudadanía (El Comercio, 2019).

El uso de nuevas tecnologías y la aplicación del Internet de las cosas es base para el desarrollo del proyecto para integrar el sistema de reconocimiento de imagen en la nube y ofrecer nuevas alternativas al país para fomentar la transformación digital en entidades públicas y privadas de acuerdo con la ley de conectividad y transformación digital. La cual busca fomentar el uso efectivo de plataformas, tecnologías y servicios digitales e igualmente el uso de datos y redes con el fin de ayudar a simplificar

procesos, adaptar las nuevas tecnologías a las industrias y la de impulsar la economía digital (Rui & Danpeng, 2015).

En este sentido el cloud computing está ayudando al desarrollo y transformación de las TICs en las empresas, debido a sus múltiples servicios para el almacenamiento de datos, procesamiento y desarrollo de aplicaciones que permite perfeccionar proyectos de forma individual o interconectándolos para crear aplicaciones con mayores prestaciones. El cloud computing ofrece múltiples ventajas como son la escalabilidad, de no requerir una infraestructura fisca que administrar, reducción de costos y la accesibilidad a la información desde cualquier momento y lugar del mundo que tenga acceso a internet (Rui & Danpeng, 2015).

Justificación e importancia

La utilización de la identificación facial tiene varias aplicaciones, una de las más destacables es para realizar controles de seguridad en parques, hospitales, bancos, medios de transporte con el fin de contrarrestar delitos y emergencias. En China por ejemplo se utiliza la identificación facial en el metro de Pekín para separar grupos de personas en función de créditos de seguridad, para las personas de un feedback anormal, realizan la captura de los miles de rostros de pasajeros que serán sometidos a controles suplementarios. El uso del reconocimiento e identificación en China se está expandiendo en todos los centros donde existe gran cantidad de personas como supermercados, medios de transporte, con la finalidad de examinar gestos corporales o situaciones que sean consideradas de riesgo para la ciudadanía, por lo que la utilidad del desarrollo del sistema de identificación de personas puede ser aplicado en varios sectores empresariales, debido a la escalabilidad del sistema para ajustarse a los requerimientos según la demanda.

El empleo de las plataformas como servicio ayudan a la propuesta del sistema de identificación ya que disminuye la complejidad de la infraestructura, simplifica el código y permite manejar independientemente cada servicio para después relacionarlos para facilitar la configuración y administración de los elementos del sistema como son los módulos de reconocimiento e identificación de imágenes. Además, el uso de la plataforma como servicio permite mejorar la seguridad y ajustar los recursos computacionales que requiera el sistema.

En este sentido el desarrollo del sistema de identificación facial permitirá brindar soluciones oportunas a la comunidad , brindando facilidades para ser implementado en cualquier parte del mundo y que los servicios implementados en la nube puedan ser adaptados a los requerimientos del sistema, por lo cual el uso de sistemas de identificación en la nube permitirá la transformación y modernización de los entes privados y publicas de seguridad y salud , simplificando tareas y permitirá estar a la vanguardia de las nuevas tecnologías que se han implementado en el mundo.

La importancia del desarrollo del proyecto en el contexto nacional es que se podrá aplicarse en sistemas de seguridad como el sistema de videovigilancia ECU 911, debido a que permitirá reconocer e identificar a los individuos que presenten determinados comportamientos para prevención de riesgos y mejorar la seguridad ciudadana, además de que esta información pueda ser relevante para procesos legales, debido a que su información se puede acceder mediante conexión a internet y ser desplegada en cualquier zona del territorio nacional.

El sistema de identificación podrá ser empleado en zonas de acceso restringidas en sectores de salud, para restringir el paso de personas que no son parte del personal médico a zonas de radiación y contaminación o en el campo militar para la protección de información de investigaciones, datos confidenciales, zonas de riesgo como son

parques de artillería, o sectores minados en el que solo personal militar calificado pueda acceder.

La implementación del proyecto se puede realizar para el ámbito personal, empresarial o estatal ,para ello necesitara el entrenamiento de rostros conocidos para almacenarse en la nube y tener acceso remoto , sin la necesidad de infraestructura física ya que el almacenamiento de datos esta alojada en la nube y la escalabilidad, que permitirá que los recursos computacionales según sea la necesidad de actualizar o modernizar sistemas antiguos se ajuste los recursos computacionales del cliente.

El desarrollo del proyecto permitirá innovar en varios sectores estratégicos que buscan mejorar sus sistemas, adaptarse a nuevas tecnologías y busquen la transformación digital a través del uso de plataformas digitales, el uso de datos, la inteligencia artificial y la identificación de imágenes integrado en el cloud computing, brindando soluciones efectivas en accesibilidad y fiabilidad en la detección de personas.

Alcance del Proyecto

El desarrollo del proyecto consiste en un sistema de identificación de personas basado en servicios de reconocimiento facial en la nube, con la finalidad de brindar seguridad e informar el ingreso de personas desconocidas y de reconocidas que fueron registradas para el entrenamiento de cada grupo familiar a través de un identificativo de acuerdo con un porcentaje de confiabilidad.

El sistema de identificación de personas basado en la nube, comienza con la etapa captura y transmisión de imágenes que permitirá la captura de las imágenes por la cámara , la cual será controlada por la Raspberry Pi para que cada cierto tiempo de acuerdo al movimiento o la proximidad de los sensores, la cámara realice capturas de fotos del rostro y estas a su vez sean enviadas a la plataforma de servicios en la nube que constituye la etapa del Back End , en la cual se realiza el reconocimiento de rostros, que selecciona las características faciales y el módulo identificación que realizara el

proceso de comparar al rostro de la persona con los rostros que fueron entrenados al sistema a través de las credenciales del usuario y su identificativo pueda otorgar un porcentaje de confiabilidad para la identificación de la persona ,esta información será desplegada en un aplicativo web , el cual tiene una interfaz gráfica desde la cual los usuarios accederán al sistema que se encuentra en la nube, y se pueda cargar las imágenes para entrenar el sistema y registrar los datos de identificación de las personas ingresadas al sistema.

Para determinar la funcionabilidad y usabilidad del sistema el ambiente de pruebas se lo realizara en un escenario de hogar prototipo, para lo cual el módulo de captura y transmisión de imágenes se deberán ubicar en la puerta principal de entrada al domicilio para realizar mejores capturas del rostro de la persona en función del ángulo, el tamaño de la foto, iluminación y nitidez para cuando este perciba movimiento se realice la captura de la imagen con el fin de se pueda obtener mejores resultados en el reconocimiento de personas.

Los datos con los que se trabajaran para la funcionalidad del sistema son con el entorno familiar de 4 personas durante los periodos del día y la noche, para lo cual se registrara las fotos de los rostros al sistema de identificación para entrenarlo, para que el sistema pueda distinguir entre los usuarios registrados y no registrados, es decir si se realiza la captura del rostro de una persona que no corresponda al entorno familiar el sistema arrojara una alerta al aplicativo web de que ese rostro no pertenece al identificativo.

Objetivos

General

- Diseñar e implementar un sistema de identificación de personas basado en servicios de reconocimiento facial en la nube.

Específicos

- Investigar los fundamentos teóricos del cloud computing para los servicios de detección de rostros e identificación de imágenes.
- Implementar la etapa de captura y transmisión de imágenes de rostros de personas.
- Implementar los servicios para la detección de rostros e identificación de imágenes en la plataforma de servicios en la nube.
- Implementar backend y frontend del sistema.
- Evaluar los casos de prueba para el funcionamiento y usabilidad del sistema en un entorno experimental.

Estado del Arte

Para la construcción del estado del arte se plantea dos subsecciones. En la primera se define la metodología y los parámetros para la búsqueda del estado del arte y resultados obtenidos mediante un Mapeo Sistemático de Literatura (en ingles Systematic Mapping Study,SMS) y revisión sistemática (Systematic Literature Review, SLR,).Mediante estas metodologías se busca un mayor conocimiento del estado actual de las tecnologías a tratar en el proyecto.

La diferencia entre estas técnicas es que SLR realizar un estudio profundo de los estudios primarios, mientras un SMS se encarga del estudio de un número más amplio de investigaciones, pero menos profundo en detalle (Sinoara,Antunes, & Rezende,2017).

Mapeo Sistemático de la Literatura

En el presente documento se ha justificado la utilización de los servicios de reconocimiento facial en la nube, debido a sus múltiples usos en el mundo para control

de orden y seguridad en diferentes campos de acción, por lo cual permitiría la transformación digital de empresas para la actualización de sistemas de reconocimiento facial utilizando la nube para montar la infraestructura en servidores virtuales, almacenamiento, escalabilidad y el uso de recursos disponibles en la nube.

Por lo cual, se decidió utilizar el SMS para investigación de estudio sobre cuanto se está utilizando los servicios de reconocimiento facial en la nube y las aplicaciones en las que están enfocadas.

Objetivos

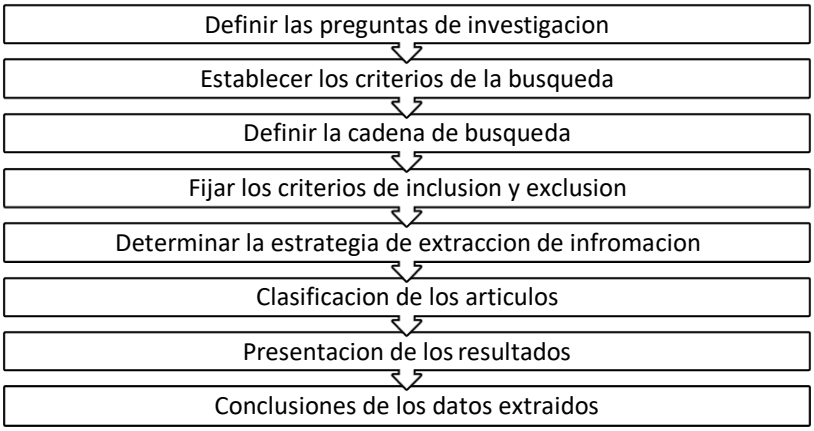
El presente análisis de SMS permitirá conocer las líneas de investigación en el mundo, al igual que las herramientas en las que se utiliza los servicios de reconocimiento facial en la nube y analizar los distintos campos de acciones. Estos resultados del SMS permitirán ser un punto de partida para futuras revisiones sistemáticas.

Método de Investigación

El SMS es una técnica que tiene como fundamento, establecer metodologías de búsqueda de estudios y extracción de informes y la presentación de estudios, a continuación, se muestra los pasos que se deben seguir para el SMS (Bailey y otros,2007), en la siguiente figura se indica el procedimiento seguido por el SMS.

Figura 1

Procedimiento seguido para el SMS.



Nota: Inspirado en (Marquez, J, 2018)

Definir las preguntas de investigación

El SMS permite examinar las técnicas utilizadas en los servicios de reconocimiento facial en la nube, al igual que los campos de investigación que se están implementado a nivel mundial, para este fin se establecen las siguientes preguntas de investigación:

- RQ1: ¿Qué se usó se dio a las plataformas de servicio en la nube para el reconocimiento facial?
- RQ2: ¿En qué campos de acción se ha implementado servicios de reconocimiento facial en la nube?
- RQ3: ¿Que herramientas o dispositivos se usaron para realizar el reconocimiento facial?

Establecer los criterios de búsqueda primarios y secundarios

En el SMS se seleccionó como motor de búsqueda la base de datos de la revista Scopus, ya que tiene gran variedad de cobertura de publicaciones en ciencias sociales,

ingeniería, medicina entre otros. Esta herramienta permite visualizar y analizar los resultados de búsqueda. La revista cuenta con varias publicaciones relacionadas a la electrónica como son (Elseiver,IEEE,ACM,etc).

Definición de la cadena de búsqueda

En el inicio se desconoce la cantidad de información de acuerdo con los criterios de búsqueda, se debe realizar un sondeo de servicios de reconocimiento facial de los artículos disponibles.

Para realizar la cadena de búsqueda del SMS, se han establecido las preguntas de investigación detalladas con anterioridad.

- **RQ1**
  - Cloud service platform OR Plataforma de servicio en la nube.
  - Facial Recognition Services OR Servicios de reconocimiento facial.
- **RQ2**
  - Fields of action OR Campos de acción.
- **RQ3**
  - Tools OR Devices OR Herramientas OR Dispositivos.

En base a la información recabada, la cadena de búsqueda SMS es la siguiente: (Cloud service platform OR Plataforma de servicio en la nube) AND (Facial Recognition Services OR Servicios de reconocimiento facial) AND (Fields of action OR Campos de acción) AND (Tools OR Devices OR Herramientas OR Dispositivos).

Para la cadena de búsqueda propuesta se obtuvieron los resultados siguientes:

- Springer Link: 277 resultados
- IEEE Xplore :105 resultados

Como se puede observar el resultado de publicaciones es de 382, los cuales son sufrientes para realizar el mapeo de garantías con el SMS.

Fijar criterios de inclusión y exclusión

Las publicaciones que se tomaron para estudio de búsqueda son aquellas hasta octubre de 2021 (fecha en la cual se termina el análisis de SMS). Estas publicaciones se seleccionaron de acuerdo con criterios de inclusión y exclusión que a continuación se detallan.

Las publicaciones se deben de adecuar a los criterios de inclusión los cuales se indican a continuación:

- Artículos completos.
- Trabajos referentes a la rama de “Computer Science” o “Engineering”.
- Trabajos de publicaciones de (revistas,conferencias,tesis doctorales entre otros).
- Artículos publicados hasta el 2020.
- Publicaciones en inglés o español

Los trabajos que no estén de acuerdo con los criterios de exclusión son los que se indican a continuación:

- Las publicaciones que estén orientados a servicios de reconocimiento facial en la nube o aquellos que usen este tópico como ejemplo, sin considerarlo como profundización de la publicación.
- Artículos que no se encuentren en las ramas de “Computer Science” “Artificial intelligence”.
- Presentaciones de tipo power-point u otros.
- Trabajos escritos en idiomas inglés o español.

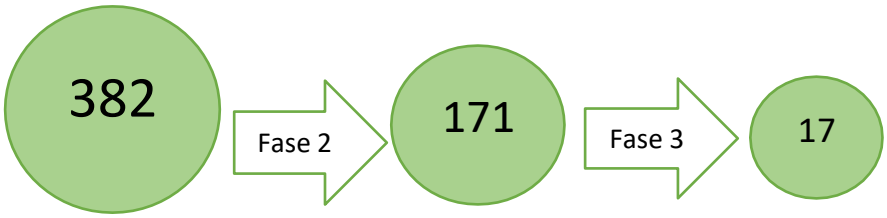
Clasificación de los Artículos

Las especificaciones de los trabajos se los realzo en tres fases:

1. Como primer paso, se especificó los parámetros de búsqueda de los documentos de acuerdo con la cadena de búsqueda. En esta etapa se obtuvo una cantidad de 382 resultados, los cuales se especificó Title+Abstract+Keywords.
2. En la segunda fase se filtró los documentos en base a los especificados en la sección de acuerdos a los criterios de exclusión e inclusión. De acuerdo con el Title+Keywords+Abstract, En la cual se obtuvo 171 resultados.
3. Como último paso se aplicó un nuevo filtro, en el que se consideró los documentos que en realidad aportan para el desarrollo del proyecto, para ello se tomó en cuenta Title+Abstract+KeyWords, se obtuvo 17 articulos. De estos 12 correspondieron a Springerlink y 5 a IEEE XPlore.

Figura 2

Documentos consultados en cada fase del SMS



Resultados

Se obtuvieron 17 artículos que, de acuerdo con los parámetros descritos, a continuación, se indican datos sobre la investigación realizada para SMS.

En el estudio de SMS se puede destacar que el área de reconocimiento facial está implementando nuevas estrategias para disminuir la latencia en el reconocimiento facial y está creciendo en cada uno de los años. En la siguiente figura, se puede

observar que a partir del año 2016 las publicaciones relacionadas a este tópico fue incrementando.

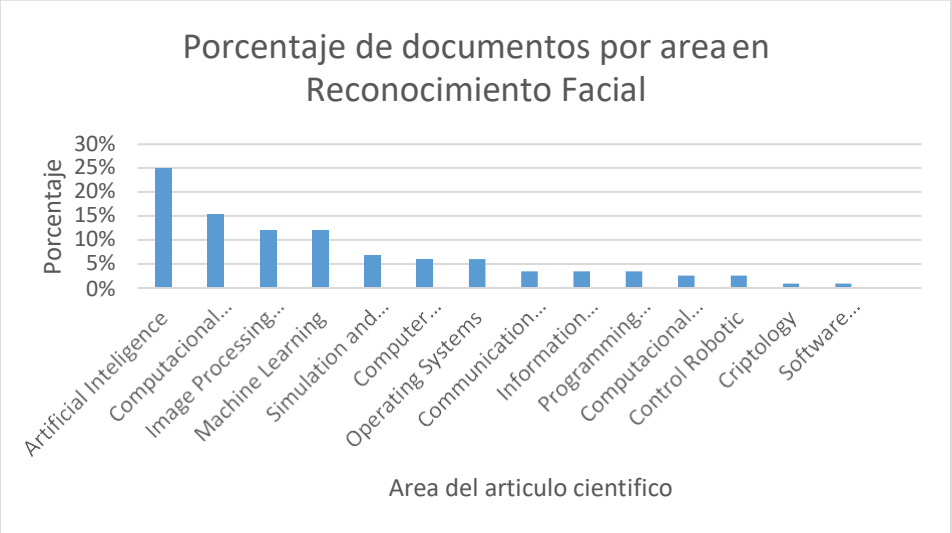
Figura 3

Numero de artículos relacionados al Reconocimiento facial

En la siguiente figura, se puede observar las temáticas que de los artículos extraídos con respecto a la figura anterior, se observa que las áreas de mayor interés son Artificial Inteligence con 22%, Computacional Intelligence con un 15%, mientras que las temáticas con menor cantidad son Control,Robotic,Criptology y Software Enginnering con 1%.

Figura 4

Porcentaje de documentos por área que relacionan el Reconocimiento Facial



Conclusiones del SMS

A continuación, se responde las preguntas formuladas:

Q1: ¿Qué se usó se dio a las plataformas de servicio en la nube para el reconocimiento facial?

Como se observa en la figura anterior , en el apartado de Artificial Inteligence y Computacional Intelligence que conforman el (44% del total), hablan de los métodos de reconocimiento y los algoritmos de detección que se usan, en su mayoría se usa la plataforma de servicios en la nube para el almacenamiento de imágenes, entre los principales trabajos que se enfocaron en plataformas de servicio en la nube para el reconocimiento facial son (Smith & Miller, 2021) , (Roundtree, 2021) , (Sabharwal & Gupta, 2021) en la cual se usa los servicios de reconocimiento facial de Amazon Web Services ,Microsoft Azure, CloudStag, IBM y otros que enfocan sus servicios para implementar servicios de machine learning que se encuentran alojados en la nube como en los trabajos de (Che, Kamphuis, & Kim, 2021), (Nemmaoui & Elhammani, 2021), en los cuales se utiliza a la plataforma para almacenamiento de imágenes y base de datos después de haber realizado el procedimiento de detección de rostros a través de algoritmos de detección de manera local. Estos tipos de servicios cada vez están

presentes en más plataformas de servicios en la nube y ofrecer gran variedad de recursos desde aplicaciones sencillas hasta empresariales a los que se accede a través de una conexión segura a Internet y se paga por el uso de ellos (Yang T, Zhang Y, Sun J, & et al, 2021).

Q2: ¿En qué campos de acción se ha implementado servicios de reconocimiento facial en la nube?

Las aplicaciones y los campos de acción del reconocimiento facial se han implementado para conservación del orden, ayudar en la seguridad, salud y en la educación como se mencionan en (Belhouchette, 2021), el cual se implementó el reconocimiento facial para el aprendizaje profundo, (Bharadwaj, Saini, Chauhan, & Kumar, 2021), aplica el reconocimiento facial para ser usado en la seguridad en hogares y oficinas, (Libby & Ehrenfeld, 2021), se usa para la identificación de rostros de personas en hospitales o mediante el rostro para pacientes que al detectar algún tipo de fatiga alerte al personal médico, (Verma, Sing, & Panigrahi, 2021), se implementó en sistemas de seguridad inteligentes a través de reconocimiento facial.

Los principales campos de acción se dedican principalmente a la vigilancia o el monitoreo de variables del rostro para brindar mejorías principalmente en la seguridad y salud, facilite la vigilancia y permita alertar o diagnosticar diversos problemas, que permitan solventar problemas con mayor rapidez.

Q3: ¿Que herramientas o dispositivos se usaron para realizar el reconocimiento facial?

Los dispositivos que se utilizaron para realizar el procesamiento o la captura de fotografías, es muy variado ya que depende del lugar en el que se va a implementar y cual va ser su finalidad, como se menciona en (Ayad, Taher, & Salem, 2014), en el cual se ocupa una Rasberry Pi para realizar el procesamiento de la información que proviene de la captura de rostros por medio de un dispositivo móvil, (Masruroh, Fiade, & Julia,

2018) ocupa igual una Rasberry Pi para el procesamiento de imágenes además de una tecnología RFID que ayuda a mejorar el registro de estudiantes, en conjunto de una webcam que permite capturar los rostros, (Mehedi , y otros, 2020), propone un sistema de reconocimiento mediante un sistema de cámaras de videovigilancia controladas de un ordenador el cual realiza el reconocimiento en conjunto con un servidor en la nube, (Yi, Jing, Zhu, & Cheng, 2012) realiza el reconocimiento a través de cámaras de videovigilancia conectadas a un DVR , de las cuales se extrae las grabaciones de los videos y realiza un reconocimiento posterior en las horas que los sensores de movimiento fueron activados.

Los dispositivos ocupados para los sistemas de reconocimiento más utilizados son desde placas de computadora como la Rasberry PI u ordenadores para el procesamiento de imágenes con ayuda del cloud computing, mientras que las cámaras ocupadas van desde webcam, cámaras del celular hasta sofisticadas sistemas de videovigilancia, por lo cual el uso de dispositivos dependerá de la aplicación como se indica en la pregunta anterior y el grado de inversión de los dispositivos para el reconocimiento facial de personas.

Otras conclusiones

- Los artículos analizados pertenecen a revistas o conferencias indexadas pertenecientes al tema de estudio.
- De la información que se extrajo del SMS, permitió responder cada una de las preguntas, debido a que los artículos relacionados al tema han tenido mayor cabida en varias revistas debido al interés de mejorar la vigilancia y el monitoreo de personas, mediante el reconocimiento facial.
- Los protocolos más utilizados en este tipo de aplicaciones en la nube son HTTP, MQTT los cuales facilitan la comunicación con los servicios web.

- Un punto de estudio importante en este estudio es el software que se utilizó para la integración de los dispositivos con los servicios web en la nube, por lo que es un tema importante para profundizar.
- Algunos artículos de reconocimiento facial utilizan la nube para montar su propia infraestructura para desarrollar algoritmos de reconocimiento y machine learning y utilizar la nube para almacenamiento de datos.
- La mayoría de los artículos presentan implementaciones de sus proyectos a diferencia de los de simulación o estudios de ensayo, lo que demuestra que los sistemas de reconocimiento fácil están siendo implementados y que esta tecnología de monitoreo está en crecimiento y está presente en varios campos de acción.

Revisión Sistemática de la Literatura

Al culminar el SMS, se pudo recabar la información sobre el avance y el despliegue de los servicios de reconocimiento facial en la nube, sin embargo, se encontró que muy pocos artículos mencionan sobre el software de control para realizar las peticiones a los servicios en la nube y los parámetros de QoS en tiempo real para el reconocimiento facial, lo cual es la motivación para la Revisión Sistemática de Literatura (SLR).

Objetivos

Se necesita realizar un estudio más profundo sobre los softwares o framework utilizados para la gestión y realizar las solicitudes a los servicios web de la nube, con la finalidad de que permite contribuir en el desarrollo del proyecto de investigación.

La información recabada que se va a adquirir mediante el SLR permitirá igualmente conocer si las metodologías de reconocimiento facial tienen mejores

resultados para cumplir los parámetros de calidad de servicios para el reconocimiento en tiempo real.

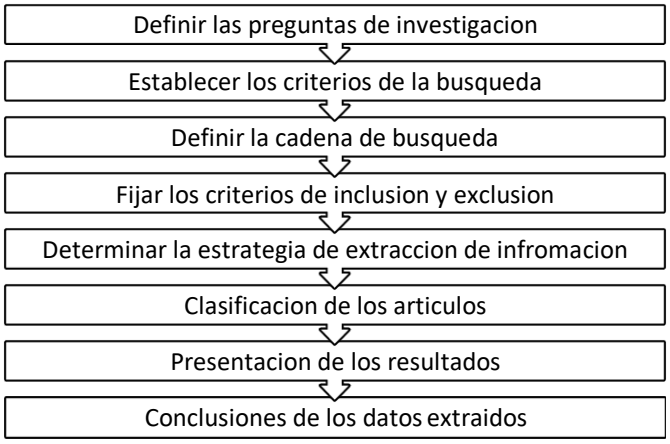
Se opto usar SLR debido a la necesidad de profundizar en estos temas a través de las preguntas necesarias, para conocer los trabajos que se han desarrollados en los años recientes relacionado con el tema.

Método de Investigación

Para el desarrollo del SLR consta de tres etapas: la planeación, ejecución y extracción de resultados. La planeación permite establecer un plan para la elaboración del SLR. La ejecución se realiza la búsqueda y extracción de información de acuerdo con los criterios establecidos para filtrar la información. La extracción permite analizar los resultados y establecer conclusiones.

Figura 5

Proceso seguido para el SLR



Nota: Inspirado en (Marquez, J, 2018)

Definir las preguntas de Investigación

La motivación del SLR es examinar los softwares o frameworks de trabajo con los cuales se realiza la interacción con los servicios web de la nube y la calidad de

servicio de los sistemas de reconocimiento facial, los cuales deben ser analizados a detalle.

- RQ1: ¿Qué software o frameworks fueron utilizados para la interacción con los servicios de reconocimiento facial en la nube?
- RQ2: ¿Qué resultados arrojaron los servicios de reconocimiento facial en calidad de servicio?

Definir cadena de búsqueda

Al igual que SMS, En el inicio se desconoce la cantidad de información de acuerdo con los criterios de búsqueda, se debe realizar un sondeo de servicios de reconocimiento facial de los artículos disponibles.

Para realizar la cadena de búsqueda del SLR, se han establecido las preguntas de investigación detalladas con anterioridad.

- **RQ1**
  - Software OR Frameworks.
  - Facial Recognition Services OR Servicios de reconocimiento facial.
  - Cloud OR Nube
- **RQ2**
  - Results OR Resultados
  - QoS OR Quality of service
  - Facial Recognition OR Reconocimiento Facial

En base a la información recabada, la cadena de búsqueda SLR es la siguiente: (Software OR Frameworks) AND (Facial Recognition Services OR Servicios de reconocimiento facial) AND (Cloud OR Nube) AND (Results OR Resultados) AND (QoS OR Quality of service).

Para la cadena de búsqueda propuesta se obtuvieron los resultados siguientes:

- Springer Link: 182 resultados
- IEEE Xplore :92 resultados

Como se puede observar el resultado de publicaciones es de 274, los cuales son sufrientes para realizar el mapeo de garantías con el SLR.

Fijar los criterios de inclusión y exclusión

Las publicaciones que se tomaron para estudio de búsqueda son aquellas hasta octubre de 2021 (fecha en la cual se termina el análisis de SLR). Estas publicaciones se seleccionaron de acuerdo con criterios de inclusión y exclusión que a continuación se detallan.

Las publicaciones se deben de adecuar a los criterios de inclusión los cuales se indican a continuación:

- Artículos completos.
- Trabajos referentes a la rama de “Computer Science” o “Engineering”.
- Trabajos de publicaciones de (revistas, conferencias, tesis doctorales entre otros).
- Artículos publicados hasta el 2020.
- Publicaciones en inglés o español.

  Los trabajos que no estén de acuerdo con los criterios de exclusión son los que se indican a continuación.

- Las publicaciones que estén orientados a servicios de reconocimiento facial en la nube o aquellos que usen este tópico como ejemplo, sin considerarlo como profundización de la publicación.
- Presentaciones de tipo power-point u otros.
- Trabajos escritos en idiomas inglés o español.

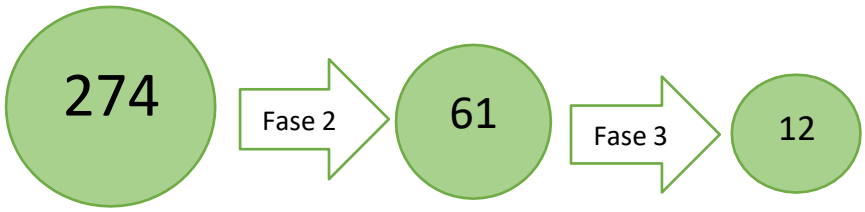
Clasificación de los Artículos

Al igual que el SMS, las especificaciones de los trabajos se los realizo en tres fases:

1. Como primer paso, se especificó los parámetros de búsqueda definidos previamente de los documentos, para ellos se tomó en cuenta Title+Abstract+Keywords.. En esta etapa se obtuvo una cantidad de 274 resultados.
2. En la segunda fase se filtró los documentos de acuerdo con los criterios de inclusión y exclusión. De acuerdo con el Title+Keywords+Abstract, En la cual se obtuvo 61 resultados.
3. Como último paso se aplicó un nuevo filtro, en el que se consideró los documentos que en realidad aportan para el desarrollo del proyecto, para ello se tomó en cuenta Titl+Abstract+KeyWords, se obtuvo 12 artículos. De estos 9 correspondieron a Springerlink y 4 a IEEE XPlore.

Figura 6

Documentos consultados en cada fase del SMS



Resultados

Al culminar la clasificación de artículos se obtuvo 12 artículos de interés, se puede visualizar que el número de artículos obtenidos de acuerdo al SMS es menor, lo que puede ser debido que en enfoque de los artículos científicos en su mayoría se dedica a relacionarse más con los algoritmos de detección y reconocimiento de

personas y que en años anteriores al 2016 no existe gran variedad de análisis de la respuesta en calidad de servicio de estos servicios en reconocimiento facial en la nube en tiempo real, en correlación con los últimos dos años donde si han aumentado esta temática en el análisis de las investigaciones realizadas.

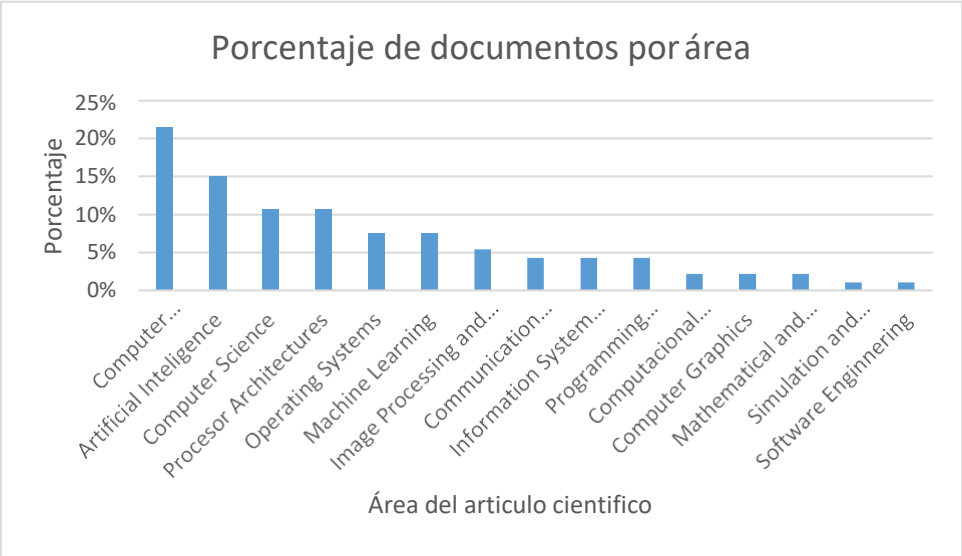
Figura 7

Número de artículos que relacionan Calidad de servicio en Reconocimiento facial por año

Igualmente es necesario para el estudio, en que áreas de aplicación tienen un mayor estudio profundo en la calidad de los servicios del reconocimiento facial en la nube, en la figura siguiente se puede visualizar que las áreas con mayor relación es Computer Communication, Artificial Intelligence y Computer Science con un 50% de trabajos, mientras que las áreas con menor interés son Mathematical, Simulation y Software Enginnering.

Figura 8

Porcentaje de documentos por área que relacionan QoS y Reconocimiento Facial



Conclusiones SLR

A continuación, de acuerdo con los resultados obtenidos y la extracción de información se responden las preguntas de SLR, de cada uno de los documentos y artículos de la fase 3.

RQ1: ¿Qué software o frameworks fueron utilizados para la interacción con los servicios de reconocimiento facial en la nube?

Existen varios framework de IoT que permiten realizar peticiones a los servicios de reconocimiento facial en la nube, entre los que se puede destacar (Lee, Lee, Hyuck, & Kang, 2021) , el cual utiliza iEdge que permite trabajar dentro de un marco de trabajo computacional de borde asistido por IoT que permite la ejecución de aplicaciones en un servidor de borde, (Nazri, Gaafar, & Sajak, 2020), utiliza Node-RED para conectar las cámaras con APIS de reconocimiento de caracteres para establecer vigilancia y monitoreo de placas, al igual que (Sabr, Kanakis, & Belton, 2019),el cual a través de Node-RED realiza el control la captura de fotos de una Rasberry Pi y realiza solicitudes a un servicio de Reconocimiento Facial en la nube en ambientes de prueba Indoor, (Mittal & Singh, 2020), utilizan BLYNK como plataforma de IoT para dispositivos

Android e iOs y THINGSPEAK para establecer una plataforma donde se despliegue en una base de datos de la edad y genero de las personas, en forma estadística, (Subrahmanya, Srinivas, & Rojee, 2017), igualmente existen varios proyectos que prefieren utilizar librerías que faciliten el reconocimiento facial como son Open CV, TensorFlow para mcahine learning para desarrollar el sistema de reconocimiento directamente en lenguajes de programación como Java o Phyton.

RQ2: ¿Qué resultados arrojaron los servicios de reconocimiento facial en calidad de servicio?

Debido a la gran cantidad de plataformas de servicios que realizan reconocimiento en el estudio de (Xue, Zhang, & Ma, 1018), se realiza una simulación de verificación facial, en el cual se selecciona 2000 pares de imágenes faciales del conjunto de datos.1000 pares de ellos contienen las caras con diferentes identidades, otros 1000 pares contienen las caras con la misma identidad. Después de recibir las solicitudes de servicio, la plataforma devuelve la confianza de similitud entre cada par de imágenes faciales. La confianza determina que dos caras pertenezcan a una misma persona. En el cual los resultados fueron que AWS tiene mayor precisión, junto a Tencent y Azure, mientras el tiempo de procesamiento la latencia fue menor para Baidu y Microsoft Azure con 250 ms, además se realiza el mismo procedimiento para OCR y video.

Sin embargo en otros trabajos realizan un enfoque a determinada variable de QoS de acuerdo al sistema de reconocimiento desarrollado en cada estudio, del cual podemos destacar el trabajo de (Srirama, Paniagua, & Flores, 2016), en el cual las pruebas de velocidad de transmisión se obtuvo que para la carga fue de 1.409 kbps mientras la tasa de descarga fue de 3.692 kbps para una velocidad de CPU de 1GHz con 576 MB de RAM, (Yunqui, Liangliang, & Bastani, 2015), en el cual se analiza de acuerdo al tiempo de respuesta a los servicios de reconocimiento y de acuerdo al

tamaño de la imagen, realiza una evaluación del teléfono móvil cuya velocidad de CPU es de 1.6GHz con 1GB de memoria, en el mejor de los casos para una imagen de tamaño de 50X50 permite un ahorro de energía del 66.7% y la latencia de acceso en un 54.3%,lo cual confirma que utilizar los servicios de reconocimiento en la nube pueden ser viables y no realizan un gran consumo de recursos computacionales y de energía en la implementación de estos servicios en dispositivos móviles.

Otras conclusiones

Adicionalmente de respuestas a las preguntas formuladas, también se tienen otras conclusiones de interés para el desarrollo de proyecto de investigación.

- En el estudio SLR, del QoS de los servicios de reconocimiento facial en la nube, confirmo que Microsoft Azure es ideal para el desarrollo del proyecto debido a que posee un alto grado de confiabilidad y latencia muy baja lo que lo hace ideal para trabajar en implementaciones de reconocimiento facial en tiempo real.
- El uso de los frameworks facilita la programación ya que puede integrar dispositivos y APIS, lo cual lo hace ideal para la implementación del proyecto de reconocimiento facial.
- La comparación en parámetros de QoS, entre los servicios de reconocimiento facial en la nube y los algoritmos de machine learning para reconocimiento facial, sería de gran utilidad para establecer el tipo de aplicación y el campo de acción en el que se desempeña mejor cada metodología de reconocimiento facial.
- Las publicaciones de servicios de reconocimiento facial en la nube en los últimos 3 años han ido aumentando, posiblemente debido a la facilidad de programación, usabilidad y las ventajas del cloud computing.

CAPÍTULO II

FUNDAMENTO TEÓRICO Y CONCEPTUAL

Reconocimiento Facial

Es una herramienta que es muy comúnmente usada en el campo de la biometría que a través de la utilización de algoritmos informáticos en un ordenador a través de un procesamiento puede realizar la identificación de personas a través de un hardware que realice la captación de imágenes digitales, para lo cual usa los rasgos faciales de las personas para ser comparadas por los rostros de los individuos (Perez & Agudelo, 2012).

El reconocimiento facial es usando dentro del campo de la biometría para ubicar y determinar las posiciones de las características faciales como son nariz, labios, ojos, cejas entre otros que pueden ser asociados a un individuo. Para tener mejores resultados en el reconocimiento se debe contar con buena iluminación, ángulo y ubicación del dispositivo que realice la captura de la imagen del rostro (Perez & Agudelo, 2012).

Cloud Computing

El cloud computing es una tecnología que abarca áreas tecnológicas como software y hardware que a través de servicios informáticos que se entregan bajo demanda a través de Internet. El cloud computing también es definido como un paradigma escalable y que se puede redimensionar sus recursos físicos o virtuales a través del acceso a la red de Internet con la propiedad de poder gestionar a voluntad y reconfigurar sus servicios (Unión Internacional de Telecomunicaciones, 2014).

La computación en la nube se puede mencionar que se maneja como una arquitectura orientada a servicios (SOA), cuya principal función es el suministro de

recursos informáticos de peticiones a través del Internet, como son el almacenamiento, la computación distribuida, administración e instalación de aplicaciones, acceder a software a través de la web, simplifica la ejecución de aplicaciones, entre otros (Saeed, Baras, & Hajjdiab, 2019).

Cloud computing es un modelo de negocios y tecnológico mediante el cual se puede acceder a un grupo de servicios informáticos dentro de los cuales se puede mencionar el almacenamiento, servidores, aplicaciones y servicios específicos que permiten resolver problemas que en la globalización tecnológica tienen mayor demanda (Duque & Sanchez, 2014).

Ventajas

- Simplifica y facilita el desarrollo e implementación de aplicaciones o sistemas integrados por servicios.
- Costo accesible y en muchos casos depende del tiempo de uso o de los servicios ocupados, además de no necesitar el despliegue y mantenimiento de infraestructura física (Google Cloud, 2021).
- La accesibilidad de los recursos en tiempos cortos, ya que solo es necesario conexión a internet.

Tipos de cloud computing

IaaS (Infraestructura como servicio)

El IaaS brinda una infraestructura computacional para softwares y sistemas Operativos cuyas principales necesidades son el alojamiento web, bases de datos, capacidad de almacenamiento, servidores y entorno de desarrollo de aplicaciones, streaming de video y codificación a través de la red, mientras los clientes gestionan el

control de las operaciones y el despliegue de sus aplicaciones, sistemas operativos y configuraciones que se realicen (Nuñez, 2013).

PaaS (Plataforma como servicio)

Este modelo permite que todas las funciones que estén disponibles en la web para que el cliente pueda desplegar y comercializar aplicaciones generadas por el usuario en la infraestructura del proveedor, el que brinda la plataforma de desarrollo y los servicios para programar, lo que permite que el cliente administre el control de la aplicación, pero no toda infraestructura que permite la implementación de la aplicación (Licencias Online, 2013).

SaaS (Software como servicio)

El servicio SaaS, es un modelo de implementación de software, la cual una aplicación computacional se brinda como un servicio mediante conexión a Internet, que permite presidir el requerimiento de instalación y ejecución de aplicaciones en los ordenadores de los usuarios finales (Relica, 2014).

El software como servicio permite que una infraestructura en la nube se realicen las aplicaciones del proveedor, las cuales se podrá acceder a través de una interfaz web que estará en interacción con el cliente. Las aplicaciones que se realizan comúnmente en el modelo SaaS se orientan en proveer funcionalidades a bajos costos, con la misma garantía sin la complejidad del soporte administración y soporte, de esta manera el cliente SaaS tiene la responsabilidad de gestionar el software contratado, mientras el proveedor es el encargado de mantener la operación del sistema en la nube en la cual se encuentra el servicio (Nuñez, 2013).

Ventajas de SaaS (Software como servicio)

- Elimina la carga del mantenimiento y el soporte técnico para mantener operativo al sistema.
- Abarata costos en inversiones iniciales en la adquisición de licencias o sistemas informáticos.
- Permite la escalabilidad y modernización de aplicaciones ya que se encuentra en constante evolución mediante la solicitud de nuevos servicios que permiten la modernización de la aplicación o sistema del cliente (Duque & Sanchez, 2014).
- Manipulación de aumento o disminución de almacenamiento.

Internet of Things

Es una Infraestructura a nivel global que permite la interoperabilidad de las tecnologías de la información y comunicación (TICs) y la comunicación entre de dispositivos virtuales o físicos con el fin de obtener los servicios que permitan el, procesamiento, identificación y adquisición de datos, con garantías de seguridad y privacidad hacia el usuario (ITU Corporation, 2015).

Es la interconexión de elementos u objetos de uso cotidiano como sensores, actuadores, dispositivos, entre otros los cuales son localizables y reconocibles a través de esquemas de direccionamiento único con la cualidad de compartir información con otros dispositivos mediante internet y de esta manera brindar u obtener un servicio (Serbanati, Medaglia, & Ceipidor, 2011).

Microservicios

Los microservicios son herramientas tecnológicas que permiten administrar las aplicaciones de código a través de una metodología practica y se pueden ejecutar por pequeñas bases de código y despliegues independientes, las ventajas de usar

microservicios es la capacidad de publicar una aplicación que se pueda desplegar, desarrollar, escalar, visualizar y manejar de manera independiente (Fowler & Lewis, 2014).

Los microservicios brindan los beneficios de desarrollar una herramienta o aplicación y dividirla en partes más pequeñas, lo que permite que se pueda desarrollar cada aplicación independiente lo que produce que los costos sean bajos, rapidez, escalabilidad y automatización (Villamizar, et al., 2015).

Una de las ventajas de utilizar microservicios es la capacidad de publicar una aplicación grande como un conjunto de pequeñas aplicaciones (microservicios) que se pueden desarrollar, desplegar, escalar, manejar y visualizar de forma independiente. Los microservicios permiten a las empresas gestionar las aplicaciones de código base grande usando una metodología más práctica donde las mejoras incrementales son ejecutadas por pequeños equipos en bases de código y despliegues independientes. La agilidad, reducción de costes y la escalabilidad granular, traen algunos retos de los sistemas distribuidos y las prácticas de gestión de los equipos de desarrollo que deben ser considerados. (Villamizar et al., 2015).

SOAP

Los servicios que están basados en SOAP tiene su funcionamiento a partir de tecnologías como Universal Description, Discovery an Integration (UDDI) y el Web Service Description Language (WSDL).El protocolo SOAP tiene su origen en recomendaciones de la W3C en el que describe que el protocolo transfiere la información entre los nodos (solicitante y receptor) a través de XML, debido a que se requería un protocolo que sea independiente del sistema operativo, de la plataforma y del transporte, ventajas que permite utilizarlo en ambientes heterogéneos requeridos por el servicio Web (W3C, 2007).

REST

La arquitectura REST se presenta como un conjunto de recursos y condiciones basada en la red, la cual se representa de acuerdo con un (uniform resources identifier) URI, que permite los recursos se puedan mostrar como objetos de datos en formatos JSON o XML, por lo cual REST construye servicios web escalables y flexibles a partir del protocolo HTTP (Fielding, 2000). Debido a que los servicios web son sistemas informáticos que son diseñados que permiten la comunicación y transferencia de información en una red entre ordenadores ejecutados en un servidor donde se encuentra alojado (Fielding, 2000).

Los beneficios de utilizar la arquitectura REST por delante de otras más robustas como la arquitectura SOAP es que esta no requiere un gran ancho de banda (Navarro, 2007). Estas características permiten que la programabilidad y la manipulación de los datos sea más convenientes para el desarrollo del proyecto. Los servicios web que están basados en la arquitectura REST tratan de emular al protocolo de transferencia de hipertexto HTTP sin la necesidad de establecer comunicación directa con la interfaz, por lo cual utilizan una serie de métodos para manipular los recursos de los servicios (Navarro, 2007).

URIs

Se refiere al recurso de identificación uniforme, cuyo propósito es ubicar y utilizar el recurso para intercambiar información de páginas, secciones o archivos para interpretar los datos ya que de esta manera se puede borrar, acceder o modificar en el formato que requiera el cliente (Santos & Serrano, 2017).

Métodos HTTP

Los métodos HTTP definen que acción se a realizar sobre los recursos de acuerdo con la siguiente Tabla.

Tabla 1

Descripción de los métodos de HTTP

Método HTTP	Descripción
GET	Es usado para solicitar la representación del recurso sin cambiar su estado.
POST	Genera nuevos recursos que son subordinados del recurso solicitado direccionados por la URI, este tipo de método también permite modificar los recursos que existen.
HEAD	Realiza la petición del encabezado del recurso al servidor, para verificar el estado de los archivos.
PUT	Se utiliza para modificar los recursos, para lo cual los datos actualizarse se deben de modificar en el cuerpo del mensaje
PATCH	Modifica una sección del recurso.
DELETE	Elimina el recurso.

Nota: Esta tabla muestra la descripción de los métodos que son utilizados para el desarrollo del sistema de reconocimiento, inspirado en (Lopez, 2018).

Microsoft Azure

Es una plataforma de Microsoft que está compuesta por un gran conjunto de servicios y productos en la nube para solucionar problemas y dificultades que permiten el desarrollo y modernización de las tecnologías a través de la conexión a internet (Microsoft Azure, 2021).

Es un conjunto de servicios informáticos en la nube que ayuda a afrontar los desafíos tecnológicos empresariales que brinda servicios de plataforma (PaaS) e infraestructura (IaaS) de tal forma que el cliente pueda subir los servicios de forma sencilla y utilizar servicios como almacenamiento, bases de datos, máquinas virtuales, infraestructura virtual, ambientes de desarrollo, análisis de datos y procesamiento en la cual el cliente solo se dedica a gestionar el software contratado, sin la necesidad de preocuparse en los recursos físicos por el mantenimiento soporte o actualización de los servidores (Ruiz, 2019).

Microsoft Azure es una plataforma que fue comercializada a partir del 2010 y continuamente mejorada a través de nuevos servicios en concordancia a las nuevas tecnologías. Azure permite que en su plataforma se alojen aplicaciones existentes además de escalar para que sean mejorados mediante la utilización de nuevos servicios o la utilización de mayores recursos (Ruiz, 2019).

La nube de Microsoft Azure está compuesta por más de 200 productos y servicios con el objetivo de dar nuevas soluciones a las dificultades. Se puede crear, ejecutar y administrar aplicaciones en distintas nubes, en el entorno local y en el perímetro, con las herramientas y marcos a su preferencia (Microsoft Azure, 2021).

Azure es compatible con una gran variedad de los sistemas operativos, lenguajes, herramientas y marcos más importantes de la industria, desde Red Hat hasta Ubuntu, Windows y openSUSE, MariaDB y SQL Server, C# a Java. Pone al alcance de su mano los mejores ecosistemas para que pueda crear excelentes aplicaciones y servicios que funcionen en muchos dispositivos (Microsoft Azure, 2021).

Características

- El hardware de Azure es un enorme conjunto de equipos de servidores virtualizados agrupados en cientos de centros de datos masivos de todo el mundo.
- El sistema operativo de Azure está conformado por servicios que asignan y niegan las asignaciones dinámicamente de distintas partes de ese grupo de recursos a medida que las aplicaciones los necesitan. Las asignaciones dinámicas permiten a las aplicaciones responder velozmente a cualquier número de condiciones cambiantes, como la demanda de los clientes.
- Las asignaciones son denominadas recursos y a cada recurso se le asigna un identificador de objeto y una dirección URL. Entre algunos tipos de recursos se incluyen máquinas virtuales, base de datos, almacenamiento, registros de contenedor, redes virtuales, motores de inteligencia artificial y etc.
- La nube de Microsoft es compatible con una gran variedad de los sistemas operativos, lenguajes, herramientas y marcos más importantes de la industria, desde Red Hat hasta Ubuntu, Windows y openSUSE, MariaDB y SQL Server, C# a Java. Pone al alcance de su mano los mejores ecosistemas para que pueda crear excelentes aplicaciones y servicios que funcionen en muchos dispositivos. (Microsoft Azure, 2021).

Microsoft Azure Blob Storage

Azure Blob Storage ayuda a crear lagos de datos para la necesidad del cliente de análisis y provee de almacenamiento por niveles para los datos a largo plazo para la creación de potentes aplicaciones móviles y nativas de la nube (Microsoft Azure, 2021).

Azure Blob Storage es de tipo no estructurado, que quiere decir que no existe inconveniente en el tipo de dato que contiene como puede ser imágenes, videos, documentos, archivos tipo JSON entre otros, para los cuales cada uno de ellos tiene una dirección URL archivo en el cual consta la cuenta de almacenamiento, el contenedor y el nombre del archivo desde cualquier dispositivo que cuente con conexión a Internet (Microsoft Azure , 2021).

Ventajas

Microsoft Azure Blob Store ofrece una copia de seguridad, restauración y recuperación de datos por algún desastre en la entrega de imágenes o documentos en el mismo navegador, almacenamiento de los datos para su respectivo análisis por parte de un servicio de Azure, transmisión de video y audio, escribir en archivos de registro y almacenamiento de archivos para acceso distribuido.

Los usuarios tienen la capacidad de acceder a los objetos en Blob Store a través de HTTP/HTTPS, desde donde sea. Se puede accederá a los objetos de Bob Store desde Api de REST de Azure Storage, la CLI de Azure Power Shell o una biblioteca cliente de Azure Storage. Las bibliotecas incluyen .NETO, Java, Node.js Pitón, Ir, PHP, Rubí (Microsoft Azure, 2021)**.**

Cognitive Services

Es un conjunto de servicios vinculados a servicios de inteligencia artificial y API cognitivas como son la visión, discurso, idioma, decisión entre otras que permiten crear y desarrollar aplicaciones inteligentes, además de disponer de datos para el entrenamiento de modelos o para que el usuario pueda incorporar sus propios datos (Rodriguez H. , 2018).

Cognitive Services es un conjunto de servicios, APIS y SDK que permite a sus usuarios crear apps de confianza usan la inteligencia artificial para dar soluciones innovadoras para el crecimiento tecnológico en cuanto lenguaje artificial, voz y visión (Rodriguez H. , 2018).

Face

Es una Interfaz de programación de aplicaciones que permite realizar el estudio en rostros de imágenes para su detección, entre los cuales se puede obtener características como sexo, edad, utilización de anteojos. Entre las principales usos y aplicaciones están la Identificación de caras similares, Identificadores de rostros de personas a través del entrenamiento con base de datos. Para utilización de la Interfaz de programación se debe tener en cuenta Visual Studio y estar en suscripción al entorno de Azure. La programación se puede llevar a cabo en una aplicación de consola (.NET Framework), Phyton, Nodo js, Ruby y Java (Rodriguez H. , 2018).

Es una API es una de las múltiples herramientas del servicio de Azure Cognitive services, el cual permite capturar y ubicar las características del rostro con alta precisión cuyas funciones más destacadas con el reconocimiento facial de rostros y detección de características físicas del rostro como son adopción de posiciones, edad, sexo, uso de anteojos, vello facial, emociones. Igualmente, el sistema puede ser entrenado por un grupo de personas para que se realice un proceso de verificación al compararlos con otros rostros (Caceres, 2018).

Algoritmos de detección facial

Microsoft Azure no muestra información de los algoritmos de detección facial ya que sus servicios no son de código abierto, sin embargo, el proceso que realiza para la detección facial y reconocimiento se utiliza los llamados “key points”, los cuales son un

conjunto de puntos informativos que se instalan en diferentes coordenadas del rostro para comunicar diferentes datos referenciales de interés para el cliente. El servicio de Face Api por defecto usa 27 tokens para ubicar las posiciones de puntos específicos del rostro como son las posiciones y volúmenes de las cejas, ojos, nariz, labios mentón entre otros (Grabovskyi & Martynovych , 2019).

El procedimiento que realiza la Api Face de Azure es el reconocimiento facial, a través del cual se puede realizar los procesos de identificación o verificación del individuo a través del rostro del individuo. El proceso de identificación permite comparar al sujeto de entrada al reconocimiento facial con otro grupo de personas asociadas a un grupo, al realizar este proceso la salida será las coincidencias de este con los otros candidatos, en cambio el proceso de verificación compara a dos sujetos que ingresas al reconocimiento facial, en el cual su salida mostrará las similitudes entre los rostros. El reconocimiento facial es importante, debido a que permite verificar que un usuario pueda comprobar quien dice ser, a partir de su identidad, en cambio la identificación permite realizar comparaciones de coincidencia de aspectos del rostro entre un grupo de rostros (Microsoft Azure, 2019).

Node Red

Node-Red es un software de código abierto, desarrollado por IBM Company usado principalmente en Internet of Things, mediante la interconexión de nodos utilizando protocolos estandarizados como MQTT o los métodos de HTTP Request para acceder a recursos de hardware, APIs permitiendo la comunicación e interacción de estos a través de una interfaz web (Node-RED, 2020).

Node-RED es una herramienta visual de programación basado en Node.js, para aprovechar el modelo sin bloqueo impulsado por eventos, lo cual permite que pueda ejecutarse en la nube como en ordenadores de bajo costo como Raspberry Pi.

Ventajas

Node-Red entrega un editor de flujo basado en un navegador que hace más fácil la conexión de flujos haciendo uso de la amplia gama de nodos de la paleta. Después con tan solo un clic los flujos se pueden implementar en el tiempo de ejecución. Una biblioteca incorporada permite almacenar las funciones, plantillas o flujos útiles para su reutilización.

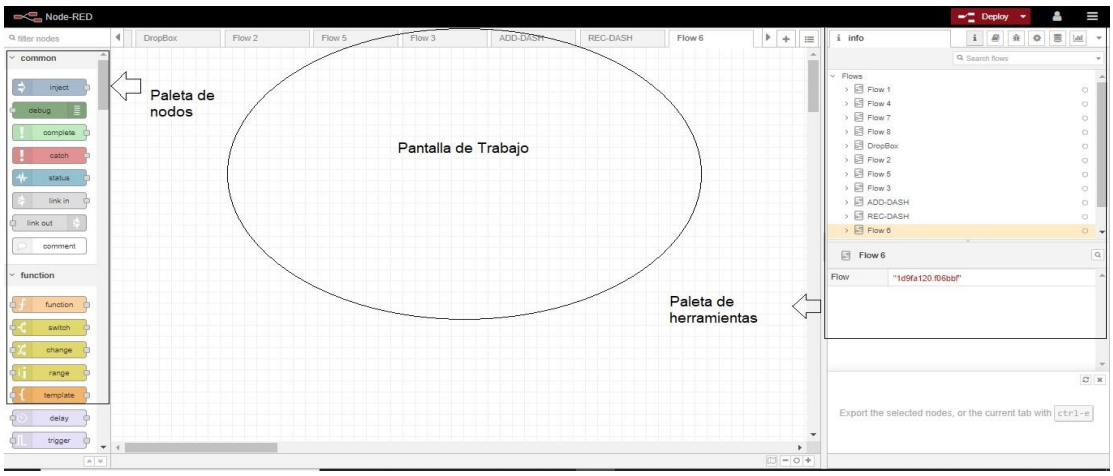
En Node, js el tiempo de ejecución es liviano, utilizando al máximo su modelo sin ningún bloqueo impulsado por eventos. Esto hace que sea ideal ejecutarse en el borde de la red en hardware de bajo costo como Raspberry Pi y en la nube. Teniendo más de **225.00** módulos en el repositorio de paquetes de Node, hace más fácil aumentando la gama de nodos de paleta para añadir nuevas capacidades (Node-RED, 2021).

Entorno grafico de Node-RED

El entorno de trabajo de Node-RED consta de tres partes la paleta nodos, el entorno de trabajo y el menú de herramientas como se indica en la siguiente figura.

Figura 9

Entorno de trabajo de Node-RED



Nota: El grafico representa el entorno de Node-RED y sus ventanas de trabajo, obtenido de (Node-RED, 2021)

En la siguiente Tabla se detalla la función de cada ventana del entorno de trabajo del software de Node-RED.

Tabla 2

Descripción de los tipos de ventana de Node-RED

Tipo de Ventana de Node-RED	Descripción
Paleta de Nodos	En esta ventana se encuentran todos los nodos de trabajo, para su utilización se debe arrastrar hacia la pantalla de trabajo, además existe la posibilidad de instalar nuevos nodos.
Pantalla de Trabajo	En esta ventana se crea la lógica de programación grafica mediante la interconexión de nodos, se puede crear más ventanas llamadas flujos que pueden compartir información entre ellas.
Ventana de herramientas	En esta ventana se puede observar la información de las ventanas, información de los nodos utilizados, depuración del programa y la ejecución del programa de forma visual mediante el dashboard de Node-RED.

Nota: Esta tabla muestra los tipos de ventanas en el entorno de trabajo de Node-RED.

Red Privada Virtual (VPN)

Es una red privada que permita la comunicación de datos de forma segura entre diferentes lugares del mundo a través de Internet, ya que el Internet al ser abierta y publica mediante túneles virtuales, los cuales brindan confiabilidad y seguridad para la transmisión de información (REQUEST FOR COMMENTS (RFC2764), 2018).

Una red VPN asegura que la transmisión de datos sea segura, gracias a la encriptación y encapsulación de datos mediante protocolos de seguridad robustos entre dos redes ubicados en sitios remotos por medio del Internet. En una VPN los enlaces de red son lógicos e independiente de la infraestructura de la topología física utilizada (REQUEST FOR COMMENTS (RFC3193), 2018).

Ventajas de una red VPN

- Asegura la confidencialidad de los datos que viajan por medio el Internet a través de métodos de cifrada de información.
- Garantiza la seguridad entre el emisor y receptor para que puedan ser identificados.
- Certifica que la información enviada no fue modificada ni alterada y que sea exactamente los datos que el emisor transmitió.
- Garantiza que el emisor del mensaje no pueda denegar la emisión del mensaje (REQUEST FOR COMMENTS (RFC3193), 2018).

Ngrok

Es una herramienta que ayuda en el acceso a la aplicación del servidor local a cualquier cliente con internet a través de una url generada de forma dinámica sin la necesidad de realizar modificación en apertura de puertos en el router o la seguridad del firewall en la computadora, esta herramienta de ejecutable puede ser descargada en

cualquiera plataforma como Windows,macOS o Linux, la cual es genera un túnel entre los servidores y la plataforma de servicio web local, permitiendo el acceso desde cualquier parte del mundo (ngrok, 2021).

FRED

Es una plataforma como servicio en la nube versátil que permite controlar y gestionar instancias de Node-RED para usuarios en la Nube a través de Docker, aplicado para optimizar aplicaciones de IoT, cada instancia se basa en un proxy inteligente que transmite comunicaciones HTTP y sockets web a las instancias de Node-RED (FRED-Sensetenic, 2021).

Rasberry Pi 3

Es un pequeño computador de bajo costo, su porte es el de tamaño de una tarjeta de crédito, puede ser conectada a través de un monitor de computadora o tv. Este pequeño computador corre el sistema operativo Linux que permite a las personas de todas las edades explorar y aprender sobre la computación. También este pequeño computador que puede hacer la mayoría de las tareas típicas como la navegación en internet, reproducir videos en alta resolución y manipular documentos. Algunas áreas en las cuales se puede realizar proyectos con las raspberry pi 3 son: La robótica, la domótica, redes de computadoras, seguridad, programación y entre otras áreas (Gomez Rodriguez, 2007).

Para el desarrollo del sistema de reconocimiento facial se utilizó la Rasberry Pi 3 Modelo B+, las características se detallan a continuación.

Características de la Rasberry Pi 3 Modelo B+:

- CPU + GPU: Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz
- RAM: 1GB LPDDR2 SDRAM
- Wi-Fi + Bluetooth: 2.4GHz y 5GHz IEEE 802.11.b/g/n/ac, Bluetooth 4.2, BLE
- Ethernet: Gigabit Ethernet sobre USB 2.0 (300 Mbps)
- GPIO de 40 pines
- HDMI
- 4 puertos USB 2.0
- Puerto CSI para conectar una cámara.
- Puerto DSI para conectar una pantalla táctil
- Salida de audio estéreo y vídeo compuesto
- Micro-SD
- Power-over-Ethernet (PoE)

Figura 10

Rasberry Pi Modelo 3B+

Nota. El grafico representa las partes de la Rasberry Pi, Tomado de (Hetpro, 2019).

Raspbian

Es un sistema operativo de código abierto basado en Debian, este sistema es optimizado para ser implementado en la plataforma de ARM a través de un entorno agradable y fácil de usar, el sistema operativo trabaja con la interfaz de escritorio de

LXDE, el cual incluye herramientas como:IDLE para Python.Scratch,Wolfram Mathematica, Libre Office y una gran cantidad de herramientas que permiten manipular las interfaces físicas de la placa CPU de las Rasberry PI (Rasberry Pi, 2021).

Cámara

Es un dispositivo tecnológico que permite capturar imágenes estáticas en la zona en la que haya sido orientada (Rodriguez J. , 2013). La cámara Rasberry Pi como el nombre lo dice es únicamente para las minicomputadoras Rasberry PI, básicamente tiene las mismas funciones que de una cámara como captar imágenes en movimientos y estáticas. Además, la cámara en el Rasberry PI 3 puede ser muy útil para proyectos como reconocimiento de personas a través de una base de datos almacenada, también es compatible con el sistema operativo raspbian, que permite fotografiar y grabar video de larga duración (Suntaxi Cantuña, 2019).

Para el desarrollo del proyecto se utilizó la cámara Modelo Ov5647 1080p Ir, las características se detallan a continuación.

Características

- Sensor de imagen de 5 megapíxeles
- Capta imágenes estáticas de 3280 x 2464
- soporta vídeo 1080p30, 720p60 y 640x480p90.
- Resolución de la captura de video: 1080p
- Interfaz mediante conector CSI.

Figura 11

Cámara utilizada para la Rasberry Pi

Nota: El grafico representa la cámara modelo Ov5647 para la Rasberry PI utilizada en el sistema de reconocimiento. Tomada de (Avelectronics, s.f.).

Sensor ultrasonido

El sensor ultrasonido es un sistema para la medición de distancias y velocidades través de ultrasonidos, además puede detectar de defectos en piezas metálicas hasta apertura autentica de puertas, también los sensores son muy útiles para la realización de diversos proyectos como en robótica, domótica electrónica, seguridad y etc.

El funcionamiento del sensor es a través de un puso ultrasónico en cual se transmite por el aire hasta que es finalmente reflejado por una superficie reflectora (Gonzalez Anton, 2015).

Características

- Tiene un voltaje operativo de 5 VDC
- Puede viajar a través del aire
- Genera una frecuencia de 40kHz para captar o generar ultrasonidos.

Figura 12

Sensor de ultrasonido

Nota: El grafico representa el sensor de ultrasonido modelo HC-SR04. Tomado de (Diosdado, 2018).

Sensor PIR

La función del sensor pir es detectar el movimiento de las personas con el fin de reconocerlas, la detección se lo realiza a través de los cambios electromagnéticos que suceden alrededor del sensor pir, además que particularmente se trabaja con luz infrarroja ya que los seres vivos como los animales y los seres humanos emiten luz infrarroja, mientras que los demás objetos emiten un rango de energía térmica que es menor a la luz infrarroja, Además cuando el sensor detecta una presencia realizara un conteo para capturar la imagen y enviar al sistema la información de la captura de imagen, también este tipo de sensor es de mucha ayuda en el ámbito de seguridad ya que siempre se mantendrá alerta sobre que personas entran en contacto con el sensor y enviara la información de inmediata al sistema (Gonzalez Anton, 2015).

Características

- Tiempo de retardo de la señal de salida ajustable y un ángulo de 100°.
- Trabaja normalmente con el rango de luz infrarroja.
- Trabaja con el lente Fresnel que le ayuda a captar a luz infrarroja a su alrededor.

Figura 13

Sensor de movimiento PIR

Nota. El grafico representa el sensor PIR. Tomado de (Iberobotics, 2021)

Capítulo III

ANALISIS Y DISEÑO

Análisis

Los requerimientos del cliente y el desarrollador para el diseño del sistema de reconocimiento, permitirá estructurar el diseño de la arquitectura, igualmente de llevar a cabo los objetivos que fueron planteados el Capítulo I, para lo cual se debe establecer los requisitos funcionales y no funcionales que se indican en las siguientes Tablas respectivamente.

Tabla 3

Requisitos de diseño funcionales

Requisito	Descripción
Requerimientos del cliente	• La interfaz de usuario esta direccionada a dispositivos móviles como el ordenador, con información fácil para manejar. • El cliente debe contener el software de programación como Node-Red. • El cliente debe poseer conocimiento básico en electrónica, como sensores, controladores y programación.
Requerimientos que debe cumplir la arquitectura del software	• Debido a la necesidad de reconocimiento facial la transferencia de datos debe ser en tiempo real. • La base de datos de las imágenes debe ser incremental debido a la gran cantidad de datos que existan. • El cliente debe contar con los claves y usuarios de la base de datos como la API de reconocimiento facial.
Funcionalidades que debe cumplir la arquitectura de software	• La arquitectura de software debe tener la capacidad de establecer comunicación con la aplicación de control del Front-End.

Detalles técnicos y manejo de datos	• Es indispensable tener conexión a internet ya que las peticiones para el reconocimiento e identificación, además de la base de datos de Azure. • La arquitectura del sistema sigue un diseño de tipo REST, en el back-end como en el fornt-end. • Los datos para ser manipulados para peticiones como respuestas son de tipo JSON. • Las peticiones para él enviaron de datos se lo realiza a través del protocolo de comunicación HTTP.

Tabla 4

Requisitos de diseño no funcionales.

Requisito	Descripción
Cualidades de la aplicación de control	• Debido al reconocimiento facial, se debe intentar que sea lo más rápido para una mejor QoS. • La aplicación debe informar cuando se ingrese de forma incorrecta el identificativo o en caso de no ingresar al sistema a través del usuario y contraseña. • La aplicación cuenta con una sola ventana, la cual cuenta con formularios y botones dinámicos que permite realizar el reconocimiento y la agregación de personas a los identificativos, además de regresar al menú principal.

Cualidades de la arquitectura de software puede tener	• La base de datos esta albergada en la nube la cual esta almacenada de acuerda a la hora que se realizó la captura de la foto. • Los microservicios funcionan de forma independiente, pero el servicio de reconocimiento depende de la base de datos para extraer la imagen.
Cualidades que puede tener el sistema de reconocimiento	• El sistema de reconocimiento puede ser fijo o móvil. • La fuente de alimentación del sistema de reconocimiento se lo realiza directamente desde la red eléctrica. • La conexión a internet puede ser mediante LAN, WLAN.

Restricciones generales	• Sin conexión a internet el sistema de reconocimiento queda obsoleto. • La latencia del sistema de reconocimiento dependerá de la calidad y nitidez de la imagen. • Para acceder a la página web, se debe contar con un navegador en el dispositivo que se va a utilizar.

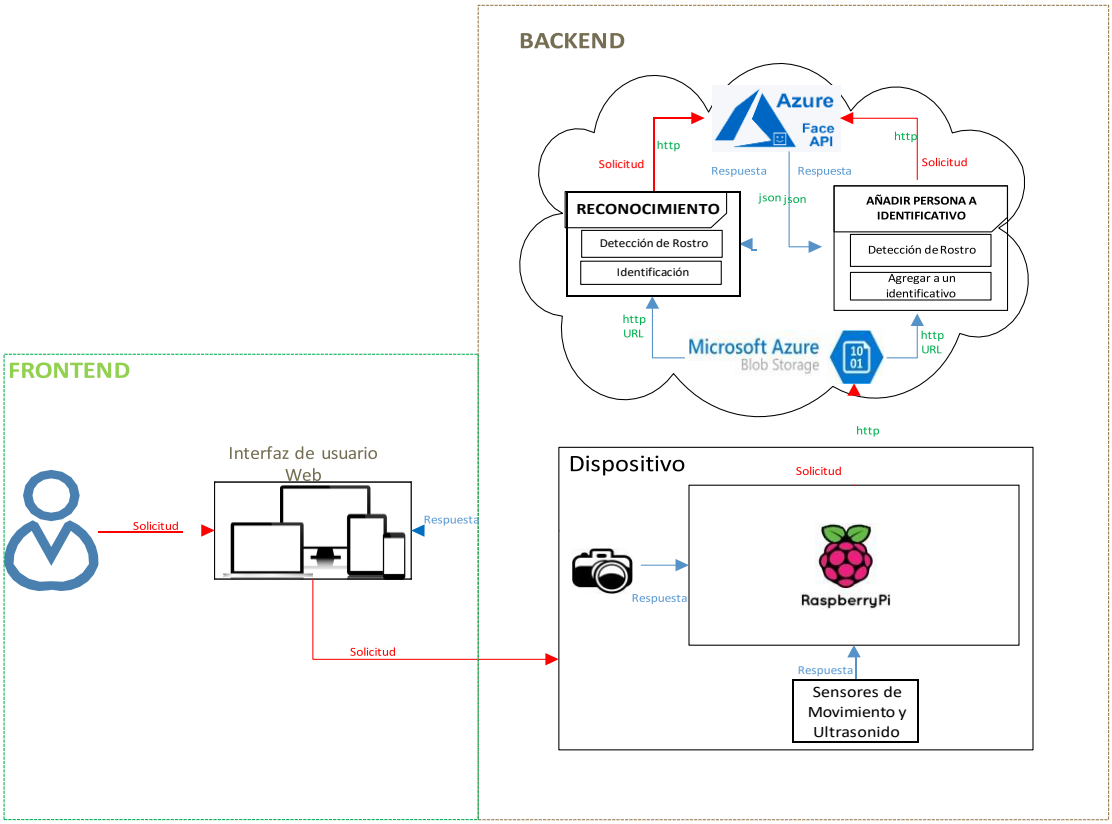
Arquitectura

El sistema de identificación de personas basado en la nube, comienza con la etapa captura y transmisión de imágenes que permitirá la captura de las imágenes por la cámara , la cual será controlada por la Raspberry Pi para que cada cierto tiempo de acuerdo al movimiento o la proximidad de los sensores, la cámara realice capturas de fotos del rostro y estas a su vez sean enviadas a la plataforma de servicios en la nube que constituye la etapa del Back End , en la cual se realiza el reconocimiento de rostros, que selecciona las características faciales y el módulo identificación que realizara el proceso de comparar al rostro de la persona con los rostros que fueron entrenados al sistema a través de las credenciales del usuario y su identificativo pueda otorgar un porcentaje de confiabilidad para la identificación de la persona ,esta información será desplegada en un aplicativo web , el cual tiene una interfaz gráfica desde la cual los usuarios accederán al sistema que se encuentra en la nube, y se pueda cargar las imágenes para entrenar el sistema y registrar los datos de identificación de las personas ingresadas al sistema.

El diseño del diagrama de bloques del sistema de identificación se indica en la figura siguiente.

Figura 14

Diagrama de bloques del sistema de identificación de personas basado en servicios de reconocimiento facial en la nube.

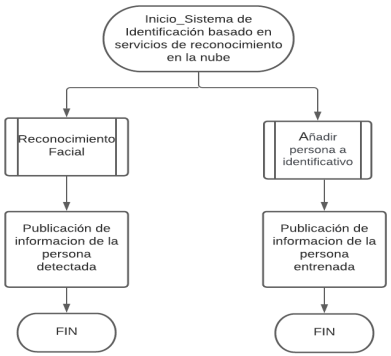


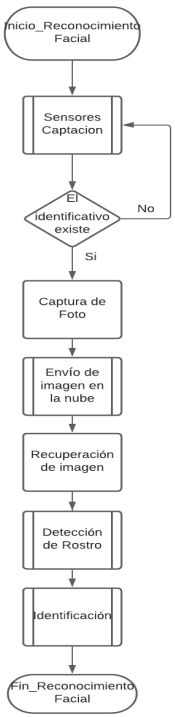
Nota: En el siguiente grafico se indica el diseño del sistema de identificación de personas basado en servicios de reconocimiento facial en la nube.

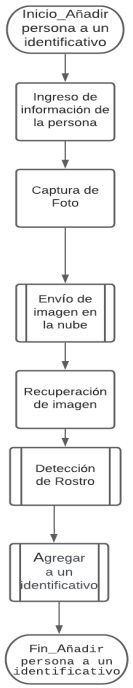
En la figura siguiente se presenta el diagrama de proceso, en el cual se explica los procesos que debe de cumplir del proyecto del sistema de reconocimiento facial, en la cual explica el funcionamiento del sistema.

Figura 15

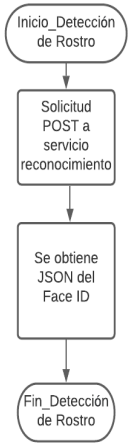
Diagramas de procesos del sistema de identificación de personas basado en servicios de reconocimiento facial en la nube.

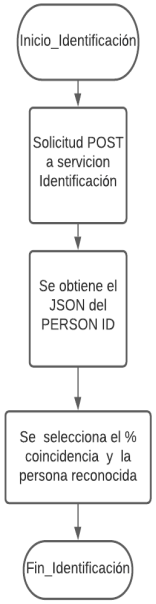

Inicio_Sistema de Identificación basado en servicios de reconocimiento en la nube
Reconocimiento Facial
Añadir persona a identificativo
Publicación de informacion de la persona detectada
Publicación de informacion de la persona entrenada
FIN
FIN

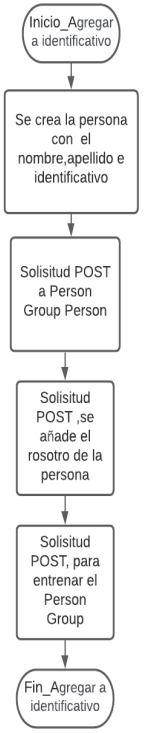

Inicio_Reconocimiento Facial
Sensores Captacion
El identificativo existe
No
Si
Captura de Foto
Envío de imagen en la nube
Recuperación de imagen
Detección de Rostro
Identificación
Fin_Reconocimiento Facial

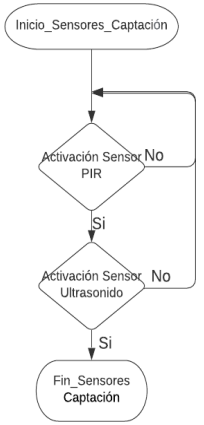

Inicio_Añadir persona a un identificativo
Ingreso de información de la persona
Captura de Foto
Envío de imagen en la nube
Recuperación de imagen
Detección de Rostro
Agregar a un identificativo
Fin_Añadir persona a un identificativo


Inicio_Envío Imagen a la Nube
Cambiar el nombre a la hora que fue capturada la foto
Envío de parámetros de clave y cuenta
Indicar el contenedor donde se va almacenar
Generar el url de la imagen
Fin_Envío Imagen a la nube


Inicio_Detección de Rostro
Solicitud POST a servicio reconocimiento
Se obtiene JSON del Face ID
Fin_Detección de Rostro


Inicio_Identificación
Solicitud POST a servicion Identificación
Se obtiene el JSON del PERSON ID
Se selecciona el % coincidencia y la persona reconocida
Fin_Identificación


Inicio_Agregar a identificativo
Se crea la persona con el nombre,apellido e identificativo
Solisitud POST a Person Group Person
Solisitud POST ,se añade el rosotro de la persona
Solisitud POST, para entrenar el Person Group
Fin_Agregar a identificativo


Inicio_Sensores_Captación
Activación Sensor PIR
No
Si
Activación Sensor Ultrasonido
No
Si
Fin_Sensores Captación

A continuación, se indican los recursos y procedimientos de los servicios en la nube utilizado para el desarrollo del proyecto.

Recursos de Blob Storage

Microsoft Blob Storage brinda tres tipos de recursos como se indica en la siguiente Tabla.

Tabla 5

Descripción de los recursos de Azure Blob Storage

Recurso	Descripción
Cuenta de almacenamiento	Es un espacio de almacenamiento para datos, a la cual se accede mediante una dirección de base de objetos. Esta cuenta de almacenamiento puede tener un número ilimitado de contenedores
Contendor	Es un directorio que organiza un conjunto de archivos, cada contenedor puede almacenar un número ilimitado de blobs.
Datos Blobs	Son archivos de almacenamiento en la nube, por medio del cual acceden las aplicaciones para leer o escribir datos, la diferencia con los archivos estáticos, a estos se puede permitir el acceso desde cualquier sitio con conexión a Internet (Microsoft Azure, 2021). Existen tres tipos de blobs: **Blobs en bloque.** – Almacenan gran cantidad de datos binarios y texto, aproximadamente 190.7 TiB

	Blobs en anexo. -Están optimizados para operaciones de anexión, resultan muy útiles para escenarios de registro de datos en máquinas virtuales.
	Blobs en páginas. -Almacenan archivos de acceso aleatorio de tamaño de hasta 8TiB, como puede ser archivos de disco duro virtual y disco de máquinas virtuales.

Nota: Esta tabla muestra los recursos que se utilizan en el servicio Azure Blob Storage utilizado para el almacenamiento de imágenes (Microsoft Azure , 2021).

Estructuras de datos relacionados de Face API de Microsoft Azure

Son estructuras que son utilizadas por los procedimientos de reconocimiento que se almacenan en forma de objetos en la nube y se puede acceder a ellas mediante cadenas de identificación, estas cadenas de identificación son únicas dentro de cada localidad de suscripción como se indica en la siguiente Tabla.

Tabla 6

Descripción de la estructura de datos relacionados.

Nombre de la operación	Descripción de la operación
DetectedFace	Permite recuperar las representaciones faciales, el identificador de la operación expira 24 horas después de su creación.
PersistedFace	Permite nombrar a cada uno de los objetos de las representaciones faciales para que se conviertan en objetos **PersistedFace**, los cuales se pueden recuperar en cualquier momento y no caducan.

Face List o Large Face List	Esta estructura permite listar los objetos **PersistedFace,** este objeto **FaceList.** Consta de cadenas de nombre, datos de usuario y un identificador único.
Person	Es una lista de objetos **PersistedFace** que permite a la misma persona. Tiene cadenas de nombre, datos de usuario y un identificador único.
PersonGroup o LargePersonGroup	Esta estructura de datos presenta una lista de objetos **Person**, tiene un identificador único, cadenas de nombre y de datos de usuario. La estructura de PersonGroup debe someterse a la operación **Train,** antes de someterse a operaciones de reconocimiento.

Nota: Esta tabla muestra la descripción de las operaciones que se utilizan el servicio de Face de Microsoft Azure (Microsoft Azure, 2019).

Las estructuras de datos mencionados con anterioridad a través de sus atributos permiten realizar los procedimientos identificación y verificación de rostros. Para que se pueda realizar estos procedimientos se debe crear un PersonGroup o Large PersonGroup para guardar a la persona o un conjunto de personas que va a evaluar la coincidencia. La estructura de datos PersonGroups contiene objetos Person, de los cuales cada uno representa a una persona que contiene a su vez un grupo de atributos faciales que pertenecen a cada individuo. Finalmente, la operación Entrenar reúne el grupo de datos que usara para establecer la coincidencia entre los rostros.

Verificación

Se utiliza la operación Verify, la cual direcciona un identificador de rostro de un objeto DetectedFace o PersistedFace y solo un objeto Person. A partir de esto puede

establecer si la cara pertenece a la misma persona. La coincidencia se lo realiza uno a uno y opcionalmente se puede usar como una verificación final de los resultados al usar el procedimiento identificación (Microsoft Azure, 2019).

Identificación

Se utilizar la operación Identify que recoge uno o varios identificadores de la cara de un objeto DetectedFace o PersistedFace y PersonGroup o LargePersonGroup, como respuesta regresa la lista de objetos Person que a través de los procesos de comparación del rostro del individuo con el PersonGroup determine una de las posibles caras de origen. Los objetos Person que serán representados como objetos candidate, con un valor confianza de predicción (Microsoft Azure, 2019).

Datos de entrada

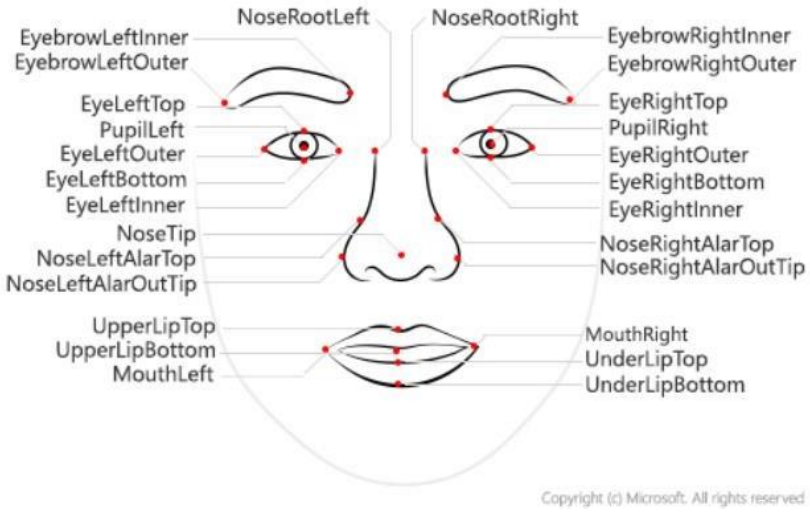
Para realizar los procedimientos anteriores, es necesario el ingreso de la imagen, para que la API Face obtenga resultados de reconocimientos más precisos se debe tener en cuenta lo siguiente:

- El peso del archivo de la imagen no debe exceder los 6 MB.
- Los formatos aceptados son JPEG, PNG y BMP
- Para mejorar la precisión en los rostros se debe tener en cuenta los siguientes pormenores técnicos en las imágenes de entrada como son:
    - Obstrucciones que se interpongan a uno o los dos ojos.
    - Imágenes con fuerte iluminación como la contraluz.
    - Alteraciones en el rostro como exceso en maquillaje, pelo o vello facial e incluso cambios en la apariencia debido a la edad.
    - Expresiones faciales forzadas.

Las siguientes indicaciones permitirán mejorar la detección del rostro y a partir de esto establecer un campo faceRectangle donde se encuentran las coordenadas de cada atributo del rostro, Face API para este propósito establece puntos de referencia en el rostro para encontrar el rostro las cejas, pupilas, tabique de la nariz, entre otros. De forma predeterminada utilizan 27 puntos predefinidos que se indican en la siguiente figura.

Figura 16

Puntos predefinidos ubicados en el rostro para realizar el reconocimiento de personas



Nota. Coordenadas de los 27 puntos de referencia para detección de atributos del rostro. Tomado de (Microsoft Azure, 2019).

Atributos

Son un grupo de rasgos y particularidades que se pueden opcionalmente detectar.

- Sexo. - De acuerdo con el rostro del individuo, los resultados pueden ser hombre, mujer e indefinido.
- Edad. – Estimación de los años del individuo.
- Accesorios. – Puede devolver respuestas si lleva artículos puestos en todo el rostro como gafas, mascarilla con puntuación de uno y cero.
- Vello facial. -Indica un estimado de vello facial en todo el rostro.

- Pelo. -Identifica el tipo de pelo que se encuentra cercano al rostro, su color, así como la calvicie.
- Emoción. -Presenta una puntuación de todas las emociones que sumen uno, las emociones detectadas son tristeza, felicidad, ira, desprecio, asco, sorpresa, temor y neutralidad.

Modelos detección y reconocimiento

El servicio de Face utiliza algoritmos de aprendizaje automático para llevar a cabo para la detección mediante la extracción de características del rostro y usarlo en procedimientos de reconocimiento y detección. Actualmente el servicio de Face tiene 3 diferentes modelos de detección como se indica en la Tabla 7 y 6 de modelos de reconocimiento que se detalla en la siguiente Tabla.

Tabla 7

Descripción de los modelos de detección.

Modelo de detección	Características
DETECTION_01	● Predeterminado para la detección de caras ● Dificulta de detección en perfiles borrosos ● Regresa los atributos del rostro como (posición, cabeza. emociones, etc.) ● Regresa los puntos de referencia para la detección

DETECTION_02	● Mayor precisión a detection_01 en perfiles borrosos, o rostros pequeños ● No regresa ningún atributo del rostro ● No regresa puntos de referencia del rostro.
DETECTION_03	● Es el último modelo de actualización, publicado en 2021 ● Mejor precisión en rostros pequeños y con diferentes orientaciones en el rostro. ● Regresa los puntos de referencia del rostro, si se especifica en la detección

Nota. Esta tabla muestra los modelos de detección utilizados en el servicio Face de Microsoft Azure. Tomado de (Microsoft Azure, 2019).

Tabla 8

Descripción de los modelos de reconocimiento

Modelo de Reconocimiento	**Características**
RECOGNITION_01	● Modelo de reconocimiento predeterminado en febrero de 2019
RECOGNITION_02	● Modelo de reconocimiento lanzado en marzo 2019

RECOGNITION_03	● Modelo de reconocimiento lanzado en mayo 2020
RECOGNITION_04	● Es el modelo más actualizado con la capacidad mejorando la precisión de en comparaciones de similitud y coincidencia de personas, además de la capacidad de reconocer cubiertas faciales en cualquier parte del rostro.

Nota. Esta tabla muestra los modelos de reconocimiento utilizados en el servicio Face de Microsoft Azure. Tomado de (Microsoft Azure, 2019).

De acuerdo con la información de los diferentes modelos de reconocimiento se utilizó el modelo detection_03, debido a que cuando se realice la detección de movimiento el rostro de la persona va a estar en movimiento u orientado del rostro y el modelo Recognition_04 debido a que es la última actualización y presenta mejores prestaciones en precesión de comparación.

El servicio de Face basado en la nube de Azure provee los algoritmos de detección y reconocimiento de caras. Las API de Face que se utilizaron en el desarrollo del sistema de reconocimiento facial, así como sus métodos HTTP y los atributos que debe de ingresar el usuario se indica en la siguiente tabla.

Tabla 9

Descripción y métodos del servicio Face de Microsoft Azure

Nombre de la API del servicio Face	Descripción de la API de Face	Métodos de cada API utilizados
Face	Se encuentran las funciones de detección, buscar similar, grupo, identificar y verificación.	Detect. - POST Identify. -POST
PersonGroup	Se utiliza para administrar los datos de PersonGroup para la identificación.	Create. - PUT Delete. - DELETE Get. - GET List. - GET Train. - POST Update. - PATCH
PersonGroupPerson	Se utiliza para administrar los rostros de personas de PersonGroup para realizar la identificación.	Add Face. - POST Create. - POST Delete. - DELETE Get. - GET List. - GET Update. - PATCH

Nota. Esta tabla muestra la descripción y métodos que se utiliza en las API de Microsoft Azure para realizar el reconocimiento de personas. Tomada de (Microsoft Azure, 2019).

Atributos y Descripción de las estructuras de Face API

En la siguiente tabla se indica la descripción de las estructuras de Face API, con los atributos que se deben completar junto a su descripción, para llevar a cabo cada uno de las APIs utilizadas.

Face

- **Detect**

Tabla 10

Descripción de los campos del servicio Detect-Face

Campos	Atributos	Descripción
Head	Content-Type	Esta cabecera indica el formato de respuesta a la solicitud, de las cuales existen dos: -application/json -application/octet-stream
	Ocp-Apim-Subscription-Key	Esta cabecera se debe llenar únicamente con el Key de suscripción de la API que está utilizando y de acuerdo con la región en la que desea utilizar la API
Request body	url	Se debe ingresar la dirección donde este almacenada la imagen la cual se va a someter a la detección.
Query parameters	returnFaceId	Retorna un valor si la persona fue detectada.
	returnFaceLandmarks	Retorna ubicación de los puntos de referencia de cada parte del rostro.
	returnFaceAttributes	Retorna los atributos que son destacados como edad, pelo, accesorios, exposición, emoción entre otros.
	recognitionModel	Se debe de especificar el método de reconocimiento, de acuerdo con el Capitulo II.
	returnRecognitionModel	Retorna cual es el método de reconocimiento utilizado.

	detectionModel	Se debe de especificar el método de detección, de acuerdo con el Capitulo II.
	faceIdTimeToLive	Se debe especificar el tiempo de respuesta para realizar la detección.
Request url	https://region/face/v1.0/detect?returnFaceId=$$&returnFaceLandmarks=$$&recognitionModel=$$&returnRecognitionModel=$$&detectionModel=&&faceIdTimeToLive=&&	
Método	POST	

Nota: Esta tabla muestra los diferentes campos para el servicio Face-Detect

- **Identify**

Tabla 11

Descripción de los campos del servicio Identify-Face

Campos	Atributos	Descripción
Head	Content-Type	Esta cabecera indica el formato de respuesta a la solicitud, de las cuales existen dos: -application/json -application/octet-stream
	Ocp-Apim-Subscription-Key	Esta cabecera se debe llenar únicamente con el Key de suscripción de la API que está utilizando y de acuerdo con la región en la que desea utilizar la API
Request body	largePersonGroupId o PersonGroupId	Se debe ingresar la dirección donde este almacenada la imagen la cual se va a someter a la detección.
	faceIds	Se debe ingresar el rostro o los rostros que pasaron por la API Detect de Face para comparar con un el PersonGroupId

	maxNumOfCandidatesReturned	Se debe de indicar el número de candidatos de las personas que puedan identificar
	confidenceThreshold	Se debe indicar el valor del umbral de confidencialidad
Request url	https://region/face/v1.0/identify	
Método	POST	

Nota: Esta tabla muestra los diferentes campos para el servicio Face-Identify

PersonGroup

- **Create**

Tabla 12

Descripción de los campos del servicio PersonGroup-Create

Campos	Atributos	Descripción
Head	Content-Type	Esta cabecera indica el formato de respuesta a la solicitud, de las cuales existen dos: -application/json -application/octet-stream
	Ocp-Apim-Subscription-Key	Esta cabecera se debe llenar únicamente con el Key de suscripción de la API que está utilizando y de acuerdo con la región en la que desea utilizar la API
Request body	name	En ambos casos se puede añadir información relevante del identificativo del PersonGroup
	userData	
	recognitionModel	Se debe de indicar el modelo de reconocimiento

Query parameters	personGroupId	Se debe asignar el nombre del grupo de personas, que tienen un parentesco definidas por un identificativo como son detalles de la familia entre otros.
Request url	https://region/face/v1.0/persongroups/{personGroupId}	
Método	PUT	

Nota: Esta tabla muestra los diferentes campos para el servicio PersonGroup-Create

- **Train**

Tabla 13

Descripción de los campos del servicio PersonGroup-Train

Campos	Atributos	Descripción
Head	Ocp-Apim-Subscription-Key	Esta cabecera se debe llenar únicamente con el Key de suscripción de la API que está utilizando y de acuerdo con la región en la que desea utilizar la API.
Query parameters	personGroupId	Se debe poner el nombre del PersonGroup, para entrenar el grupo de personas, representadas por el identificativo
Request url	https://region/face/v1.0/persongroups/{personGroupId}/train	
Método	POST	

Nota: Esta tabla muestra los diferentes campos para el servicio PersonGroup-Train.

PersonGroupPerson

- **Create**

Tabla 14

Descripción de los campos del servicio PersonGroupPerson-Create

Campos	Atributos	Descripción
Head	Content-Type	Esta cabecera indica el formato de respuesta a la solicitud, de las cuales existen dos: -application/json -application/octet-stream
	Ocp-Apim-Subscription-Key	Esta cabecera se debe llenar únicamente con el Key de suscripción de la API que está utilizando y de acuerdo con la región en la que desea utilizar la API.
Request body	name userData	En ambos casos se puede añadir información relevante del PersonGroupPerson, como son el nombre y apellido de la Persona.
Query parameters	personGroupId	Se debe poner el nombre del PersonGroup del cual pertenece la persona.
Request url	https://region/face/v1.0/persongroups/{personGroupId}/persons	
Método	POST	

Nota: Esta tabla muestra los diferentes campos para el servicio PersonGroup-Create.

- **Add Face**

Tabla 15

Descripción de los campos del servicio PersonGroupPerson-Add Face.

Campos	Atributos	Descripción
Head	Content-Type	Esta cabecera indica el formato de respuesta a la solicitud, de las cuales existen dos: -application/json -application/octet-stream
	Ocp-Apim-Subscription-Key	Esta cabecera se debe llenar únicamente con el Key de suscripción de la API que está utilizando y de acuerdo con la región en la que desea utilizar la API.
Request Body	url	Se debe ingresar la dirección donde este almacenada la imagen la cual se va a someter a la detección.
Query Parameters	personGroupId	Se debe poner el nombre del PersonGroup del cual pertenece la persona.
	personId	Se debe de poner el código como respuesta del proceso de Create, de la persona creada con su información.
	detectionModel	Se debe de especificar el método de detección, de acuerdo con el Capitulo II.
Request Url	https://region/face/v1.0/persongroups/{personGroupId}/persons/{personId}/ persons/persistedFaces?detectionModel=detection_01	
Método	POST	

Nota: Esta tabla muestra los diferentes campos para el servicio PersonGroup-Add Face.

- **Get**

Tabla 16

Descripción de los campos del servicio PersonGroupPerson-Get

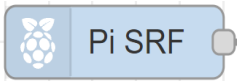
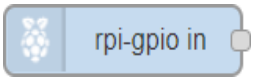
Campos	Atributos	Descripción
Head	Ocp-Apim-Subscription-Key	Esta cabecera se debe llenar únicamente con el Key de suscripción de la API que está utilizando y de acuerdo con la región en la que desea utilizar la API.
Query parameters	personGroupId	Se debe poner el nombre del PersonGroup del cual pertenece la persona.
	personId	Se debe de poner el código como respuesta del proceso de Create, de la persona creada con su información.
Request url	https://region/face/v1.0/persongroups/{personGroupId}/persons/{personId}	
Método	GET	

Nota: Esta tabla muestra los diferentes campos para el servicio PersonGroup-Get.

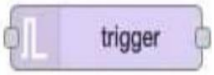
Cada uno de los servicios de API Face como Azure Blob Storage, serán utilizados en el software de programación grafico Node-RED, a través de diversos nodos interconectados con otros que permitan medir y detectar la proximidad de la persona y la captura del rostro de esta. En la siguiente Tabla se realiza la descripción de los nodos para el desarrollo del proyecto.

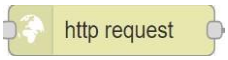
Tabla 17

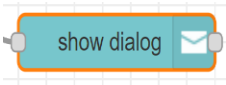
Nodos utilizados para el desarrollo del proyecto

Tipo de Nodo	Nombre del Nodo	Descripción
Rasberry PI	Rpi-srf 	Este nodo es de uso exclusivo de la Rasberry Pi que permite medir distancias a través del uso del ultrasonido, este nodo utiliza dos pines para ser activado y desactivado. Una vez activado el ultrasonido establecerá un rango de frecuencia, esto lo realizará por defecto en 0.5 s y da como resultado el msg.payload con un número que representa el rango en cm.
	Rpi-Gpio in 	Este nodo trabaja en conjunto con Rasberry Pi, usa una Liberia de phyton de raspbian para poder ser usado, permita detectar los pines físicos y envía una señal de tipo booleana en el msg.payload.
De prueba o comunes	Debug 	Se utiliza para mostrar mensajes en la barra lateral que proporciona una vista estructurada de los mensajes que se envían. A lado de cada mensaje la barra debug incluye información sobre la hora en la que se recibe el mensaje y que nodo lo

		envió. El botón de nodo debug se utiliza para habilitar o deshabilitar su salida.
De función	Function 	Permite ejecutar el código JavaScript en los mensajes que se transmiten a través de él. El mensaje se va como un objeto llamado msg. Por convención, tendrá una msg.payload propiedad que contiene el cuerpo del mensaje. En cambio, otros nodos se pueden adjuntar sus propias propiedades al mensaje y deben describirse en el documento.
	Change 	Se utiliza para cambiar las propiedades de un mensaje y establecer propiedades de contexto sin necesidad de recurrir a un nodo Función. Entre sus principales funciones están la de establecer una propiedad, remplazar partes de la propiedad de un mensaje, o mover y cambiar una propiedad.
	Switch 	Permite que los mensajes se enruten a las otras ramas de un flujo. Está configurado con la propiedad para

	aprobar basándose en cuatro tipos de reglas para determinar si es una propiedad de mensaje o una propiedad de contexto.
Delay 	Tiene dos funciones importantes, la primera función es la de permitir que se retrase un mensaje por una cantidad de tiempo arbitraria y la segunda función es limitar la velocidad de los mensajes que pasan a través de él.
Trigger 	Tiene la función de poder repetir un mensaje en un periodo arbitrario y configurarlo para que propague cualquier objeto de mensaje que reciba en su entrada y lo reenvié en un intervalo de tiempo.
Template 	Puede ser utilizado para generar texto, utilizando propiedades de un mensaje para completar una plantilla. La plantilla es validad para JSON o YALM para realizar un análisis de configuración de resultado en JavaScript. Por último, el nodo podrá ser usado para crear una dinámica

		interfaz de usuario con el fin cambiar la apariencia de la interfaz.
	Exec 	Permite ejecutar cualquier tipo de comando del sistema operativo existente, como programas como Python u otra secuencia de comandos en Node-red e incorporar los resultados en su flujo a través de un msg.payload. El nodo toma una sola entrada y tiene tres salidas.
De red	http request 	Esta función realiza una recuperación web de un sitio web, mediando la realización de un Api Request (Solicitud de la respuesta), enviando y recibiendo un jSON o UTF-8 a una página web o a un API. Adicionalmente el nodo puede cambiar automáticamente la carga de información o el msg. payload como instrucciones de cadena de consulta para una solicitud de HTTP Request.
Herramientas de imagen	Jimp-image 	Permite la salida de la imagen, además de ser útil cuando se quiere realizar un programa para publicar una imagen. Para ejecutarlo solo se

		tiene que escoger la url de la imagen que desee poner y descárgala en el node-red para que se le pueda colocar en el interfaz.
Dashboard	Ui_button 	Permiten trabajar y ejecutar tareas, de funciones o templates, además de intercambiar ventanas o grupos de pantalla, haciendo clic en el botón el cual genera un msg.payload.
	Ui_toast 	Funciona como una alerta msg.payload de notificación de ok y cancel mediante un mensaje de dialogo en la interfaz de usuario. Una vez el usuario ya haya elegido su respuesta se retornar a msg.payload y depende de su respuesta a la función continuará trabajando correctamente o por el contrario lo cancelará y el usuario tendrá que repetir el mismo proceso.
	Ui_Form 	Permite recolectar diversa información de cualquier usuario, el cual envía los mensajes al siguiente nodo de tipo msg.payload. En el que se puede indicar el tipo de mensaje

		sea texto, contraseña, email, checkbox, switch o fecha, además del label que aparecerá en el dashboard.
	Ui-Control 	Permite el control de menús de tablas y grupos dinámico dentro del entorno grafico del dashboard.
Nube de Azure	Save Blob 	Permite crear y eliminar contenedores y archivos blob. Su función principalmente es la de guardar archivos en una cuenta con clave y contenedor de la nube de Azure.

Nota: Tomado de (Node-RED, 2021).

Capítulo IV

DESARROLLO E IMPLEMENTACION

Desarrollo

El desarrollo del sistema de reconocimiento en la nube se divide en 2 procesos diferentes los cuales usan los servicios de Reconocimiento e Identificación, que dependerá de la necesidad del cliente:

- Reconocimiento de personas
- Agregar personas a un identificativo

Para el desarrollo del proyecto se utilizó los servicios en la nube de Azure como son la API Face y Azure Blob Storage, los cuales serán utilizados e implementados en el software Node-RED, para ello previamente se debe instalar el sistema operativo

Raspbian en conjunto con el escritorio y el framework Node-RED para la Rasberry Pi, para ello se deben de seguir los siguientes pasos.

Instalación del Sistema Operativo de la Rasberry Pi

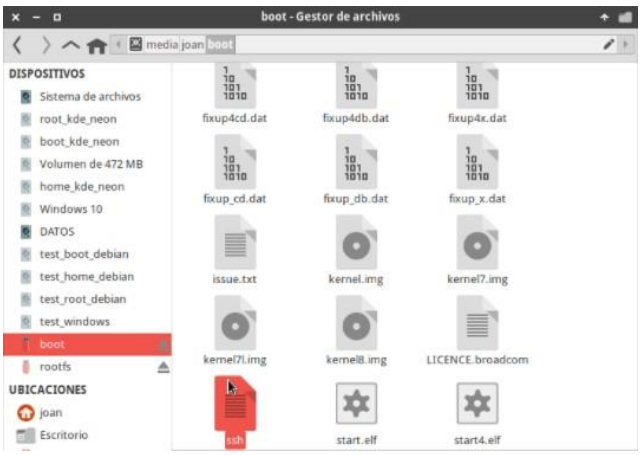
Raspbian es un sistema operativo gratuito basado en Debian optimizado para el hardware Raspberry Pi, con interfaz gráfica en PIXEL y tamaño de 4Gb en su versión completo

Como primer paso se debe instalar la imagen del SO de la Rasberry PI en Linux, descargado el archivo binario y se instalar como un archivo. amd64, seguido de esto se debe copiar el archivo a la tarjeta Micro SD.

Como segundo paso para acceder a la Rasberry vía SSH, acceder al directorio boot y crear un archivo vacío de nombre ssh, para habilitar la conexión via SSH, como se indica en la siguiente figura.

Figura 17

Creación del archivo para acceso mediante SSH



Como tercer paso se debe instalar la interfaz gráfica del escritorio, para ello se debe ingresar a un cliente SSH, en el cual se debe registrar el usuario y contraseña y a continuación seguir la siguiente línea de comandos para la instalación.

sudo apt-get install --no-install-recommends xserver-xorg

sudo apt-get install --no-install-recommends xinit

sudo apt-get install raspberrypi-ui-mods

sudo apt-get install -y rpi-chromium-mods

sudo apt-get install -y python-sense-emu-doc realvnc-vnc-viewer

sudo apt-get install -y python-sense-emu python3-sense-emu python-sense-emu-doc

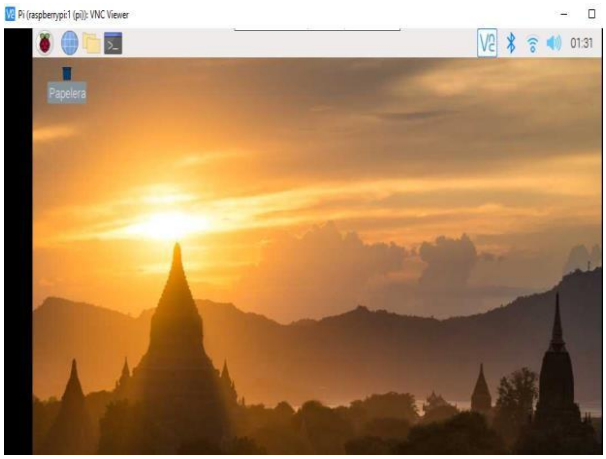
sudo apt-get install lightdm

sudo reboot

En la siguiente figura se indica la interfaz gráfica de Raspbian.

Figura 18

Interfaz gráfica de Raspbian



Instalación de Node RED

Como cuarto paso se debe instalar Node RED y configurar este software para el arranque al iniciar el dispositivo

sudo apt-get install npm (Instala npm)

sudo npm install -g npm (Actualiza npm a la última versión en modo global)

sudo systemctl restart nodered.service

cd ~/.node-red

npm install node-red

sudo su –

Codigo para inicializar el arranque de Node-RED

wget https://raw.githubusercontent.com/node-red/raspbian-deb

package/master/resources/nodered.service -O

/lib/systemd/system/nodered.service

wget https://raw.githubusercontent.com/node-red/raspbian-deb-

package/master/resources/node-red-start -O /usr/bin/node-red-start

wget https://raw.githubusercontent.com/node-red/raspbian-deb-

package/master/resources/node-red-stop -O /usr/bin/node-red-stop

chmod +x /usr/bin/node-red-st*

systemctl daemon-reload

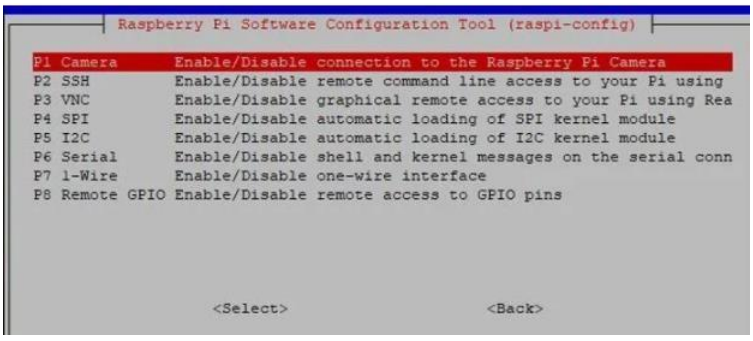
systemctl enable nodered.service

shutdown -r now

Finalmente se debe de activar la cámara de la Rasberry PI, para ello debemos ejecutar el comando **sudo raspi-config**, para ello se debe seleccionar la opción Interfacing Options y a continuación activar la opción Camara y Remote GPIO.

Figura 19

Configuración de interfaces para Rasberry PI



Una vez realizado estos procedimientos los pines GPIO y la cámara podrán ser utilizados para recolectar datos y ejecutar comandos desde el framework de Node-RED.

Procesos del sistema de reconocimiento facial

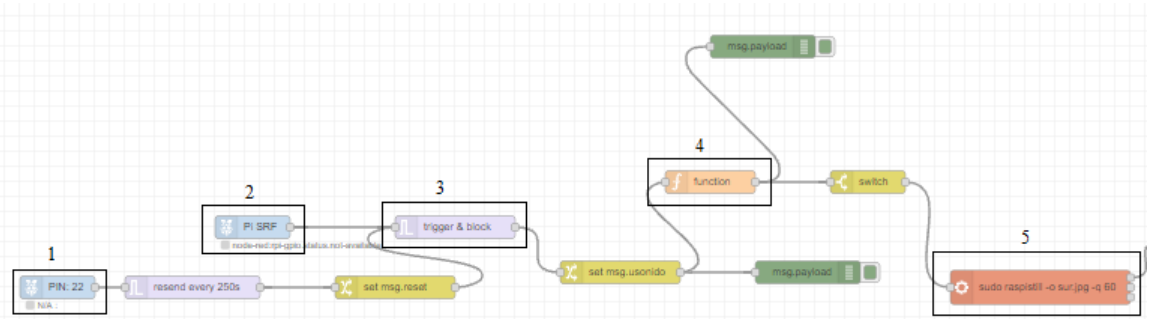
Reconocimiento de personas

- **Administración de datos de los sensores**

Para realizar la captura del rostro de la persona se usa los sensores PIR y de ultrasonido, los cuales al detectar la presencia de personas de acuerdo la activación del sensor PIR y la distancia que se encuentre el individuo del sensor de ultrasonido, al cumplirse estos dos parámetros se ejecuta el comando de la captura de foto, lo cual se detalla en la siguiente figura.

Figura 20

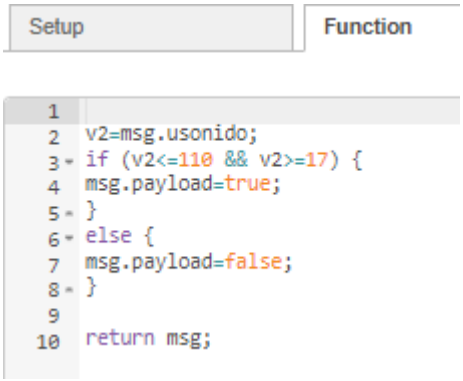
Grupo de nodos para la Administración de datos de los sensores



En los nodos 1 y 2 se receptan los datos del sensor PIR y de ultrasonido respectivamente, en los cuales se especifica el PIN de salida hacia la Rasberry Pi, el nodo 3 permite realizar un procedimiento de trigger y block el cual permite que los datos permitan seguir utilizados siempre y cuando el sensor PIR se active, en el nodo 4 function permite establecer un condicional al sensor de ultrasonido con respecto a la distancia como se indica en la siguiente figura.

Figura 21

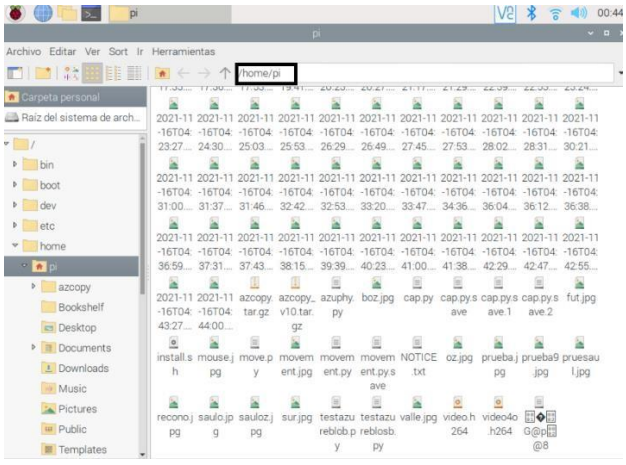
Contenido de la función del nodo 5



Posteriormente en el nodo 5 se ejecuta el comando “**sudo raspistill -o sur.jpg -q 60**” en la Rasberry PI mediante el nodo exec, el cual realiza la captura del rostro. Los datos estarán almacenados en el directorio /home/pi como se indica en la siguiente imagen.

Figura 22

Directorio de Rasberry para almacenar datos



- **Almacenamiento de imagen de rostros en la nube**

Previamente para el almacenamiento de imagen en la nube se debe contar con un contenedor de Azure Blob Storage

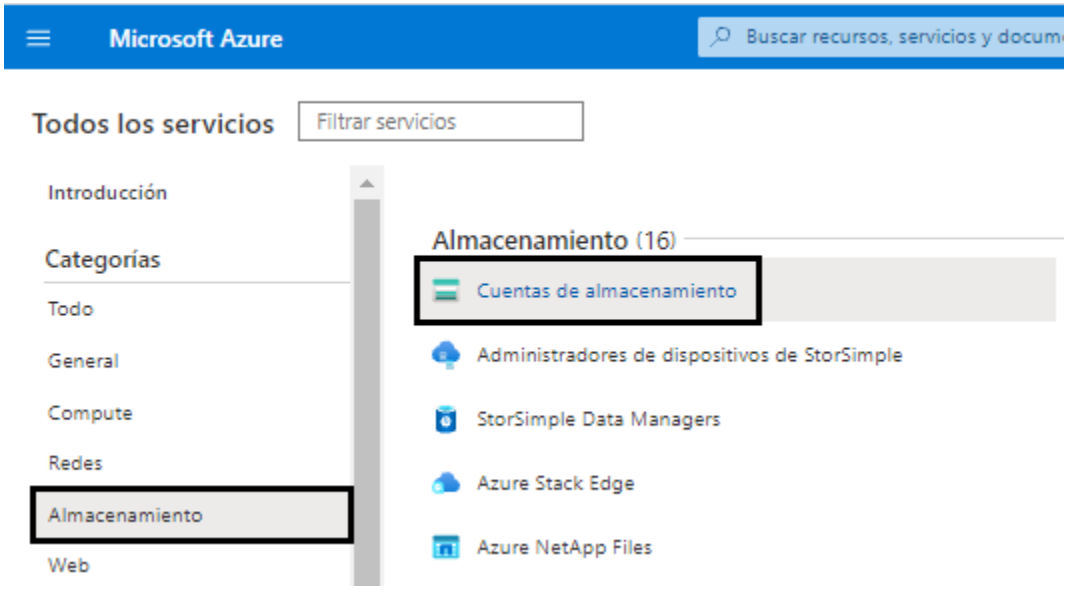
Creación del contenedor de Azure Blob Storage

Como ya se explicó en el Capítulo III se debe de contar con la clave y el usuario del desarrollador para usar los servicios de Azure. Los procedimientos para la creación del contenedor se realiza los siguientes pasos:

En primer lugar, se debe seleccionar la opción crear recursos, ubicarse en la categoría Almacenamiento y seleccionar Cuentas de almacenamiento, como se observa en la siguiente figura.

Figura 23

Creación del contenedor de Azure Blob Storage

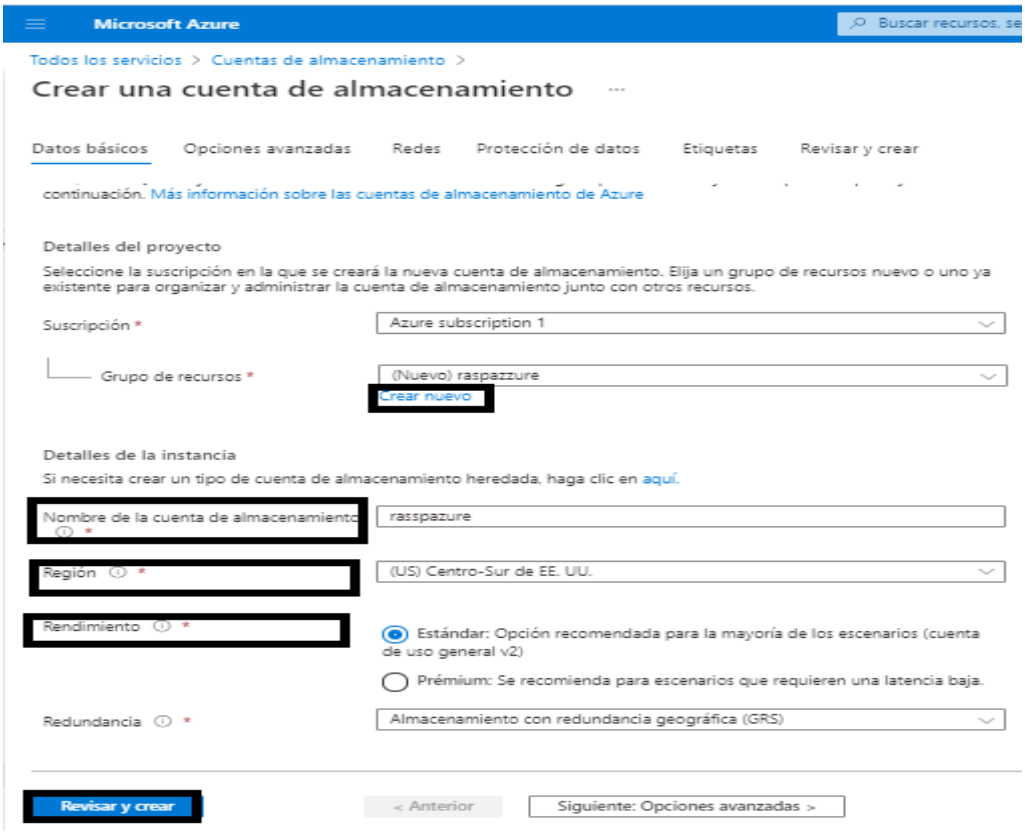


Nota: Tomado de (Microsoft Azure, 2021).

A continuación, se despliega los datos que se necesitan llenar, como dar el nombre del grupo del recurso, nombre de la Cuenta de almacenamiento, la Región en la que se va a ubicar el recurso y el rendimiento que depende del tipo de Cuenta de Azure, por último, se debe seleccionar **Revisar y Crear,** como se indica en la siguiente figura.

Figura 24

Registro de especificaciones para la creación del contenedor



Nota: Tomado de (Microsoft Azure, 2021).

Una vez creada la cuenta se debe seleccionar la opción Contenedores y seleccionar **+ Contenedores** y seleccionar el nombre para el Blob, como se indica en la siguiente figura

Figura 25

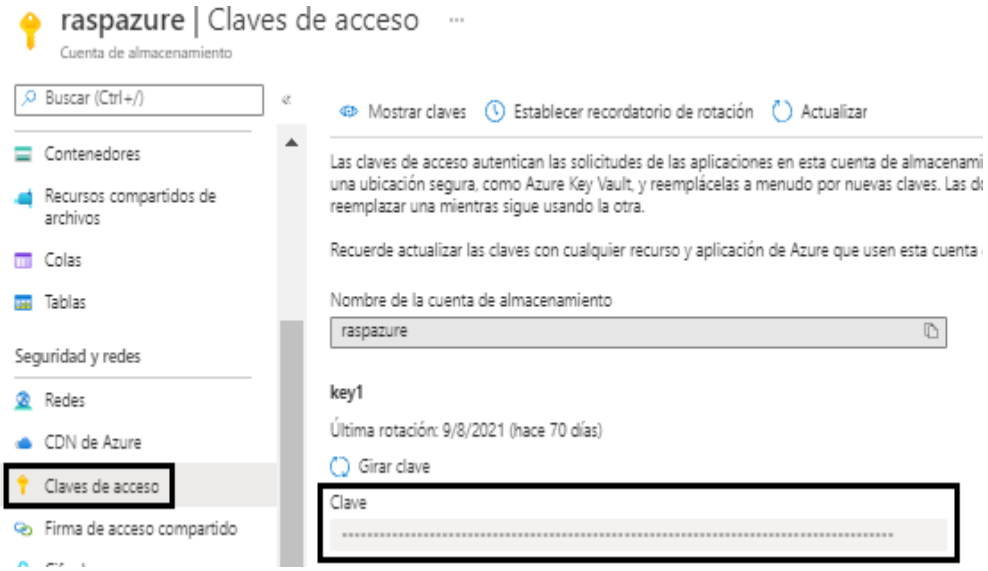
Ventana de trabajo del Contenedor de Azure Blob Storage

Nota: Tomado de (Microsoft Azure, 2021).

Como último paso para la utilización del recurso en Node-RED, se debe seleccionar Claves de acceso, la cual permitirá autenticar las solicitudes de aplicaciones de Azure Blob Storage a través de la clave generada, como se indica en la siguiente figura

Figura 26

Generación de claves para el contenedor de Azure Blob Storage.



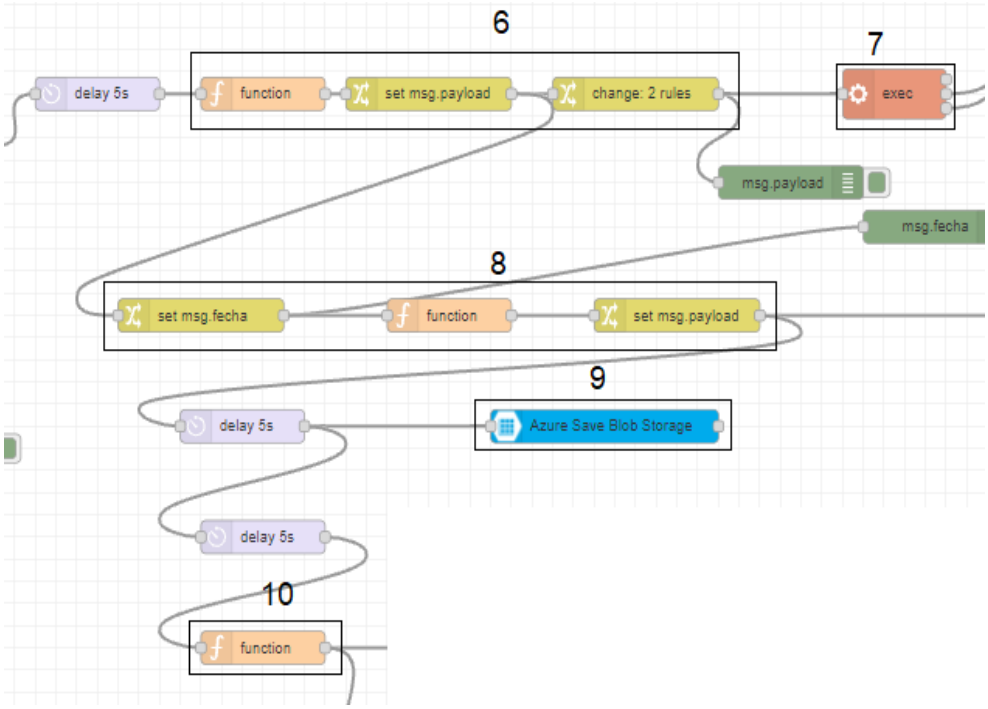
Nota: Tomado de (Microsoft Azure, 2021).

La imagen capturada en el paso anterior se modifica su nombre de acuerdo con la hora y fecha en que fue tomada la foto para ser almacenada en el servicio en la nube de Azure Blob Storage y que esto permita que la url de la imagen puede ser recuperada

de acuerdo en el tiempo que se realizó la captura de la foto como se indica en la siguiente figura.

Figura 27

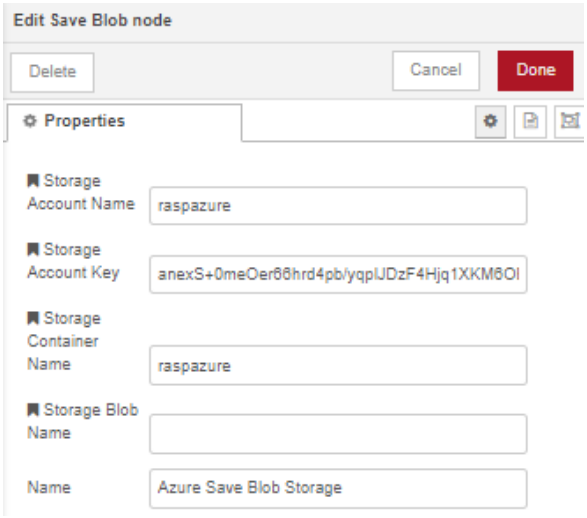
Grupo de nodos para el almacenamiento de imagen de rostros en la nube.



En el grupo de nodos 6 se cambia el nombre de la fotografía por la hora y el día en que fue realizada la captura de la foto y que se ejecutada el comando en la Rasberry PI como “**sudo mv sur.jpg fechayhoraactual.jpg**” en el nodo 7 mediante el nodo exec, en el grupo de nodos 8 se especifica el apuntamiento a la carpeta y nombre del archivo de la imagen capturada para que pueda ser subida al contenedor de Azure Blob Storage en el nodo 9, a continuación se especifica los parámetros que deben ser completados, como se indica en la siguiente figura.

Figura 28

Parámetros de ingreso del nodo 9 Azure Save Blob Storage



En Storage Account Name se llena con el nombre del recurso creado llamado “raspazure” en Account Key se completa con la clave del recurso de Azure Blob Storage que se indicó en la Figura, finalmente se debe llenar el nombre del contendor llamado “raspazure”, el cual permitirá subir el archivo local en la nube.

Finalmente, en el nodo 10, se especifica el formato de la url con el cual se puede recuperar el archivo el cual es “https://raspazure.blob.core.windows.net/raspazure/+(la fecha y hora en la que se tomó la foto)”

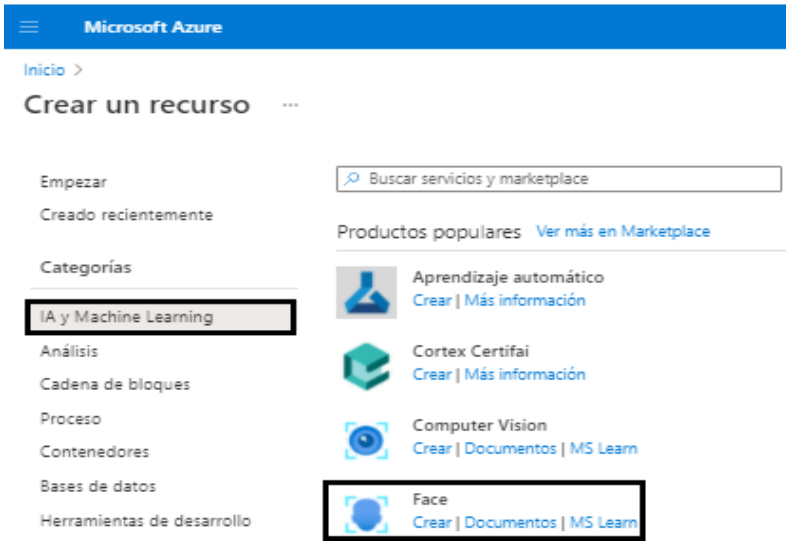
Realizado los procedimientos anteriores, se debe ocupar los servicios de Face API para ello como ya se explicó en el Capítulo III, es necesario la clave y el usuario del desarrollador de Azure, los procedimientos para creación del recurso de API FACE se detallan a continuación.

Creación del recurso Face API

En primer lugar, se debe de Crear un nuevo recurso, ubicarse en la categoría IA y Machine Learning y seleccionar la opción Face en **Crear,** como se indica en la siguiente figura.

Figura 29

Creación del recurso Face API

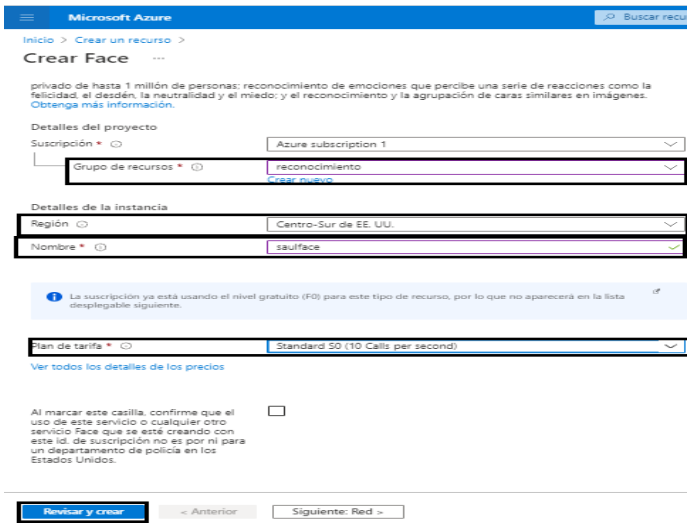


Nota: Tomado de (Microsoft Azure, 2021).

Posteriormente, se despliega los datos que se necesitan llenar, como dar el nombre del grupo del recurso, nombre de la Cuenta de almacenamiento, la Región en la que se va a ubicar el recurso y el rendimiento que depende del tipo de Cuenta de Azure, por último, se debe seleccionar **Revisar y Crear**, como se indica en la siguiente figura.

Figura 30

Especificaciones para la creación del recurso Face API.

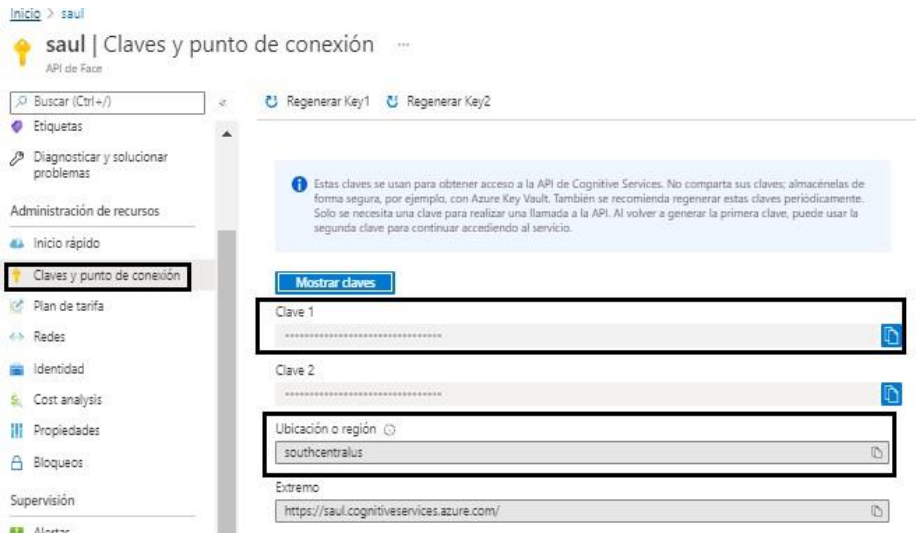


Nota: Tomado de (Microsoft Azure, 2021).

Como último paso para la utilización del recurso en Node-RED, se debe seleccionar Claves de acceso, la cual permitirá autenticar las solicitudes de aplicaciones de FACE Api a través de la clave generada, como se indica en la siguiente figura.

Figura 31

Generación de claves para el recurso Face API.



Nota: Tomado de (Microsoft Azure, 2021).

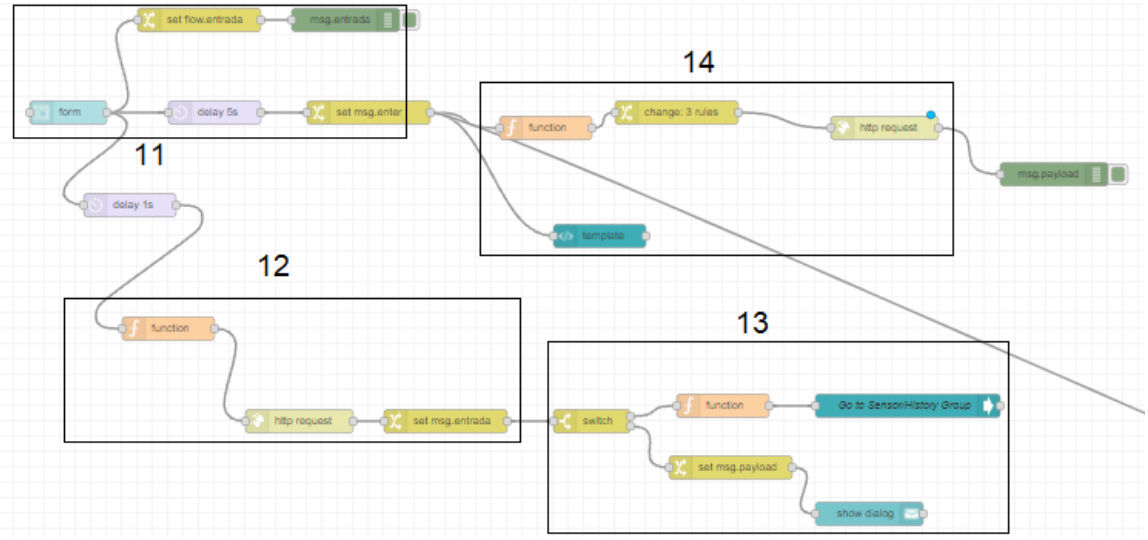
Con la creación del recurso, permitirá el acceso a todos los servicios de la API FACE, para la etapa del reconocimiento de personas se ocupará los servicios Detect, Identify, PersonGroup Person -Update como se indica a continuación.

- **Especificación del identificativo**

Este paso permite seleccionar la familia denotado por un identificativo al cual pertenece y según esta opción, realizar el reconocimiento de personas, para ello el identificativo del grupo familiar debe existir y debe ser entrenado en la API Face como se indica en la siguiente figura.

Figura 32

Grupo de nodos para la especificación del identificativo



En el grupo de nodos 11 se debe especificar el nombre del identificativo y se constituye como una variable de flujo la cual puede ser ocupada en todo el flujo de reconocimiento, el grupo de nodos 12 realiza una solicitud para comparar si el identificativo existe como se indica en la siguiente tabla.

Tabla 18

Contenido de la función y petición del grupo de nodos 12.

	Mensaje	**Contenido**
Función	msg.headers	Content-Type: application/json "Ocp-Apim-Subscription-Key": "807a1ed2a37e47d58a7b5876e0dab7f0"
	CAMPO	**CONTENIDO**
Petición http	Method	GET
	URL	https://southcentralus.api.cognitive.microsoft.com

		/face/v1.0/persongroups?top=1000&return RecognitionModel=false

A continuación, en el nodo 13 se guarda la lista de PersonGroups y se compara con el que se ingresó en el formulario, para lo cual se realiza un condicional para evaluar si el identificativo existe y permita pasar a la ventana de reconocimiento o caso contrario se enviara una notificación de que el identificativo no existe y vuelva escribir otra vez.

Finalmente, en el grupo de nodos 14 se realiza el entrenamiento del identificativo ingresado a través del servicio Train-Person Group, como se indica en la siguiente tabla.

Tabla 19

Contenido de los atributos, función y petición del grupo de nodos 14.

	Variable	**Descripción**
Atributos	msg.payload.persentrain	Contiene el identificativo ingresado
	Mensaje	**Contenido**
Funcion	msg.headers	Content-Type: application/json "Ocp-Apim-Subscription-Key": "807a1ed2a37e47d58a7b5876e0dab7f0"
	msg.payload	
	Campo	**Contenido**
Peticion http	Method	POST
	URL	https://southcentralus.api.cognitive.microsoft.com/face/v1.0/persongroups/msg.paylo ad.persentrain/train

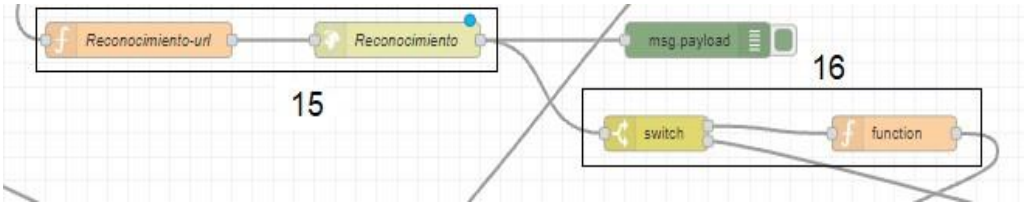
Nota: En el grupo de nodos 11 se debe de registrar el nombre del identificativo que se va a entrenar en la dirección de URL para realizar la petición.

- **Reconocimiento de rostros**

Este paso permite realizar el reconocimiento de rostros a partir de la url de la imagen que se indica en el paso del almacenamiento de imagen de rostros en la nube en el nodo 10 y como respuesta se obtiene el faceId, a continuación, se realiza el condicional para evaluar si el faceId existe, en el caso de existir se almacenara para seguir ocupado en la siguiente etapa, como se indica en la siguiente figura.

Figura 33

Grupo de Nodos de la etapa Reconocimiento de imagen



En el grupo de nodos 15 se especifica la solicitud del reconocimiento, mediante la solicitud HTTP Request al servicio Face-Detect como se indica en la siguiente Tabla.

Tabla 20

Contenido de los atributos, función y petición del grupo de nodos 15

	Variables	**Descripción**
Atributos	msg.datos	Contiene la dirección url de la imagen
	Mensaje	**Contenido**
Función	msg.headers	content-type: application/json

		"ocp-apim-subscription-key": "807a1ed2a37e47d58a7b5876e0dab7f0"
	msg.payload	“url”:msg.datos

	Campo	Contenido
Petición http	Method	POST
	URL	https://southcentralus.api.cognitive.microsoft.com/face/v1.0/detect?returnFaceId=true&returnFaceLandmarks=false&returnFaceAttributes=age,gender,headPose,smile,facialHair,glasses,emotion,hair,makeup,occlusion,accessories,blur,exposure&recognitionModel=recognition_03&returnRecognitionModel=true&detectionModel=detection_01

Nota: En la Tabla 20 se debe de especificar la url de la imagen que está en la nube y especificar el modelo detección y reconocimiento, además de los atributos que desea retornar.

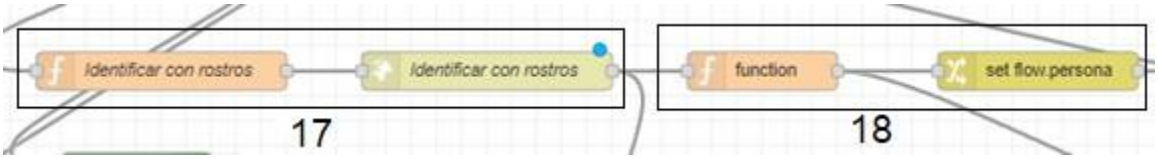
En el grupo de nodos 16 se evalúa si existe el faceId, por lo cual si existe se almacena este valor como variable para el siguiente proceso, en el caso de no existir el faceId se indica que la persona es desconocida.

Identificación de rostros

Este paso permite realizar la identificación de rostros en la cual se compara la imagen capturada por la cámara con el grupo familiar representado por el identificativo, como respuesta se obtiene el PersonId de la persona a la cual tiene mayores similitudes con un grado de confiabilidad, para lo cual en el caso de que se obtenga el PersonId y se almacena esta variable, como se indica en la siguiente figura.

Figura 34

Grupo de nodos para la identificación de rostros



En el grupo de nodos 17 se especifica la solicitud de identificación, mediante la solicitud HTTP Request al servicio Face-Identify como se indica en la siguiente Tabla.

Tabla 21

Contenido de los atributos, función y petición del grupo de nodos 17

	Variable	**Descripción**
Atributos	msg.enter	Contiene el identificativo ingresado
	Msg.faceIds	Contiene el valor del faceId de la persona capturada por la camara
	Mensaje	**Contenido**
Función	msg.headers	Content-Type: application/json

		"Ocp-Apim-Subscription-Key": "807a1ed2a37e47d58a7b5876e0dab7f0"
	msg.payload	“PersonGroupId”:msg.enter “faceIds” : [msg.faceIds], “maxNumoOfCandidatesReturned” :1, “ConfidenceThreshold:0.5

	Campo	Contenido
Petición http	Method	POST
	URL	https://southcentralus.api.cognitive.microsoft.com/face/v1.0/identify

Nota: En la Tabla 21, el cuerpo del mensaje se debe de especificar el PersonGroup con el cual se va a comparar el nuevo rostro representado por el faceId, además del valor de confiabilidad.

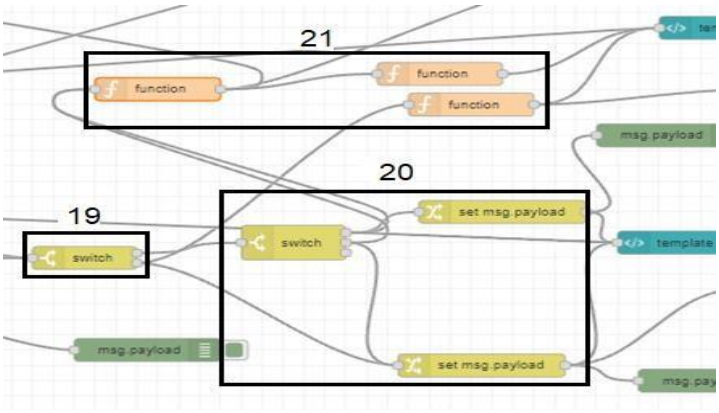
En el grupo de nodos 18 se almacena en una variable el valor del PersonId con mayores similitudes con el rostro de la persona captada por la cámara.

- **Visualización de información de reconocimiento**

Este paso nos permite desplegar la información del porcentaje de confiabilidad y si es que la persona es reconocida dentro del identificativo del grupo familiar, para ser desplegado en el dashboard, como se indica en la siguiente figura.

Figura 35

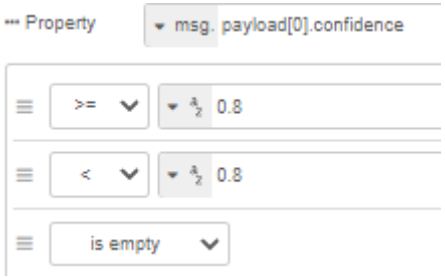
Grupo de nodos para la confiabilidad y reconocimiento



En el grupo de nodos 19 permite condicionar si el valor del PersonId existe y según esta respuesta comunicar a los nodos 20 y 21 de que la persona es Desconocida y que sea desplegada esta información para el nodo 20 se establece el condicional si la persona es desconocida o conocida según el umbral de confiabilidad debe ser mayor o igual que 0.8, mientras en el nodo 21 se indica el valor en porcentaje de confiabilidad en el caso de existir, como se indica en la siguiente figura.

Figura 36

Comparación del valor del umbral para establecer reconocimiento del grupo de nodos 20



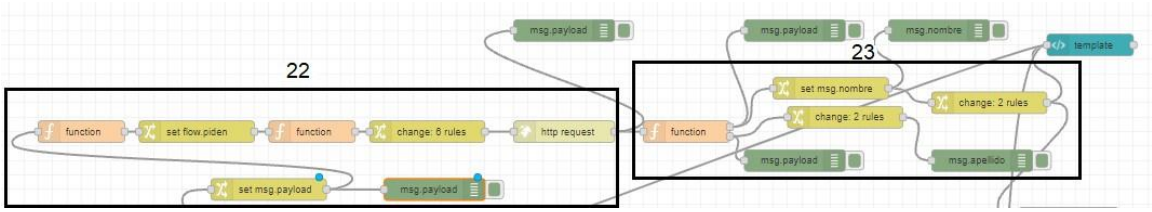
- **Identificación de persona reconocida**

Este paso nos permite desplegar la información de la persona reconocida, según el valor del umbral del grupo de nodos 20, para realizar este proceso se almacena el valor del PersonId y se realiza la solicitud al servicio Person Group Person- Update el cual permite regresar los parámetros del nombre y apellido de la persona que está

registrada dentro del identificativo que se está realizando el reconocimiento, como se indica en la siguiente figura.

Figura 37

Grupo de nodos para el reconocimiento de personas.



En el nodo 22 se registra la variable del PersonId del grupo de nodos 18 y se realiza la peticion HTTP Request, al servicio de Person Group Person Update como se indica en la siguiente Tabla.

Tabla 22

Contenido de la función y petición del grupo de nodos 22

	Variable	**Descripción**
Atributos	msg.entrada	Contiene el identificativo ingresado
	Msg.identificar	Contiene el valor del PersonId de la persona que tiene similitud
	Mensaje	**Contenido**
Funcion	msg.headers	Content-Type: application/json "Ocp-Apim-Subscription-Key": "807a1ed2a37e47d58a7b5876e0dab7f0"
	Msg.payload	

	Campo	Contenido
Peticion http	Method	GET
	URL	https://southcentralus.api.cognitive.microsoft.com/face/v1.0/persongroups/msg.entrada/person s/msg.identificar

Nota: En la Tabla en el campo de URL de la petición se debe incluir del valor del personId, que se obtuvo de la etapa anterior para que se regrese los datos de registro de nombre y apellido.

En el grupo de nodos 23 se organiza la información del nombre y apellido para que pueda ser desplegado en una sola línea y presentar el dashboard.

Añadir persona a un identificativo

- **Captura de Rostro de la imagen**

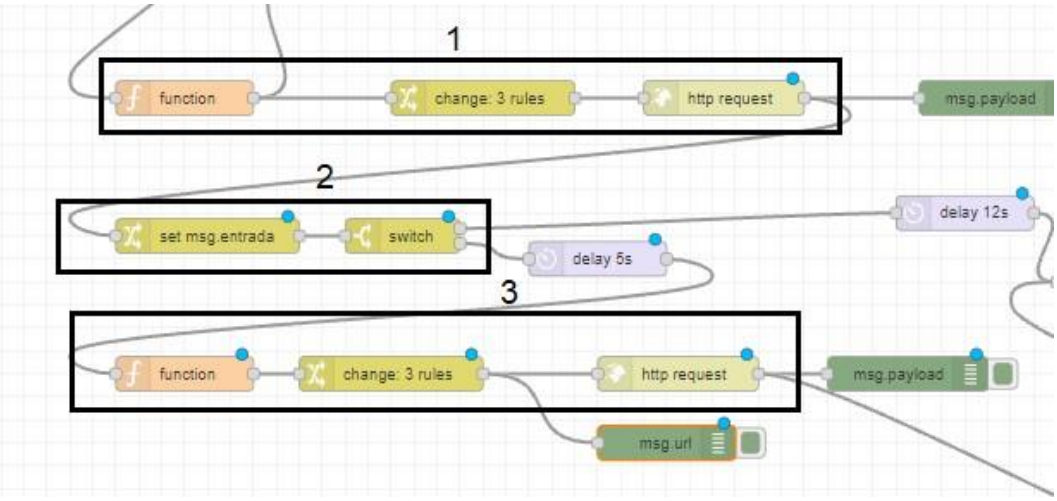
Para realizar este procedimiento previamente se debe llenar un formulario con los datos del nombre, apellido e identificativo, estos parámetros serán utilizados en las peticiones HTTP Request, al llenar esta información se realiza la captura del rostro de la persona.A continuación, se repite el proceso de **almacenamiento de imagen de rostros en la nube** que se indicó en la Figura 24.

Crear grupo de personas

Para realizar este proceso se debe de partir de la creación del identificativo a través del servicio Person Group Create, en el cual se puede almacenar un grupo de personas de acuerdo con la foto capturada como se indica en la siguiente figura.

Figura 38

Grupo de nodos para crear grupo de personas



En el grupo de nodos 1, se realiza la petición de PersonGroup-List, como se la realizo en el grupo de nodos 15 del Reconocimiento de imagen. En el grupo de nodos 2, se establece un condicional para establecer si el Identificativo constaba, dentro de la lista PersonGroup y se almacena el valor de la variable del identificativo.

En el grupo de nodos 3 se establece la creación de un PersonGroup si en el proceso anterior el condicional determino que el identificativo registrado por el cliente no existe, a continuación, se indica la función y la petición HTTP Request para general el PersonGroup como se indica en la siguiente Tabla.

Tabla 23

Contenido de la función y petición del grupo de nodos 3

	Variable	**Descripción**
Atributo	msg.payload.personGroupId	Contiene el nombre del identificativo registrado
	Mensaje	**Contenido**
	msg.headers	Content-Type: application/json

		"Ocp-Apim-Subscription-Key": "807a1ed2a37e47d58a7b5876e0dab7f0"
	Msg.payload	"name":"aniadir" "userData":"nueva familia" "recognitionModel": "recognition_03"

	Campo	Contenido
Peticion http	Method	PUT
	URL	https://southcentralus.api.cognitive.microsoft.com/face/v1.0/persongroups/msg.payload.personGroupId

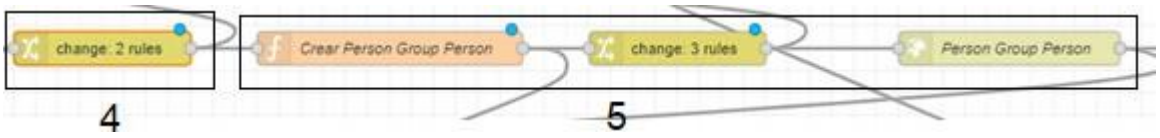
Nota: En la Tabla 23 se registra en el cuerpo del mensaje el identificativo y modelo de reconocimiento, además se debe incluir el nombre del identificativo.

- **Crear una persona y añadirla a un identificativo**

En este proceso se realiza la creación de una persona en la que se registra el nombre, el apellido y el identificativo el cual fue registrado o si se desea registrar a una persona en un identificativo ya existente, como se indica en la siguiente figura.

Figura 39

Grupo de nodos para crear una persona y añadirla a un identificativo



En el grupo de nodos 4 se recolecta la información del nombre y apellido de la persona, además del identificativo, mientras en el grupo de nodos 5 se realiza la

petición HTTP Request al servicio Create-Person Group Person como se indica en la siguiente Tabla.

Tabla 24

Contenido de los atributos, función y petición del grupo de grupo de nodos 5

	Variable	**Descripción**
Atributos	msg.name	Contiene el nombre registrado
	msg.userdata	Contiene el apellido registrado
	msg.payload.pid	Contiene el identificativo registrado
	Mensaje	**Contenido**
Funcion	msg.headers	Content-Type: application/json "Ocp-Apim-Subscription-Key": "807a1ed2a37e47d58a7b5876e0dab7f0"
	Msg.payload	"name": msg.name “userData”: msg.userData,
	Campo	**Contenido**
Petición http	Method	POST
	URL	https://southcentralus.api.cognitive.microsoft.com/face/v1.0/persongroups/msg.payload.pid/persons

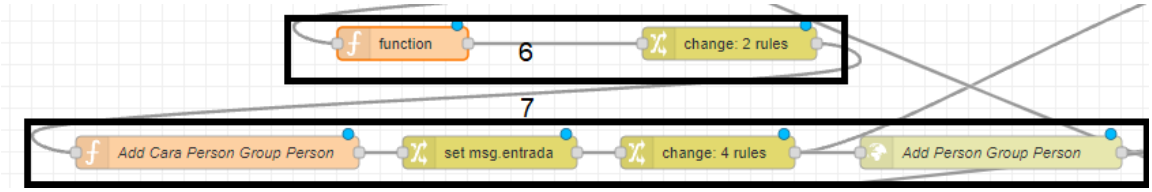
Nota: En la Tabla 24 el cuerpo del mensaje ingresa como parámetros los datos de la persona como son el nombre, apellido, identificativo, además de incluir el nombre del identificativo en el enlace de la URL.

- **Añadir el rostro a una Persona que pertenezca a un identificativo**

En este proceso se realiza la agregación de rostro a una persona que pertenezca a un identificativo en la que se registra el url de la imagen que se encuentra almacenada en el contenedor Azure Blob Storage como se indica en la Figura 24.

Figura 40

Grupo de nodos para añadir el rostro a una Persona que pertenezca a un identificativo



En el grupo de nodos 6 se registra el valor del PersonId del anterior proceso de Create-Person Group Person, para ser utilizado en el proceso de Añadir rostro en el grupo de nodos 7 en el cual se realiza una solicitud para agregar la imagen capturada a la persona creada, como se indica en la siguiente Tabla.

Tabla 25

Contenido de los atributos, función y petición del grupo de nodos 7

	Variable	**Descripción**
Atributos	msg.info	Contiene el url de la persona capturada por la camara
	msg.entrada	Contiene el identificativo registrado
	msg.personid	Contiene el valor del PersonId
	Mensaje	**Contenido**
Funcion	msg.headers	Content-Type: application/json "Ocp-Apim-Subscription-Key": "807a1ed2a37e47d58a7b5876e0dab7f0"

	msg.payload	“url”: msg.info

	Campo	Contenido
Peticion http	Method	POST
	URL	https://southcentralus.api.cognitive.microsoft.com/face/v1.0/persongroups/msg.entrada/persons/msg.personId/persistedFaces?detectionModel=detection_03

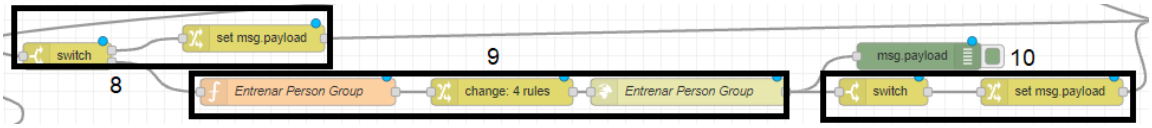
Nota: En la Tabla 25 el cuerpo del mensaje ingresa como parámetros la url de la imagen almacenada en la nube, además de registrar el nombre del identificativo y del PersonId obtenido del proceso de crear persona.

Entrenamiento del grupo de personas

Este paso permite entrenar al grupo con la nueva persona agregada a un identificativo, para que la actualización del identificativo pueda ser usada en el proceso de reconocimiento de personas como se indica en la siguiente figura.

Figura 41

Grupo de nodos para el entrenamiento del grupo de personas



En el grupo de nodos 8 se establece un condicional si el mensaje recibido es un error, para lo cual la imagen capturada no será validada, debido a que el servicio Add-Person Group Person no pudo realizar el reconocimiento de rostro, en el grupo de nodos 8 en el condicional al no detectar error, realiza el entrenamiento del PersonGroup en el grupo 9 como se indica en la siguiente Tabla.

Tabla 26

Contenido de los atributos, función y petición del grupo de grupo de nodos 9

	Variable	**Descripción del atributo**
Atributos	msg.entrain	Contiene el nombre del identificativo registrado
	MENSAJE	**CONTENIDO**
Función	msg.headers	Content-Type: application/json "Ocp-Apim-Subscription-Key": "807a1ed2a37e47d58a7b5876e0dab7f0"
	msg.payload	
	CAMPO	**CONTENIDO**
Petición http	Method	POST
	URL	https://southcentralus.api.cognitive.microsoft.com/face/v1.0/persongroups/msg.entrain/train

Nota: En la Tabla 26 el cuerpo del mensaje debe estar vacío, mientras en la Tabla 2 se registra el nombre del identificativo que se va a entrenar en el enlace de la URL de la petición.

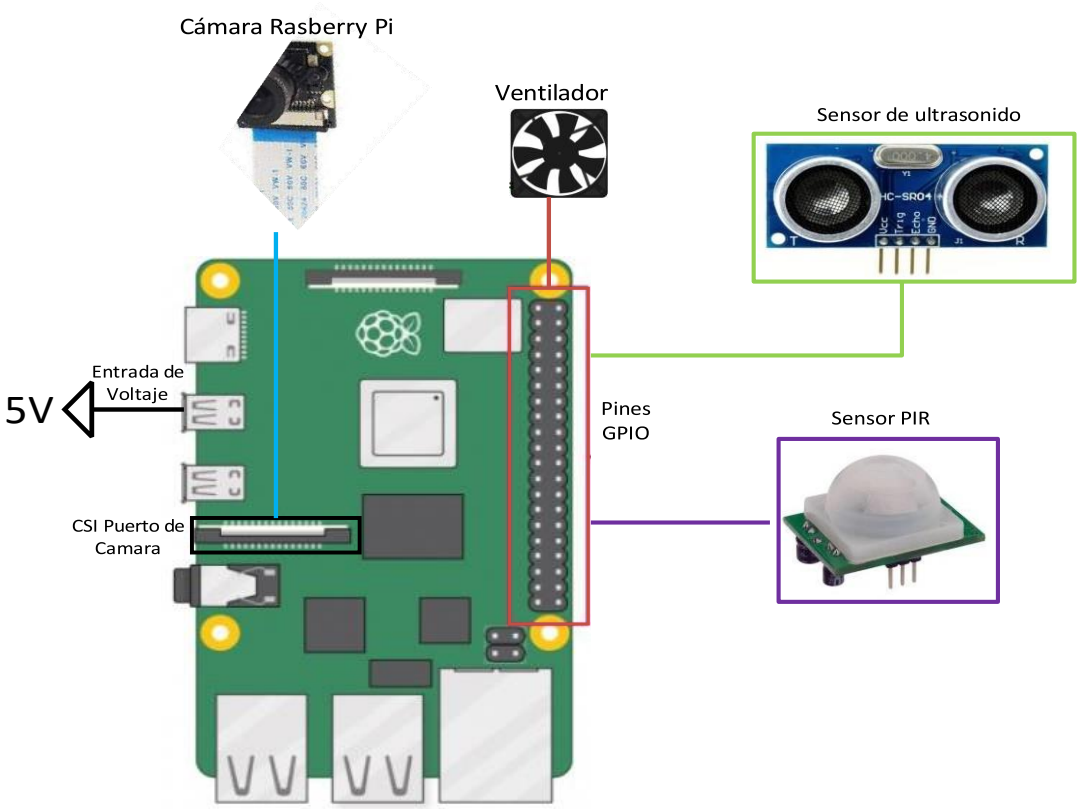
En el grupo de nodos 10 permitirá indicar al usuario si la agregación de la persona a un identificativo se realizó de forma exitosa, dependiendo de la buena captura del rostro de la persona.

Diseño del circuito electrónico

El diagrama de conexión de los elementos electrónicos como los sensores, el ventilador y la cámara están conectados directamente a la Rasberry PI como se indica en la siguiente figura

Figura 42

Diagrama de conexión del circuito electrónico



A continuación, se indica los pines de conexión entre los elementos electrónicos y los pines GPIO de la Rasberry Pi en la siguiente Tabla.

Tabla 27

Pines de conexión entre la Rasberry con los componentes electrónicos

Componente electrónico	PIN del dispositivo electrónico	PIN GPIO de la Rasberry PI
Ventilador	Ground	PIN 8.-Ground
	VCC	PIN 1.-3V3 power
Modulo sensor de ultrasonido	VCC	PIN 2.-5V power
	Echo	PIN 16.-GPIO 23
	Trigger	PIN 18.-GPIO 24
	Ground	PIN 14.-Ground
Modulo Sensor PIR	VCC	PIN 2.-5V power
	Output	PIN 22,.GPIO 25
	Ground	PIN 20.-Ground

De igual manera se indican los valores y características eléctricas de cada componente en la siguiente Tabla.

Tabla 28

Características de los componentes electrónicos

Componente electrónico	Características eléctricas	Características extra
Rasberry PI Model 3B+	Voltaje: 5V	SO: Raspbian
	Corriente: 2.5 A	Memoria:16 GB
Módulo de sensor ultrasónico HC SR04	Voltaje: 2 - 5 V	Rango: 1 a 10 m.
	Corriente: 2 – 5 mA	
Módulo de sensor PIR HC-SR501	Voltaje: 4.5 – 20 V	Tiempo de retardo: 5- 200 s
	Corriente: <60 uA	Rango: 3 a 7 m.

Rasberry PI Camera Module	Voltaje: 5V	Resolución: 5 MP.
IR 5MPX OV5647	Corriente: 3 – 5 mA	Video:1080p

Diseño gráfico de la aplicación web

El diseño del entorno grafico fue desarrollado a través de los nodos de dashboard, en la primera ventana se puede apreciar una portada y un formulario que realice la autenticación del cliente a través de un usuario y contraseña, como se indica en la siguiente figura

Figura 43

Ventana de portada al sistema de identificación de personas

En la segunda ventana presenta los dos procesos que se pueden realizar en el sistema, los cuales son el reconocimiento de personas y el de añadir una persona, como se indica en la siguiente figura.

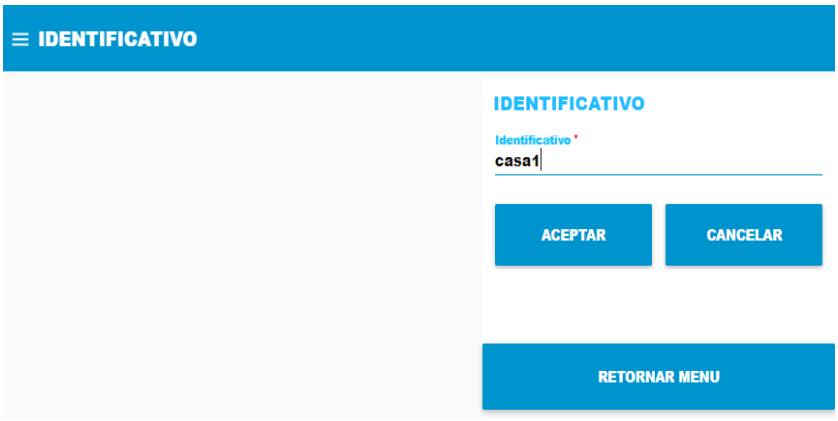
Figura 44

Ventana de menú

Para el caso de haber seleccionado la opción Reconocimiento, se abrirá una nueva ventana en la que solicitará el ingreso del identificativo el cual denota al grupo de personas al que pertenece el cliente como se indica en la siguiente figura.

Figura 45

Ventana del formulario de reconocimiento



En la siguiente ventana del proceso de reconocimiento se desplegará una pantalla en la que aparece la fotografía de la persona captada por los sensores y la información de la persona reconocida, como se indica en la siguiente figura.

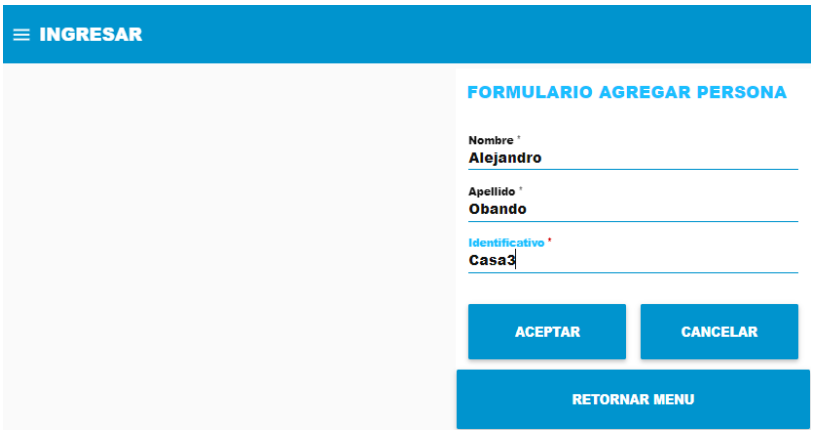
Figura 46

Ventana de información del proceso reconocimiento facial

En la ventana de añadir persona se presenta un formulario en el que pide registrar su nombre, apellido, e identificativo al cual pertenece o crear un nuevo grupo, al seleccionar aceptar se realizara la captura del rostro de la persona, como se indica en la siguiente figura.

Figura 47

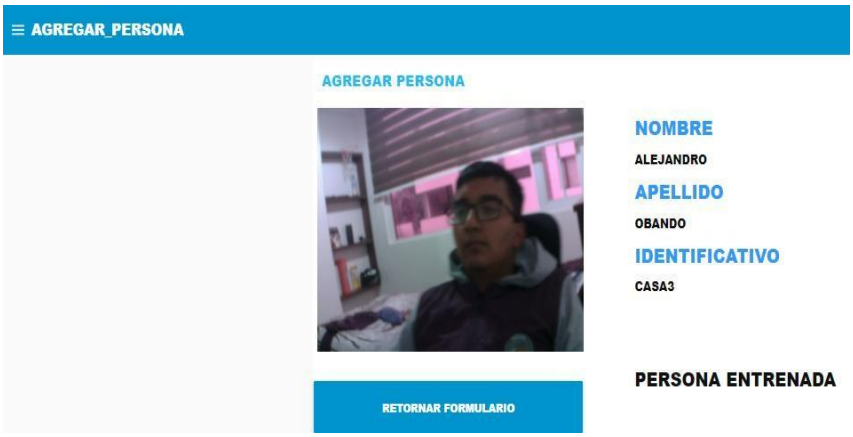
Ventana de Formulario para agregar personas



En la ventana de añadir siguiente se presenta una pantalla con el rostro de la persona que fue capturada al registrar sus datos y la información con la que se registró, como se indica en la siguiente figura.

Figura 48

Ventana de información del proceso agregar personas



Creación de túnel entre servidor local y servidor IoT

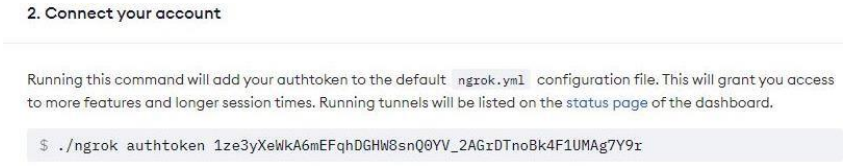
Para generar el dominio de acceso a internet se utilizó la herramienta de software de “ngrok”, que permite establecer un túnel y establecer la comunicación con el servidor local.

Para la instalación de ngrok en la Rasberry Pi, seguir los siguientes pasos:

- Ingresar a la página de ngrok “https://dashboard.ngrok.com/get-started/setup” y crear usuario y contraseña.
- Descargar el paquete ngrok para Linux (ARM) en la carpeta “home/pi”.
- Ubicar el archivo ngrok a la direccion “usr/bin”, ejecutar el comando **sudo mv /home/pi/ngrok /usr/bin**
- Activar los permisos de ejecución a través del comando **sudo chmod 755 ngrok**
- Copiar el código de authtoken de la página web de ngrok y pegar en la ruta de “usr/bin”, como se indica en la siguiente figura.

Figura 49

Código authtoken de la página de ngrok



- Para especificar la creación del túnel entre el servidor local hacia el internet, ejecutar el comando **ngrok http 80**
- Copiar la dirección http en el navegador para que su servidor local pueda ser desplegado en el internet, como se indica en la siguiente figura

Figura 50

Generación del túnel de la aplicación local hacia el internet

Creación del front end en la plataforma de servicio en la nube FRED

- Se debe crear una cuenta en la plataforma https://fred.sensetecnic.com/ en la cual se debe registra con un nombre y contraseña, como se indica en la siguiente figura

Figura 51

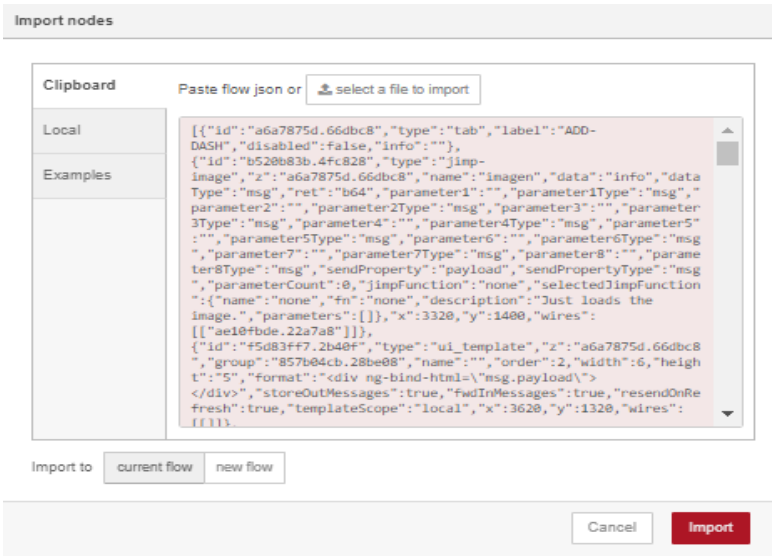
Registro de datos en la plataforma FRED

Nota: Tomado de (FRED, 2021).

- Seguido de eso se debe importar los códigos de ADD-DASH y REC-DASH en la plataforma de Node- RED, además de instalar los nodos: node-red-contrib-azure-blob-storage, node-red-contrib-image-tools, node-red-dashboard, node-red-node-pisrf, como se indica en la siguiente figura.

Figura 52

Ventana de importación de los flujos ADD-DASH y REC-DASH



- A continuación, en el NODE_RED del servidor local se debe instalar los nodos node-red-contrib-fred, en la plataforma FRED, para el caso del flujo REC-DASH se debe conectar un nodo que reciba los datos del url de la imagen de la persona reconocida, mientras en ADD-DASH se debe conectar un nodo para

enviar el identificativo y otro para recibir el url de la persona que se agrega. De manera viceversa se lo realiza en el servidor local de node-red, como se indica en las figuras siguientes.

Figura 53

Conexión de nodos REC-DASH con el nodo FRED

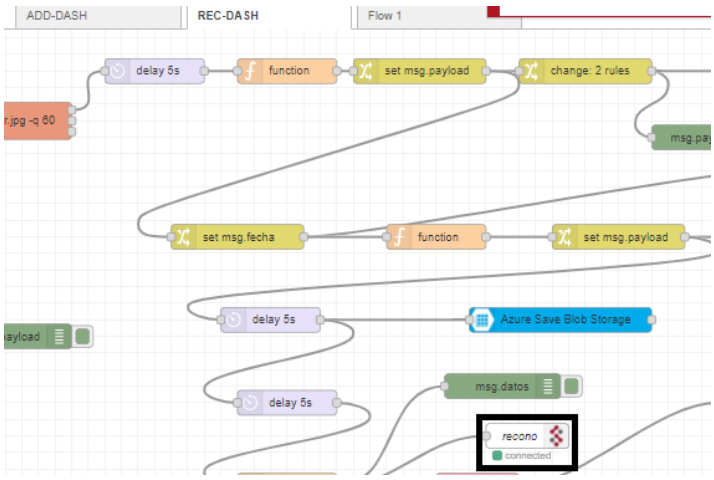
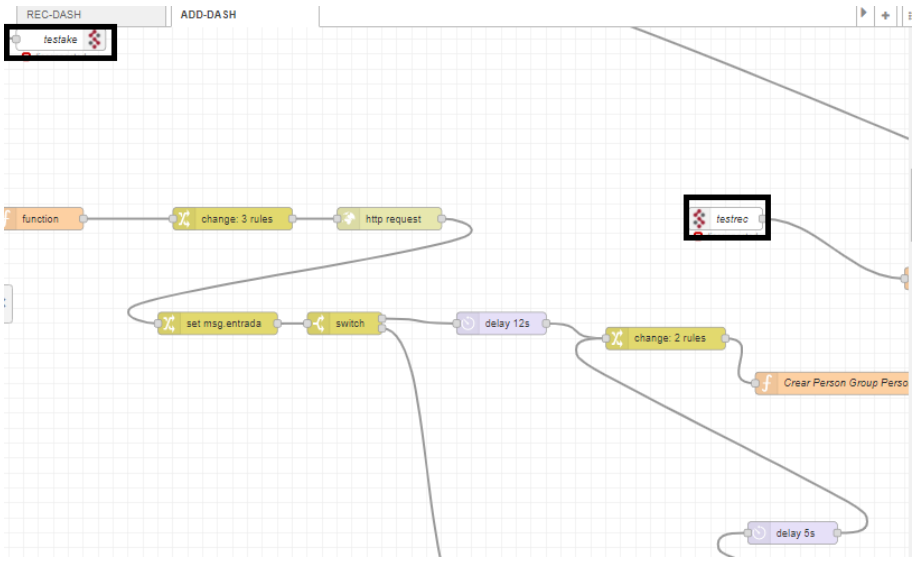


Figura 54

Conexión de nodos ADD-DASH con los nodos FRED



- Finalmente se debe presionar el botón Deploy y posterior Dashboard para utilizar el aplicativo web, como se indica en la siguiente figura.

Figura 55

Front-END del aplicativo web

Capitulo V

Pruebas de funcionamiento

Para la prueba de funcionamiento, el ambiente para el reconocimiento de las personas del hogar se lo realiza en los periodos del día y la noche para analizar la precisión del sistema, para ello el dispositivo se instalará en la puerta domiciliar.

Para verificar el funcionamiento el sistema se someterá a la siguiente simulación, para la cual cuatro personas del hogar serán registradas pertenecientes a un identificativo llamado casa 8, en el cual se compara la precisión del reconocimiento en el ambiente antes descrito, como se indica en la siguiente Tabla.

Tabla 29

Pruebas de funcionamiento de un mismo identificativo

Usuario	Dia		Noche	
Identificativo: casa8	**Precisión**	**Estado**	**Precisión**	**Estado**
U1	98.06%	Reconocido	94.12%	Reconocido
U2	92.65%	Reconocido	92.49%	Reconocido
U3	95.54%	Reconocido	84.16%	Reconocido
U4	97.59%	Reconocido	91.53%	Reconocido
Promedio	**95.96%**		**90.58%**	

De acuerdo con los resultados obtenidos del sistema se puede decir que existe que el valor de precisión aumenta en el periodo del dia en un 5% al periodo de la noche, sin embargo, los dos cumplen el reconocimiento, ya que superan al umbral especificado del 80% que permite reconocer o desconocer al rostro de la persona.

Para el segundo experimento se registrará a una persona como identificativo casa10, para comprobar el reconocimiento en dicho identificativo y probar el reconocimiento en el identificativo casa8, el mismo procedimiento se lo realizará para una persona de la identificativa casa 8 para verificar el reconocimiento en los identificativos casa 8 y casa10, como se indica en la siguiente Tabla.

Tabla 30

Pruebas de funcionamiento de un diferente identificativo

Usuario	Identificativo casa8		Identificativo casa10	
	Precisión	**Estado**	**Precisión**	**Estado**
U1-Casa 8	98.06%	Reconocido	0%	Desconocido
U1-Casa 10	59.27%	Desconocido	94.57%	Reconocido

De acuerdo con los resultados obtenidos, se comprueba que el sistema realizo con éxito el reconocimiento al usuario que pertenece a su identificativo, mientras que, si un usuario que no pertenece a un identificativo no reconocerá a la persona, denotándolo como desconocido según el umbral especificado del 80%

Pruebas de carga

Para realizar este procedimiento se utiliza el programa de software libre Gatling mediante el cual se analiza el funcionamiento de un sistema web en un entorno controlado mediante un numero especifico de peticiones enviadas de forma simultánea, que permite que un aplicativo web trabaje sin inconvenientes bajo un nivel de tráfico, para la simulación el sistema será estresado con 6 distintas cantidades de usuarios, los cuales son 100,300 400,450 y 500. Las cuales se presentan a continuación.

Figura 56

Prueba de Carga con 100 usuarios

Nota: Figura obtenida de la carta de datos generada por Gatling.

Como se observa en la figura anterior, existe un error del 47%, lo cual indica que el 53% de las peticiones realizadas al servidor tuvieron éxito de las 100 muestras, con una media de respuesta de 1915 ms.

Figura 57

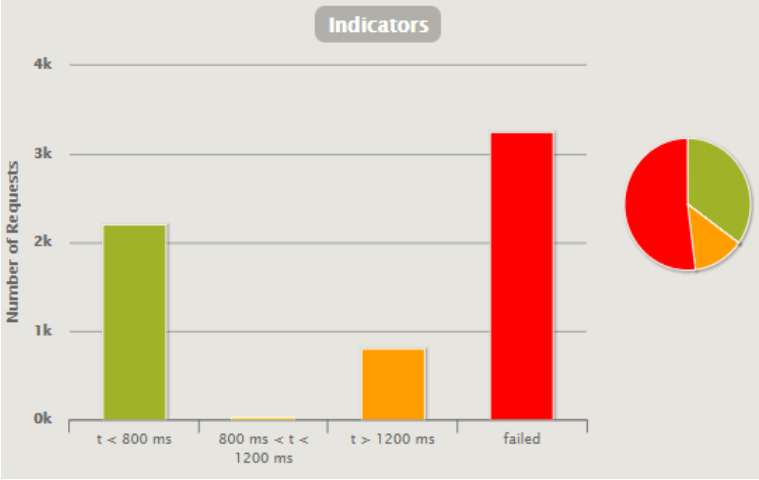
Prueba de carga con 300 usuarios

Nota: Figura obtenida de la carta de datos generada por Gatling.

Como se observa en la figura anterior, existe un error del 51%, lo cual indica que el 49% de las peticiones realizadas al servidor tuvieron éxito de las 300 muestras, con una media de respuesta de 2451 ms.

Figura 58

Prueba de carga con 500 usuarios

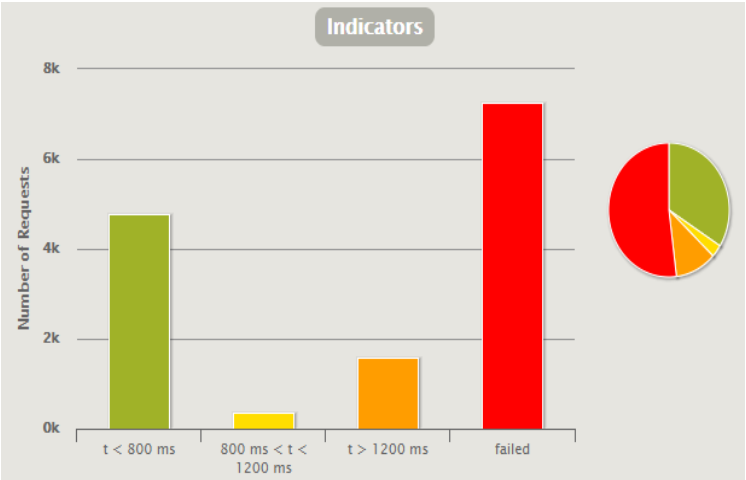


Nota: Figura obtenida de la carta de datos generada por Gatling.

Como se observa en la figura anterior, existe un error del 53%, lo cual indica que el 47% de las peticiones realizadas al servidor tuvieron éxito de las 500 muestras, con una media de respuesta de 3348 ms.

Figura 59

Prueba de carga con 1000 usuarios

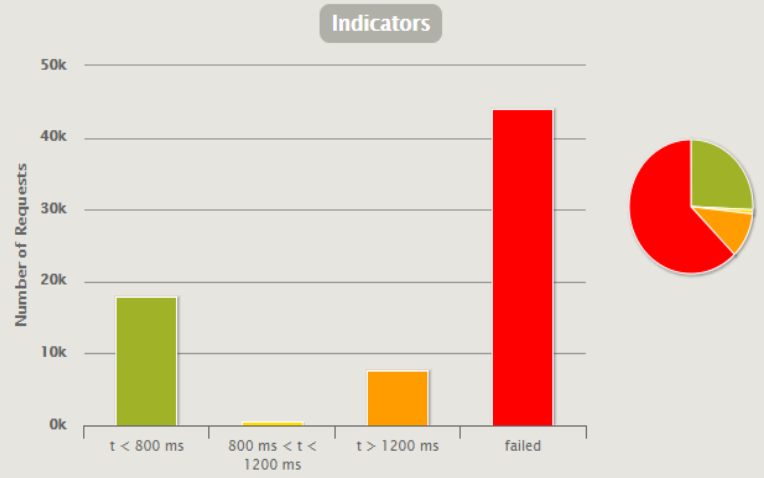


Nota: Figura obtenida de la carta de datos generada por Gatling.

Como se observa en la figura anterior, existe un error del 53%, lo cual indica que el 47% de las peticiones realizadas al servidor tuvieron éxito de las 1000 muestras, con una media de respuesta de 4758 ms.

Figura 60

Prueba de carga con 5000 usuarios

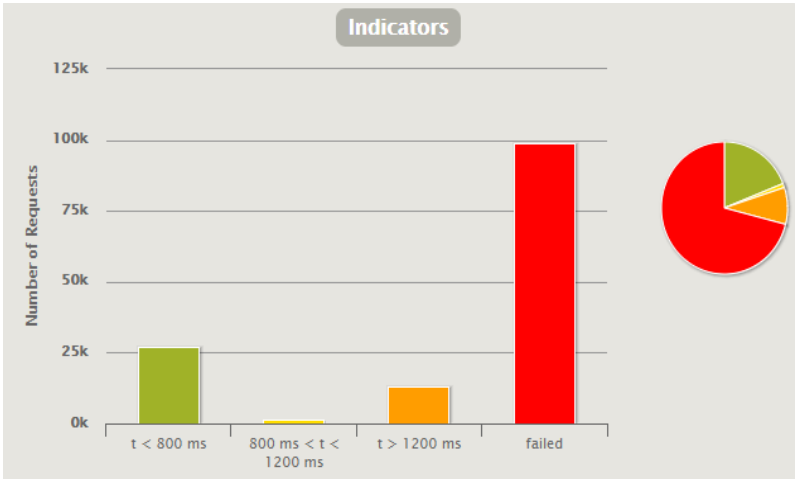


Nota: Figura obtenida de la carta de datos generada por Gatling.

Como se observa en la figura anterior, existe un error del 63%, lo cual indica que el 37% de las peticiones realizadas al servidor tuvieron éxito de las 5000 muestras, con una media de respuesta de 8088 ms.

Figura 61

Prueba de carga con 10000 usuarios



Nota: Figura obtenida de la carta de datos generada por Gatling.

Como se observa en la figura anterior, existe un error del 71%, lo cual indica que el 29% de las peticiones realizadas al servidor tuvieron éxito de las 10000 muestras, con una media de respuesta de 12088 ms.

En la siguiente tabla se presenta los resultados de las pruebas realizadas con respecto a los usuarios con el que fue estresado el sistema, el error y tiempo medio de respuesta, en la cual se puede observar que, al aumentar el número de usuarios, el error y la media del tiempo de respuesta aumentan.

Tabla 31

Resumen de las pruebas de carga

Número de usuarios	**Media de tiempo de respuesta (ms)**	**%Error**
100	1915	47
300	2451	51
500	3488	53
1000	4758	53
5000	8088	63
10000	12088	71

Pruebas de usabilidad

Este tipo de prueba tiene como propósito de cuantificar la experiencia de interacción del usuario hacia un dispositivo. Para la elaboración de la prueba de usabilidad se utilizará el test de SUS, el cual a través de una encuesta de 10 preguntas con 5 diferentes valoraciones de respuesta de acuerdo a la Escala de (Likert) (1 a 5), las cuales van desde totalmente de acuerdo (5), hasta totalmente desacuerdo (1).

Para el desarrollo de la prueba de usabilidad se lo realizará con 10 personas, a través del cual se conocerá la valoración de la aplicación desarrollada. Este valor de SUS deberá ser expresado en forma de porcentaje. A continuación, se presentan los resultados de las preguntas impares y pares de las pruebas de usabilidad en la siguiente figura.

Figura 62

Resultados de las preguntas impares de las pruebas de usabilidad

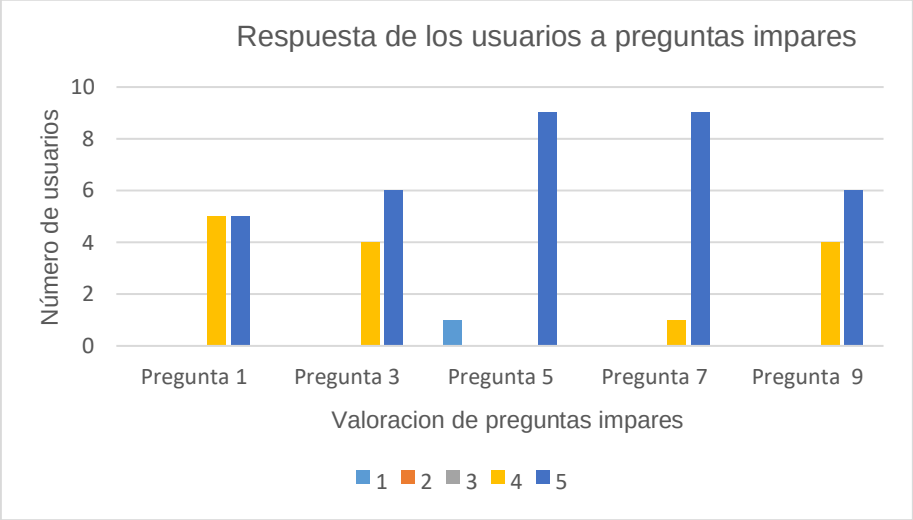
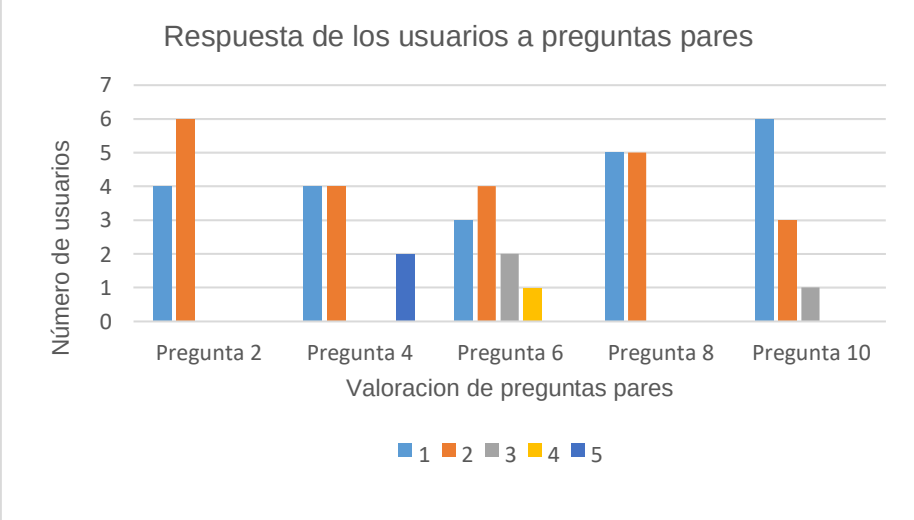


Figura 63

Resultados de las preguntas pares de las pruebas de usabilidad



Obtenida las respuestas, para obtener el valor de usabilidad del sistema se debe sumar las respuestas de las preguntas impares y restar de ese valor 5, mientras que para las preguntas pares se debe de igual manera sumar estos valores y de 25 restar el valor obtenido. El resultado obtenido para ambos casos se debe sumar y multiplicar por 2.5. Cabe mencionar que si el sistema obtiene un valor de 100% el sistema es perfecto, si el sistema se obtuvo el valor de 70% es considerado bueno y un valor menor a 50% será muy ineficiente, como se indica en la siguiente Tabla.

Tabla 32

Promedio y resultados de la prueba de usabilidad

Usuarios	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	Puntaje SUS
1	4	2	5	2	5	1	5	1	4	1	90
2	5	1	5	1	5	1	5	1	5	1	100
3	5	2	4	2	5	3	5	2	5	1	85
4	4	2	4	1	5	2	5	2	4	1	85
5	5	1	5	5	5	4	5	1	5	1	82.5
6	5	1	5	5	5	2	5	1	5	1	87.5
7	4	1	3	1	1	2	4	2	4	3	67.5
8	4	2	5	1	5	1	5	1	5	2	92.5
9	5	2	4	2	5	2	5	2	5	2	85
10	4	2	5	2	5	3	5	2	4	2	80
Promedio:											85.5

Nota: En esta tabla se muestra el puntaje de cada pregunta y el promedio de usabilidad.

En conclusión, se obtuvo el valor de 85.5% que nos indica que la aplicación es muy favorable para los usuarios, cabe recalcar que se podría mejorar el porcentaje si es que se instruye y se guie al cliente en la aplicación web y dispositivo ya que en la encuesta se refleja que el cliente percibe inconsistencias y que necesita apoyo técnico, lo cual se observa en las preguntas pares 4 y 6 de la figura 63.

Capítulo VI

Conclusiones y Recomendaciones

Conclusiones

Se desarrollo un sistema de identificación de personas basado en servicios de reconocimiento facial en la nube, mediante la utilización de los servicios de Face Api y Azure Blob Storage los cuales fueron integrados con el dispositivo físico a través del framework Node-RED.

Se investigo los fundamentos teóricos de la nube de Azure en la cual se exploró el funcionamiento y la utilización de las API Face y Blob Storage que permitieron el desarrollo del proyecto de investigación.

Se implemento dispositivo que permite la captura de rostros a través de una cámara y sensores de movimiento y ultrasonido que permiten el reconocimiento de personas, los cuales están conectados al microcomputador Rasberry Pi.

Se implemento al aplicativo web la opción agregar personas la cual permitió entrenar al sistema con un identificativo de un grupo de personas y la opción reconocimiento la cual a través del identificativo que se quiera poner en marcha el sistema para detectar personas e indicar el grado de coincidencia y el reconocimiento de la persona.

Se implemento el backend del sistema a través de la integración de los servicios en la nube con los dispositivos físicos a través del framework node-red,mientras en frontend del sistema se despliega a través del dashboard de la plataforma servicio en la nube FRED.

Mediante las pruebas de funcionamiento se evaluó que el sistema funciona correctamente bien en los periodos del día y en la noche en los cuales presentaron un valor promedio de grado de confiabilidad de 95.96% y 90.58% respectivamente.

Mediante las pruebas de carga, se obtuvo que el sistema funciona con el 47% de error y 1,9 s de respuesta para 100 usuarios y 71% de error y con 12,1 s de respuesta para 10000 usuarios, lo cual no presentara problemas debido a que el sistema en ambiente no será sometido a una gran cantidad de tráfico debido a que los usuarios que usaran el servicio en forma simultanean no es tan alta.

Al realizar las pruebas de usabilidad, el sistema obtuvo el 85,5% lo cual es considerada una buena valoración para implementar el sistema de reconocimiento, sin embargo, hay tomar en cuenta el tipo de usuarios y la aplicación en la cual se va a implementar.

Recomendaciones

Instalar las librerías de los nodos de Fred, dashboard, azure blob storage, rasberry pi que permiten interactuar con los servicios de reconocimiento y para almacenamiento de imágenes de las personas registradas.

Al seleccionar la nube se recomienda investigar sobre los servicios en la nube que presenten APIs de reconocimiento en cuanto a la confiabilidad del servicio de reconocimiento y almacenamiento en base a su rendimiento, latencia para el desarrollo e implementación del dispositivo.

Instalar el dispositivo a la altura del rostro de la persona en un lugar central en sitios de concurrencia o de ingreso de personas y cuente con cobertura de internet, ya que esto afectara al rendimiento y latencia del sistema de reconocimiento facial.

Se recomienda investigar y conocer sobre los métodos HTTP para realizar peticiones a los servicios de reconocimiento facial y almacenamiento de imágenes de en la nube de Azure.

Trabajos Futuros

Para los trabajos futuros se propone:

- Implementar el reconocimiento facial a través de algoritmos de detección, que funcione a la par del servicio de reconocimiento facial en la nube, el cual funcione en las situaciones que no se disponga de internet.
- Agregar al sistema de reconocimiento facial otros sensores que ayuden a precisar la detección u otro sistema de identificación como biométricos, RFID entre otros que permite incrementar la seguridad y confiabilidad de los sistemas, además de otros sistemas que facilite el registro de personas a través de asistentes virtuales.
- Complementar al sistema con envió de mensajes automáticos que permitan alertar y notificar al usuario al momento que una persona desconocida sea capturada por el sistema, además de complementarlos con sistemas de control mediante sensores magnéticos, de contacto y actuadores hidráulicos, neumáticos aplicados en cerraduras o automatización de puertas.
- Realizar investigaciones sobre los parámetros de calidad de servicio entre los servicios de reconocimiento facial en la nube y los algoritmos de detección y reconocimiento, que permitan determinar líneas de acción y aplicaciones.
- Implementar mecanismos de seguridad a través de certificados de autenticidad SSL y encriptación de datos en el aplicativo web para mejorar la privacidad al usuario.

Acrónimos

- HTTP. Hypertext Transfer Protocol.
- IoT. Intenet of Things.
- MQTT. Messege Queuing Telemetry Transport (Transporte de telemetria).
- REST. Representational state transfer (Transferencia de estado representacional).
- SLR. Revision Sistemática de la Literatura.
- SMS. Mapeo Sistemático de la Literatura.
- SOA. Arquitectura Orientada a Servicios.
- API. Application Programming Interfaces (Interfaz de Programación de aplicaciones).
- SSL. Secure Sockets Layer (Seguridad capa de transporte).
- TICS. Tecnologías de la Información y Comunicación.
- PIR. Passive Infrared.
- QoS. Quality of Service (Calidad de Servicio).
- URIs. Uniform Resource Identifier (Identificador de recursos uniforme).
- URL. Uniform Resource Locater (Localizador de recursos uniforme).
- JSON. JavaScript Object Notation (Notacion de Objeto Java Script).
- VPN Virtual Private Network (Red Privada Virtual).

Referencias

Nuñez, D. (2013). *Implementacion de un prototipo de software como servicio (SAAS) para pequeñas y medianas empresas.* Quito: Escuela Politecnica Nacional.

Alshamsi, H., Kepuska, V., & Meng, H. (2018). Automated Facial Expression and Speech Emotion Recognition App Development on Smart Phones using Cloud Computing. *IEEE*, 730-738. doi:10.1109/IEMCON.2018.8614831.

Avelectronics. (s.f.). *Modulo cámara Raspberry Pi visión nocturna*. Obtenido de https://avelectronics.cc/producto/raspberry-pi-camera-module-vision-nocturna/

AWS. (s.f.). *Amazon Rekognition.* Obtenido de https://aws.amazon.com/es/rekognition/?blog-cards.sort-by=item.additionalFields.createdDate&blog-cards.sort-order=desc

Ayad, M., Taher, M., & Salem, A. (2014). Real-Time Mobile Cloud Computing: A Case Study in Face Recognition. *IEEE*, 73-78. doi:10.1109 / WAINA.2014.22.

Belhouchette, K. (2021). Facial Recognition to Identify Emotions: An Application of Deep Learning. *Springer Link*. doi:https://doi.org/10.1007/978-3-030-70713-2_46

Bharadwaj, G., Saini, S., Chauhan, A., & Kumar, P. (2021). Automated Surveillance Security System Using Facial Recognition for Homes and Offices. *Springer Link*. doi:https://doi.org/10.1007/978-981-15-9509-7_13

Caceres, E. (2018). *Aplicacion movil de reconocimiento facial en personas con antecedentes de abuso sexual en la provincia de Andahuaylas.* Andahuaylas: Universidad Nacional Jose Maria Arguedas.

Che, S., Kamphuis, P., & Kim, J. (2021). A Comparative Analysis of Attention to Facial Recognition Payment Between China and South Korea: A News Analysis Using Latent Dirichlet Allocation. doi:https://doi.org/10.1007/978-3-030-78642-7_11

Diaz,J.G. (2012). *Tecnicas de biometria basadas en patrones faciales del ser humano.* Pereira: Universidad Tecnologica de Pereira. Obtenido de http://repositorio.utp.edu.co/dspace/bitstream/11059/2738/1/0053682L864.pdf

Diosdado, R. (2018). *Sensor de ultrasonidos Hc-Sr04*. Obtenido de https://www.zonamaker.com/arduino/modulos-sensores-y-shields/ultrasonido-hc-sr04

Domingo, M. (2003). Detección mediante el reconocimiento de imagenes. *Universidad Santiago de Chile*.

Duque Quezada, R., & Sanchez Vaca, J. (2014). *Implementacion de servicios computacionales flexibles en la nube para el Area de Investigacion del Departamento de la Computacion (DECC).* Departamento de Ciencias de la Computacion(DECC).

El Comercio. (30 de Octubre de 2019). *China usará reconocimiento facial para reforzar la seguridad en el metro*, pág. 1.

Fielding, R. (2000). Architectural Styles and the Design of Network-based Software Architectures. *[PhD thesis].* University of California, California.

Fowler, M., & Lewis, J. (2014). *Microservices*. (MartinFowler.com) Obtenido de https://martinfowler.com/articles/microservices.html

FRED-Sensetenic. (2021). *FRED Documentacion*. Obtenido de http://docs.sensetecnic.com/fred/instance/

Gomez Rodriguez, e. a. (2007). Computadoras de placa reducida Raspberry Pi 3 yAsus Tinker Board. *Computadoras de placa reducida Raspberry Pi 3 yAsus Tinker Board.* Universidad de Guadalajara, Mexico, Guadalajara. Obtenido de https://www.reci.org.mx/index.php/reci/article/view/86/384

Gomez, M. (2011). *Seguridad en el cloud computing.* INTECO-CERT.

Gonzalez Anton, J. (2015). *Medición de Distancia y Velocidad empleando de sensor de ultrasonidos.* Unicersidad de Valladolid, Valladolid. Obtenido de https://core.ac.uk/download/pdf/211100237.pdf

Google Cloud. (2021). *Ventajas de Google Cloud*. Obtenido de https://cloud.google.com/why-google-cloud?authuser=2

Grabovskyi, V., & Martynovych , O. (2019). FACIAL RECOGNITION WITH USING OF THE MICROSOFT FACE. *International Scientific and Practical Conference "Electronics and Information Technologies"*, 30-38. doi:https://doi.org/10.30970/eli.12.3

Iberobotics. (2021). *Sensor de movimiento PIR detector de presencia HC-SR501*. Obtenido de https://www.iberobotics.com/producto/sensor-de-movimiento-pir-detector-de-presencia-hc-sr501/

ITU Corporation. (2015). *Intenet of Things Global Standards Initiative.* Obtenido de http://www.itu.int/en/ITU-T/gsi/iot/Pages/default.aspx

Lee, H., Lee, Y., Hyuck, H., & Kang, S. (2021). iEdge: An IoT-assisted Edge Computing Framework. *IEEE*.

Libby, C., & Ehrenfeld, J. (2021). Facial Recognition Technology in 2021: the Future of Healthcare. *Springer*. doi:https://doi.org/10.1007/s10916-021-01723-w

Licencias Online. (2013). *Modelos de Servicio CLoud*. Recuperado el 9 de Agosto de 2021, de http://www.licenciasonline.com/ec/es/cloud/modelos-de-servicio

Lopez, J. (2018). Sistema de reservas web para el servicio de transporte puerta a puerta aplicando. *[Tesis de Pregrado].* Universidad Tecnica de Ambato, Ambato.

Masruroh, S., Fiade, A., & Julia, I. (2018). NFC Based Mobile Attendence System with Facial Authorization on Raspberry Pi and Cloud Server. *IEEE*, 1-6. doi:10.1109/CITSM.2018.8674293.

Mehedi , M., Ghulam, M., Hesham, A., Cheikhrouhou, O., Ibrahim, S., & M, S. (2020). Deep learning-based intelligent face recognition in IoT-cloud environment,

Computer Communications. *IEEE*, 215-222. doi:https://doi.org/10.1016/j.comcom.2020.01.050

Microsoft Azure . (2021). *Que son los blobs*. Obtenido de https://docs.microsoft.com/es-es/learn/modules/store-app-data-with-azure-blob-storage/2-what-are-blobs

Microsoft Azure. (2017). *Ques es Iaas*. Recuperado el 21 de Enero de 2020, de https://azure.microsoft.com/es-es/overview/what-is-paas/

Microsoft Azure. (2019). *Conceptos del reconocimiento facial FACE*. Obtenido de https://docs.microsoft.com/es-es/azure/cognitive-services/face/concepts/face-recognition

Microsoft Azure. (2019). *La guia de Azure para desarroladores.* Washington: Microsoft Press.

Microsoft Azure. (2021). *Que es Azure*. Obtenido de https://azure.microsoft.com/es-es/overview/why-azure/

Microsoft Azure. (2021). *Que son los blobs*. Obtenido de https://docs.microsoft.com/es-es/learn/modules/store-app-data-with-azure-blob-storage/2-what-are-blobs

Mittal, S., & Singh, V. (2020). Gender and Age based Census System for Metropolitan Cities. *IEEE*, 10.1109/ICRITO48877.2020.9198030.

Navarro, R. (2007). Modelado, Diseño e Implementación de Servicios. *[Tesis de Grado].* Universidad Politenica de Valencia, Valencia.

Nazri, T., Gaafar, T., & Sajak, H. (2020). IoT Parking Apps with Car Plate Recognition for Smart City using Node Red. *IEEE*. doi:10.1109/ICICS49469.2020.239511

Nemmaoui, S., & Elhammani, S. (2021). A New Approach Based on Steganography to Face Facial Recognition Vulnerabilities Against Fake Identities. *Springer*. doi:https://doi.org/10.1007/978-3-030-76508-8_19

ngrok. (2021). *What is ngrok*. Recuperado el 18 de Octubre de 2021, de What is ngrok: https://ngrok.com/product

Node-RED. (2020). *Introduction*. Obtenido de https://nodered.org/

Node-RED. (2021). *Node-RED Library*. Obtenido de https://flows.nodered.org/

Relica, Y. (2014). *Analisis y evaluacion de parametros de calidad en servicios cloud computing en el Ecuador.* Sangolqui: Departamento de Electrica y Electronica.

REQUEST FOR COMMENTS (RFC2764). (2018). *Vritual Private Network*.

REQUEST FOR COMMENTS (RFC3193). (2018). *VPN*. Obtenido de https://tools.ietf.org/html/rfc3193

Rodriguez , H. (2018). *Estuidio de herramientas basadas en IA Cloud y su aplicacion en el desarrollo de las actividades academicas de la carrera de Telamtica de la Universidad de Guayaquil.* Guayaquil: Universidad de Guayaquil.

Rodriguez, J. (2013). Circuito cerrado de televicion y seguridad electronica. Ecuador: Paraninfo.

Roundtree, A. (2021). Correction to: Testing Facial Recognition Software for Young Adults and Adolescents: An Integrative Review. In: Moallem A. (eds) HCI for Cybersecurity, Privacy and Trust. *Springer*, 2-31. doi:https://doi.org/10.1007/978-3-030-77392-2_31

Rui, J., & Danpeng, S. (2015). rchitecture Design of the Internet of Things based on Cloud Computing. *IEEE*, 1-4.

Ruiz, A. (2019). *Migracion de servidores a la nube de Microsoft Azure para mejorar la continuidad de los servicios de TI.* Lima: Universidad de San Ignasio de Loyola.

Sabharwal, T., & Gupta, R. (2021). Facial marks for enhancing facial recognition after plastic surgery. *Springer Link*, 391-396. doi:https://doi.org/10.1007/s41870-020-00566-x

Sabr, O., Kanakis, T., & Belton, J. (2019). Identifying and Tracking Individuals in a Smart Indoor Environment. *iIEEE*. doi:10.1109/IEC47844.2019.8950637

Saeed, I., Baras, S., & Hajjdiab, H. (2019). Security and Privacy of AWS S3 and Azure Blob Storage Services. *IEEE*, 388-394. doi:10.1109/CCOMS.2019.8821735

Santamaria, L. (2018). Inteligencia artificial en la nube, la última tendencia. Obtenido de https://www.e-volucion.es/2018/04/inteligencia-artificial-nube-ultima-tendencia

Santos, W., & Serrano, J. (2017). Desarrollo de una api rest con sus aplicaciones weby móvil para la venta de ropa online de la Empresa Roosman. *[Tesis de Grado].* Universidad Central del Ecuador, Quito.

Serbanati, A., Medaglia, C., & Ceipidor, U. (2011). Building Blocks of the Internet of Things: State of the Art and Beyond. (INTECH, Ed.) doi:10.5772/963, ISBN 978-953-307-380-4

Servicio integrado de seguridad ECU 911. (Diciembre de 2019). *Rendicion de cuentas 2019*. Obtenido de https://www.ecu911.gob.ec/rendicion-de-cuentas-2019/

Smith, M., & Miller, S. (2021). Correction to: The etical application of biometric facial recognition technology. *AI & Soc*. doi:https://doi.org/10.1007/s00146-021-01236-7

Srirama, S., Paniagua, C., & Flores, H. (2016). Social group formation with mobile cloud services. *SOCA*, 351-362. doi:https://doi.org/10.1007/s11761-012-0111-5

Subrahmanya, M., Srinivas, Y., & Rojee, J. (2017). Insider Threat detection with Face Recognition. *IEEE*.

Suntaxi Cantuña, O. (2019). *DISEÑO DE UN SISTEMA DE VIDEOVIGILANCIA BASADO EN RASPBERRY PI.* INSTITUTO TECNOLÓGICO SUPERIOR, Quito, Ecuador. Obtenido de http://201.159.223.6/bitstream/123456789/91/1/44.0260-SUNTAXI-CANTU%C3%91A-OSCAR-GEOVANNY.pdf

Unión Internacional de Telecomunicaciones. (agosto de 13 de 2014). *Tecnologia de la Infromacion-Computacion en la nube- Descripción general y vocabulario*. Obtenido de https://www.itu.int/es/Pages/default.aspx.

Verma, R., Sing, P., & Panigrahi, C. (2021). ISS: Intelligent Security System Using Facial Recognition. *Springer Link*. doi:https://doi.org/10.1007/978-981-15-6584-7_10

Villamizar, M., Garces, O., Castro, H., & Verano, M. (2015). Evaluating the monolithic and the microservice architecture pattern to deploy web. Computing Colombian Conference. *IEEE*, 583-590.

W3C. (2007). *SOAP Version*. Obtenido de https://www.w3.org/TR/soap12-part1/

Xue, Y., Zhang, H., & Ma, H. (1018). Performance Evaluation of Image and Video Cloud Services. *IEEE*. doi:https://doi.org/10.1109/HPCC/SmartCity/DSS.2018.00126

Yang T, Zhang Y, Sun J, & et al. (2021). Privacy Enhanced Cloud-Based Facial Recognition. *Springer Link*. doi:https://doi.org/10.1007/s11063-021-10477-y

Yi, S., Jing, X., Zhu, J., & Cheng, H. (2012). The Model of Face Recognition in Video Surveillance Based on Cloud Computing. In: Jin D., Lin S. (eds) Advances in Computer Science and Information Engineering. Advances in Intelligent and Soft Computing. *Springer*, 105-111. doi:10.1007/978-3-642-30126-1_18

Yunqui, Y., Liangliang, X., & Bastani, F. (2015). Leveraging Service Clouds for Power and QoS Management for Mobile Devices. *IEEE*. doi:10.1109/CLOUD.2011.55

ANEXOS