



**ESCUELA SUPERIOR POLITÉCNICA AGROPECUARIA DE MANABÍ
MANUEL FÉLIX LÓPEZ**

CARRERA DE COMPUTACIÓN

**INFORME DE TRABAJO DE INTEGRACIÓN CURRICULAR PREVIO A
LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN CIENCIAS DE LA
COMPUTACIÓN**

**MECANISMO: SISTEMATIZACIÓN DE EXPERIENCIAS PRÁCTICAS DE
INVESTIGACIÓN Y/O INTERVENCIÓN**

TEMA:

**SISTEMA BIOMÉTRICO PARA EL CONTROL DE ACCESO A LA
UDIV DE DESARROLLO COMPUTACIONAL DE LA ESPAM-MFL
EMPLEANDO TÉCNICAS DE DEEP LEARNING**

AUTORES:

**LELIS JAVIER MUÑOZ RIOS
JULIO JOSUE PONCE ALMEIDA**

TUTOR:

VICTOR JOEL PINARGOTE BRAVO, MGTR.

CALCETA, FEBRERO DE 2024

DECLARACIÓN DE AUTORÍA

Nosotros, Lelis Javier Muñoz Ríos y Julio Josué Ponce Almeida, con cédulas de ciudadanía 131349320–5 y 135078488–8 respectivamente, declaramos bajo juramento que el Trabajo de Integración Curricular titulado: “Sistema Biométrico para el Control de Acceso a la UDIV de Desarrollo Computacional de la ESPAM–MFL empleando Técnicas de Deep Learning” es de nuestra autoría, que no ha sido previamente presentado para ningún grado o calificación profesional, y que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración, concedemos a favor de la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López una licencia gratuita, intransferible y no exclusiva para el uso no comercial de la obra, con fines estrictamente académicos, conservando a nuestro favor todos los derechos patrimoniales de autor sobre la obra, en conformidad con el Artículo 114 del Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación.



LELIS JAVIER MUÑOZ RÍOS

CC: 131349320–5



JULIO JOSUÉ PONCE ALMEIDA

CC: 135078488–8

AUTORIZACIÓN DE PUBLICACIÓN

Lelis Javier Muñoz Ríos y Julio Josué Ponce Almeida, con cédulas de ciudadanía 131349320–5 y 135078488–8 respectivamente, autorizamos a la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López, la publicación en la biblioteca del Trabajo de Integración Curricular titulado: “Sistema Biométrico para el Control de Acceso a la UDIV de Desarrollo Computacional de la ESPAM–MFL empleando Técnicas de Deep Learning”, cuyo contenido, ideas y criterios son de nuestra exclusiva responsabilidad y total autoría.



LELIS JAVIER MUÑOZ RÍOS
CC: 131349320–5



JULIO JOSUÉ PONCE ALMEIDA
CC: 135078488–8

CERTIFICACIÓN DEL TUTOR

Mgtr. Víctor Joel Pinargote Bravo, certifica haber tutelado el Trabajo de Integración Curricular Titulado: “Sistema Biométrico para el Control de Acceso a la UDIV de Desarrollo Computacional de la ESPAM–MFL empleando Técnicas de Deep Learning”, que ha sido desarrollado por Lelis Javier Muñoz Ríos y Julio Josué Ponce Almeida, previo a la obtención del título de “Ingeniero en Ciencias de la Computación”, de acuerdo al **REGLAMENTO DE LA UNIDAD DE INTEGRACIÓN CURRICULAR DE CARRERAS DE GRADO** de la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López.

MGTR. VÍCTOR JOEL PINARGOTE BRAVO
CC: 131086793–0
TUTOR

APROBACIÓN DEL TRIBUNAL

Los suscritos integrantes del Tribunal correspondiente, declaramos que hemos **APROBADO** el Trabajo de Integración Curricular titulado: “Sistema Biométrico para el Control de Acceso a la UDIV de Desarrollo Computacional de la ESPAM–MFL empleando Técnicas de Deep Learning”, que ha sido desarrollado por Lelis Javier Muñoz Ríos y Julio Josué Ponce Almeida, previo a la obtención del título de “Ingeniero en Ciencias de la Computación”, de acuerdo al **REGLAMENTO DE LA UNIDAD DE INTEGRACIÓN CURRICULAR DE CARRERAS DE GRADO** de la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López.

MGTR. LUIS CRISTOBAL CEDEÑO VALAREZO
CC: 130624665-1
PRESIDENTE DEL TRIBUNAL

MGTR. ALFONSO TOMAS LOOR VERA
CC: 131165593-8
SECRETARIO DEL TRIBUNAL

MGTR. ÁNGEL ALBERTO VÉLEZ MERO
CC: 130864856-5
MIEMBRO DEL TRIBUNAL

AGRADECIMIENTO

Agradezco a la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López que nos dio la oportunidad de crecer como profesionales a través de una educación superior de calidad que ha forjado nuestros conocimientos profesionales en el día a día durante la duración de la carrera;

A Víctor Joel Pinargote Bravo, quien bajo su tutelaje llevo a cabalidad el correcto desarrollo del aspecto técnico de este trabajo;

A Jessica Johanna Morales Carrillo que quién con paciencia nos guio y ayudo a comprender la estructura y redacción del presente trabajo de titulación.



LOS AUTORES

DEDICATORIA

Dedico este trabajo a Dios, quien con su bendición llena siempre mi vida y a toda mi familia por estar siempre presentes, mis compañeros de clases y docentes que estuvieron allí brindando sus conocimientos para llevar a cabo este proyecto de tesis. A la Universidad Politécnica de Manabí por ser la sede de todo el conocimiento adquirido en estos años, y a mi familia que tanto me ha apoyado para seguir adelante para cumplir este logro que es uno de los muchos que tengo planteados en mi vida.

A handwritten signature in black ink, appearing to read 'Lelis Muñoz Ríos', with a stylized, cursive script.

LELIS JAVIER MUÑOZ RÍOS

DEDICATORIA

Dedico este trabajo de titulación a mi madre, quien con esfuerzo, sudor y lágrimas dedicó su tiempo y recursos para ayudar en mi desarrollo profesional y con quien amor siempre me apoyó en lo que decidiera estudiar, a mi familia quienes siempre han estado allí para mí en las buenas y las malas, a mi pareja quien me ha dado paz mental durante el desarrollo de este trabajo de titulación y quien también me ha ayudado a no desfallecer en lograr este hito importante en mi vida.



JULIO JOSUÉ PONCE ALMEIDA

CONTENIDO GENERAL

PORTADA.....	i
DECLARACIÓN DE AUTORÍA.....	ii
AUTORIZACIÓN DE PUBLICACIÓN	iii
CERTIFICACIÓN DEL TUTOR	iv
APROBACIÓN DEL TRIBUNAL.....	v
AGRADECIMIENTO.....	vi
DEDICATORIA.....	vii
CONTENIDO GENERAL.....	ix
CONTENIDO DE TABLAS	xii
CONTENIDO DE FIGURAS	xiii
RESUMEN	xv
PALABRAS CLAVE.....	xv
ABSTRACT	xvi
KEYWORDS	xvi
CAPÍTULO I. ANTECEDENTES	1
1.1. DESCRIPCIÓN DE LA INSTITUCIÓN	1
1.2. DESCRIPCIÓN DE LA INTERVENCIÓN	2
1.3. OBJETIVOS	4
1.3.1. OBJETIVO GENERAL.....	4
1.3.2. OBJETIVOS ESPECÍFICOS.....	4
CAPÍTULO II. DESARROLLO METODOLÓGICO DE LA INTERVENCIÓN	5
2.1. IDENTIFICACIÓN DE REQUERIMIENTOS.....	5
2.2. DESARROLLO DEL SISTEMA	5
2.2.1. IDENTIFICACIÓN DEL HARDWARE	6
2.2.2. DESARROLLO DE LA RED NEURONAL.....	6

2.2.3. DESARROLLO DEL PROGRAMA	7
2.3. IMPLEMENTACIÓN.....	8
CAPÍTULO III. DESCRIPCIÓN DE LA EXPERIENCIA	9
3.1. ELICITACIÓN DE REQUERIMIENTOS	9
3.2. ENSAMBLAJE DEL SISTEMA.....	10
3.2.1. SELECCIÓN DE LOS COMPONENTES FÍSICOS DEL SISTEMA	10
3.2.2. SELECCIÓN DEL ALGORITMO DE RECONOCIMIENTO.....	10
3.2.3. PREPARACIÓN DEL EQUIPO RASPBERRY	13
3.2.4. PROGRAMACIÓN DEL APLICATIVO WEB.....	13
3.3. IMPLEMENTACIÓN.....	15
CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES	17
4.1. CONCLUSIONES	17
4.2. RECOMENDACIONES	18
BIBLIOGRAFÍA	19
ANEXOS	22
ANEXO 1. PREGUNTAS DE LA ENTREVISTA.....	23
ANEXO 2. ELICITACIÓN DE REQUERIMIENTOS.....	25
2.1. REQUERIMIENTOS FUNCIONALES.....	25
2.2. REQUERIMIENTOS NO FUNCIONALES	27
ANEXO 3. INFORME SOBRE LOS COMPONENTES DE UN SISTEMA BIOMÉTRICO	30
ANEXO 4. FOTOS DE LAS PRUEBAS DE LA INTELIGENCIA ARTIFICIAL	32
ANEXO 5. INFORME DE PROGRAMACIÓN DE LA INTELIGENCIA ARTIFICIAL.....	35
ANEXO 6. ESTRUCTURA DEL CÓDIGO.....	36
6.1. CÓDIGO PRINCIPAL PARA LA CÁMARA	36
6.2. CÓDIGO LIBRERÍA PARA EL RECONOCIMIENTO FACIAL	38

ANEXO 7. SERVICIO DAEMON.....	41
ANEXO 8. ESTRUCTURA DEL CABLEADO.....	42
ANEXO 9. ARCHIVOS DE BITÁCORA GENERADOS.....	44
11.1. IMÁGENES.....	44
11.2. EJEMPLO DE BITÁCORA GENERADA.....	45

CONTENIDO DE TABLAS

CAPÍTULO III. DESCRIPCIÓN DE LA EXPERIENCIA	9
3.1. ELICITACIÓN DE REQUERIMIENTOS	9
Tabla 1.....	9
3.2. ENSAMBLAJE DEL SISTEMA.....	10
3.2.1. SELECCIÓN DE LOS COMPONENTES FÍSICOS DEL SISTEMA ..	10
Tabla 2.....	10
ANEXOS	22
ANEXO 2. ELICITACIÓN DE REQUERIMIENTOS	25
2.1. REQUERIMIENTOS FUNCIONALES.....	25
Tabla 3.....	25
Tabla 4.....	25
Tabla 5.....	26
Tabla 6.....	26
Tabla 7.....	27
Tabla 8.....	27
2.2. REQUERIMIENTOS FUNCIONALES.....	27
Tabla 9.....	27
Tabla 10.....	28
Tabla 11.....	28
Tabla 12.....	28
Tabla 13.....	29
Tabla 14.....	29

CONTENIDO DE FIGURAS

CAPÍTULO III. DESCRIPCIÓN DE LA EXPERIENCIA	9
3.2. ENSAMBLAJE DEL SISTEMA.....	10
3.2.2. SELECCIÓN DEL ALGORITMO DE RECONOCIMIENTO	10
Imagen 1	11
Imagen 2.....	11
Imagen 3.....	12
3.2.3. PREPARACIÓN DEL EQUIPO RASPBERRY	13
Imagen 4.....	13
3.2.4. PROGRAMACIÓN DEL APLICATIVO WEB.....	13
Imagen 5.....	14
Imagen 6.....	15
3.3. IMPLEMENTACIÓN.....	15
Imagen 7.....	15
Imagen 8.....	16
ANEXOS	22
ANEXO 4. FOTOS DE LAS PRUEBAS DE LA INTELIGENCIA ARTIFICIAL	32
Imagen 9.....	32
Imagen 10.....	32
Imagen 11.....	33
Imagen 12.....	33
Imagen 13.....	34
ANEXO 5. INFORME DE PROGRAMACIÓN DE LA INTELIGENCIA ARTIFICIAL.....	35
Imagen 14.....	35
Imagen 15.....	35

ANEXO 8. ESTRUCTURA DEL CABLEADO.....	42
Imagen 16.....	42
Imagen 17.....	42
Imagen 18.....	43
ANEXO 9. ARCHIVOS DE BITÁCORA GENERADOS.....	44
11.1. IMÁGENES.....	44
Imagen 19.....	44
Imagen 20.....	44

RESUMEN

En el presente trabajo de titulación se desarrolló un sistema biométrico de control de acceso mediante el uso de técnicas de Deep Learning enfocado en el reconocimiento facial, mismo que se implementó en la UDIV de Desarrollo Computacional de la Escuela Superior Politécnica de Manabí MFL, como una alternativa de sistema de seguridad; en primera instancia se realizó un levantamiento de los requerimientos mediante una entrevista al encargado del área sobre la seguridad del sitio y cualquier tipo de antecedente existente produciendo un documento bajo el estándar IEEE 890-1998, la solución implementada consta de dos partes fundamentales, como son: la inteligencia artificial, que es la encargada de recibir los datos de la cámara de vigilancia y compararlos con una base de datos de fotos usando autoencoders, todo esto siguiendo la metodología CRISP-DM, y un módulo web para administrar la información de los usuarios y visualizar los reportes de los diferentes accesos, esta segunda fase se la desarrolló mediante el uso de Programación Extrema (XP), entre los principales resultados se evidencia una precisión de un 98% en el modelo de reconocimiento facial gracias a los datos obtenidos de los trabajadores para su utilización en el anteriormente mencionado y un módulo de reportes que muestra los diferentes horarios de acceso de los empleados.

PALABRAS CLAVE

Control de acceso, sistema biométrico, seguridad, Deep Learning, reconocimiento facial, inteligencia artificial.

ABSTRACT

In this degree work, a biometric access control system was developed through the use of Deep Learning techniques focused on facial recognition with the express purpose of being implemented in the UDIV of Computational Development as a security system; first, a requirements gathering was carried out through an interview with the main head of the area about the security on site and any type of existing precedents, producing a document under the IEEE 890-1998 standard, the development and correct design of the fundamental parts of the system was carried out then, such as the artificial intelligence that will be in charge of receiving the data from the surveillance camera and interpreting them, all this following the CRISP-DM methodology, a database to manage the information and a graphical user interface that allows an optimal and understandable user experience through the use of Xtreme Programming (XP), amongst the principal results It was evident a precision of 98% in the face recognition model thanks to the obtained data from the workers for its usage in the aforementioned and a report module that showcases the different access schedules of employees

KEYWORDS

Control Access, biometric system, security, Deep Learning, facial recognition, artificial intelligence.

CAPÍTULO I. ANTECEDENTES

1.1. DESCRIPCIÓN DE LA INSTITUCIÓN

La Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López, la cual a partir de ahora será referida como ESPAM MFL, se trata de acuerdo con su propio estatuto (ESPAM MFL, 2019) como “una institución de educación superior sin fines de lucro” la cual se rige por los principios de autonomía responsable y calidad, cogobierno, igualdad de oportunidades, democracia, integridad, autodeterminación para el desarrollo de producción científica y tecnológica global; así como principios de la universidad tales igualdad, equidad y la no discriminación por citar algunos. La misión de la ESPAM MFL, de acuerdo con el mismo estatuto, es “Formar profesionales pertinentes con compromiso ético y social desde la calidad de las funciones sustantivas de la ESPAM MFL” y su visión como institución es ser un centro de referencia siguiendo su misión para que estos contribuyan al desarrollo agropecuario de la región entera.

La ESPAM-MFL (1999), comparte una misión y visión estrechamente relacionada con la carrera de Computación (2003), siendo la tecnología un puente para lograr los diferentes objetivos planteados en el marco institucional, enfocándose en el siempre cambiante mundo tecnológico y en ofrecer los mejores graduados al ámbito profesional, dentro de la estructura que conforma la facultad de esta carrera tenemos las Unidades de Docencia, Investigación y Vinculación o UDIV para acortar, las cuales se tratan, como su nombre lo indica, de todas las áreas referentes a los ámbitos de la docencia y a la educación, tales como son la investigación y la vinculación de alumnos a proyectos universitarios tanto dentro como fuera de la ESPAM-MFL así como las actividades y proyectos relacionados con la docencia, y como parte de este conglomerado se encuentra la UDIV de Desarrollo Computacional, que como su nombre lo indica se encuentra encargada del acompañamiento a los estudiantes en proyectos de desarrollo de software así como su gestión, capacitaciones tecnológicas y eventos académicos, por lo que la seguridad e integridad de esta zona con sus equipos tecnológicos y archivos digitales son de suma importancia para salvaguardar toda la unidad.

1.2. DESCRIPCIÓN DE LA INTERVENCIÓN

La inteligencia artificial, de acuerdo con Brynjolfsson y McAfee (2017), junto a los autores Malik et al (2019) y Mintz y Brodie (2019); se trata de un término acuñado por John McCarthy, quien fue un profesor de matemáticas en Dartmouth, en el año 1956 como la ciencia e ingeniería de crear máquinas inteligentes durante una conferencia en el tema; si bien en su momento evocaba un futuro fantástico y en cierto modo etéreo actualmente solo se refiere un conjunto de algoritmos que tratan de imitar los procesos humanos, da como paso a un subconjunto de técnicas muchas veces denominada Machine Learning y dentro de estas encontramos las redes neuronales o Neural Networks, de las cuales nace el Deep Learning como concepto en el año 2006 (Goh et al, 2017), (Hu et al, 2021) y (Alom et al, 2019); y de acuerdo con Albert et al (2019), pueden ser aplicadas en una infinidad de problemáticas los cuales son el reconocimiento de imágenes y voz, sintetizadores, entre otras diversas aplicaciones como predicciones o manejo autónomo de maquinaria como son los automóviles (Pouyanfar et al, 2018), y estos procesos pueden ser aplicados en varios ámbitos profesionales, tales como medicina, biología, o entre otros tantos la seguridad, sin embargo nos enfocaremos en este último caso en el cual nos dice que la inteligencia artificial en este campo, según un Rondo en 2021, es útil en este campo porque se tendría menos falsos positivos con los que se debe lidiar, un modelo bien entrenado podría reducir el tiempo de respuesta entre las posibles decisiones que se pueden y podrían llegar a tomar. Uno de los tantos usos de la técnica de redes neuronales se trata, de acuerdo con Dahea y Fadewar (2018) junto con Gawande y Golhar (2018) y Gavrilova et al (2017), de los sistemas biométricos para frenar posibles amenazas de seguridad, los cuales miden al menos un atributo o comportamiento físico como son, por citar algunos ejemplos: huellas digitales (Yang et al, 2019), iris (Severo et al, 2018), firmas (Channegowda & Prakash, 2021), voz (Boles & Rad, 2017) y entre ellas encontramos la cara (Ortiz et al, 2018), por lo que puede reconocer facialmente a una persona mediante redes neuronales adecuadas para lograr trabajar en este sistema.

La ESPAM-MFL cuenta con la UDIV de Desarrollo Computacional, misma que se encarga de acompañar a los estudiantes en proyectos de desarrollo de software

así como su gestión, capacitaciones tecnológicas y eventos académicos, todas las actividades antes mencionadas interactúan con información sensible, por lo que el acceso a personal no autorizado a estas instalaciones, se considera una vulnerabilidad y amenaza a futuro, por tanto, en la presente sistematización de experiencias se implementó un sistema de reconocimiento facial basado en redes neuronales para aumentar los protocolos de seguridad al ingresar a esta zona. Esta propuesta, además, cuenta con el total apoyo y gestión económica por parte de la institución de los equipos necesarios para la implementación del proyecto.

1.3. OBJETIVOS

1.3.1. OBJETIVO GENERAL

Implementar un Sistema Biométrico para el control de acceso a la UDIV de Desarrollo Computacional de la ESPAM MFL empleando técnicas de Deep Learning.

1.3.2. OBJETIVOS ESPECÍFICOS

- Identificar los requisitos necesarios para el diseño del Sistema Biométrico.
- Desplegar los componentes fundamentales que conforman el Sistema Biométrico.
- Instalar el sistema resultante a la Unidad de Desarrollo Computacional de acuerdo con el alcance planteado.

CAPÍTULO II. DESARROLLO METODOLÓGICO DE LA INTERVENCIÓN

El presente trabajo de titulación se realizó con la expresa finalidad de diseñar e implementar un sistema biométrico con reconocimiento facial para controlar el acceso a la UDIV de Desarrollo Computacional, en base a los objetivos específicos propuestos se llegó a la conclusión que el desarrollo estaría planificado mediante fases, las cuales son:

2.1. IDENTIFICACIÓN DE REQUERIMIENTOS

El proceso que llevó a la identificación de requerimientos que conformaron el desarrollo del Sistema Biométrico inició con la aplicación de una entrevista formal abierta la cual se trata de un documento altamente estructurado pero que permite libertad en las respuestas para que el entrevistado provea toda la información que desee (Turner III y Hagstrom-Schmidt, 2022), mediante la cual se logró una correcta toma de requisitos gracias al jefe encargado de la UDIV quién era el Magister Yimmy Salvador Loor Vera en el momento de la realización de la entrevista, este proceso dio como resultado un documento que se encuentra bajo el estándar IEEE 830-1998 cuyo acrónimo significa Instituto de Ingenieros Eléctricos y Electrónicos, que según el autor Rodríguez (2020), y que permite la descripción precisa de lo que se requiere en el software, así como definir el formato y contenido de una manera estándar todo esto para evitar que se inicie el trabajo sin antes revisar y aprobar de manera rigurosa todo los requisitos planteados lo cual permite que ambas partes estén de acuerdo en cómo será el software una vez diseñado y codificado.

2.2. DESARROLLO DEL SISTEMA

En esta fase, tal y como su nombre indica, se desarrolló el Sistema Biométrico en base a los requisitos levantados en la fase anterior, sin embargo para poder lograr esto de manera óptima, se tuvo que diferenciar esta fase en dos secciones específicas, las cuáles se enfocaron en la sección de hardware y en las secciones de software, esta última se refiere al conjunto de datos (datasets) donde se obtuvieron y guardaran los datos para el entrenamiento de la Red Neuronal y el

aplicativo Web, desde donde se realizaron las tareas de administrador tales como: añadir, editar o eliminar personas, mismos que fueron los usuarios de este sistema.

2.2.1. IDENTIFICACIÓN DEL HARDWARE

Para el desarrollo este tipo de sistema biométrico fue necesario determinar que hardware mínimo y de qué tipo se necesitaba para poder empezar a trabajar, las características y requisitos de sistema, ya sean procesadores, equipos tecnológicos, mecánicos y periféricos en general, así como el cableado necesario para interconectar todo el sistema.

Es por ello, que se investigó el tipo de componentes básicos y fundamentales que se suelen encontrar en un sistema biométrico con base en reconocimiento de patrones en imágenes lo cual dio como resultado un informe investigativo mediante una metodología investigativa o bibliográfica, que de acuerdo con Fuentes-Doria et al (2020), nos permite la recopilación, codificación y análisis de diversos datos e información de forma simultánea con tal de demostrar la plausibilidad de una teoría o proyecto para saber si llevarla a cabo es posible.

2.2.2. DESARROLLO DE LA RED NEURONAL

Dentro de este proyecto el desarrollo y programación de una Red Neuronal capaz de discernir la información de una foto o video y dictaminar la presencia de una persona en específico fue y es la parte troncal de todo, es por ello que para su correcto desarrollo y creación de un correcto flujo de trabajo fue necesario escoger una metodología de trabajo que se acople a las técnicas de Deep Learning, que en este caso se trata de CRISP – DM que proviene del inglés “CRoss-Industry Standard Process for Data Mining” el cuál podría interpretarse como “Proceso Estándar Trans – Industria para la Minería de Datos”, que de acuerdo con las aseveraciones del autor Martínez-Plumed et al en el año 2019, es una metodología nacida en la segunda década de los años noventa, por lo que este método supera las dos décadas de antigüedad y en su tiempo de existencia se convirtió en un estándar de facto para este tipo de proyectos.

De acuerdo con los autores Huber et al (2019) el CRISP – DM se encuentra

conformado por las siguientes fases:

- **Análisis del problema:** En esta parte de la metodología se realiza el estudio de caso, se establece los objetivos mediante la evaluación de la situación actual y se obtiene el plan a seguir
- **Análisis de los datos:** Se comienza a describir el dataset a utilizar a la par que se comienza a explorar y capturar los datos a utilizar, en caso de no lograr esto se vuelve a la fase de “Análisis del problema”
- **Preparación de los datos:** Se comienzan a recortar el dataset para limpiarlo de datos no requeridos o redundantes, todo esto para lograr un buen entrenamiento del modelo.
- **Modelado:** Esta fase como su nombre lo indica se dedica a modelar las relaciones en base a los datos y objetivos planteados mediante las técnicas más adecuadas, de no poder lograr esto satisfactoriamente es porque se necesita volver a la “Preparación de los datos”.
- **Evaluación:** Una vez realizado el modelado se procede a evaluar el modelo generado, en caso de no pasar la evaluación, se empieza de nuevo desde el “Análisis del problema”
- **Despliegue o Implementación:** Esta fase final es donde implementamos el resultado a la par que se realiza el seguimiento y mantenimiento, en este caso la red neuronal será implementada a un aplicativo Web

2.2.3. DESARROLLO DEL PROGRAMA

En esta fase del proyecto se tuvieron que considerar dos aspectos importantes, la base de datos con la cual la Red Neuronal trabaja e interactúa, así como el aplicativo Web que funciona como interfaz de usuario para los administrativos.

Para la codificación del aplicativo así como el diseño y enlazamiento con las partes anteriores fue utilizada la metodología de Programación Extrema o XP (por “eXtreme Programming” en inglés) y , según explican los autores Salazar et al en el año 2018, es una metodología adaptable y flexible que permite que un proyecto pueda moverse libremente con requisitos previamente establecidos, así como mantener al cliente en estrecha relación durante la vida del mismo, esta forma de trabajo fue propuesta por Kent Beck en el año 1999 y fue concebida con las

siguientes fases:

- **Planificación:** En esta fase se toma en cuenta toda la información a trabajar por lo que es importante contar con el documento en el estándar IEEE 830-1998 que se levantó durante la primera fase.
- **Diseño:** En esta fase se trabaja con una representación visual sencilla del sistema y se definen las características principales del futuro código esto usualmente mediante el uso de Modelado UML, de acuerdo con los autores Silva et al (2018), la abstracción de un sistema para lograr construirlo de acuerdo a los requerimientos, especificaciones o términos concretos mediante la comprensión de estos dejando de lado detalles no esenciales que permitan facilitar la comprensión de como funcionaria el sistema, esto es fundamental para lograr explicar el funcionamiento a un cliente que no conozca mucho del lenguaje técnico involucrado en el desarrollo de software, generado todo esto finalmente diagramas varios tales como Diagramas de Entidad – Relación y de Clase junto con demás métodos y procesos de abstracción
- **Desarrollo:** Como su nombre lo indica esta es la fase crítica donde se realiza toda la codificación del sistema modelado con anterioridad y siguiendo los requerimientos planteados.
- **Pruebas:** En esta fase se realizan las pruebas necesarias para verificar el correcto flujo del sistema y de que no existan incongruencias y de existir las se deben de corregir. Una vez realizado esto se entrega el software y el manual de programador o documentación del programa.

2.3. IMPLEMENTACIÓN

Finalmente, el sistema biométrico fue implementado dentro de la UDC mediante el diseño de la disposición del cableado que unió la máquina central con los dispositivos (cámaras, el cerrojo magnético, entre otros) y así fue concretada la correcta implementación del sistema con la conexión entre dispositivos dando acceso solo a personal autorizado y guardando capturas de personas desconocidas o no autorizadas para su posterior reseña.

CAPÍTULO III. DESCRIPCIÓN DE LA EXPERIENCIA

A lo largo del desarrollo del presente trabajo de titulación se cumplieron los diferentes objetivos propuestos tratando de seguir lo más posible la planeación expuesta en la sección de Desarrollo Metodológico debido a que por motivos de presupuesto, tiempo o de circunstancias inesperadas esto no fue del todo posible, por lo que se procederá a describir la experiencia en diferentes secciones bien diferenciadas.

3.1. ELICITACIÓN DE REQUERIMIENTOS

Durante las primeras semanas se redactó un documento con una serie de preguntas respecto a la seguridad de la UDIVs de Desarrollo Computacional y posteriormente a esto se concertó una cita con el Mgtr. Yimmy Salvador Loor Vera, responsable de la UVIDs, una vez realizada la entrevista (Anexo 1), con información obtenida se procedió a levantar los requisitos necesarios para lograr un programa que se adecue a las necesidades recabadas (Tabla 2).

Tabla 1

Resumen de los principales requerimientos obtenidos de la entrevista (Véase Anexo 2 para mayor detalle)

Número de requisito	Descripción	Tipo de Requisito
RF-01	Autenticación de usuario	Funcional
RF-02	Registro de personas	Funcional
RF-04	Consulta de información de acceso	Funcional
RF-06	Control de Acceso	Funcional
RNF-01	Interfaz Gráfica de Usuario	No funcional
RNF-03	Diseño de la interfaz ESPAM-MFL	No funcional
RNF-05	Confiabilidad continúa	No funcional

A partir de la extrapolación de los requisitos, mediante el estándar IEEE 80-1998, se obtuvieron las directrices para poder llevar a cabo la siguiente fase.

3.2. ENSAMBLAJE DEL SISTEMA

3.2.1. SELECCIÓN DE LOS COMPONENTES FÍSICOS DEL SISTEMA

Tras la fase anterior, se realizó una investigación sobre los componentes básicos que conforman un Sistema Biométrico común con las características propuestas (Anexo 3) para identificar los objetos o hardware necesarios para llevar a cabo el desarrollo, y en la Tabla 2 podemos observar los componentes físicos necesarios para llevar a cabo el proyecto.

Tabla 2

Componentes necesarios para el Sistema Biométrico.

Cantidad	Detalle de insumo	Descripción
1	Placa Raspberry Pi 4 de 8GB de memoria RAM	Equipo utilizado para el desarrollo y entrenamiento de la red neuronal
1	Cámara HD (720p)	Entradas de datos para las pruebas
2	Relay USB	Dispositivos interruptores USB
1	Cerradura Magnética	Dispositivo de cierre electrónico
1	Brazo Hidráulico	Dispositivo de cierre electrónico

3.2.2. SELECCIÓN DEL ALGORITMO DE RECONOCIMIENTO

Tras definir los componentes de un Sistema Biométrico se procedió a dilucidar el método de reconocimiento facial a utilizar, se pasaron por diversos métodos basados en tecnologías como Torch y CNN, que, si bien eran buenas estrategias debido a las limitantes dadas por usar una "RaspBerry Pi 4" fueron, finalmente, por ser desechadas, pero dejando constancia en las Imágenes 1 y 2 (Anexo 4 para más imágenes).

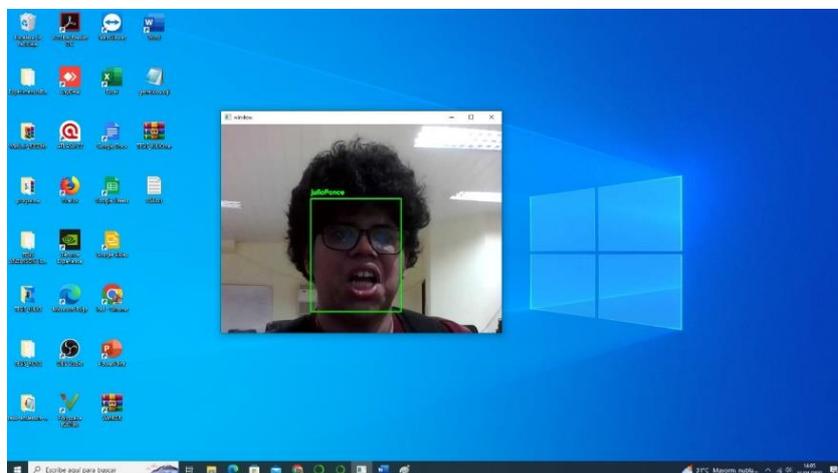
Imagen 1

Captura de pantalla del reconocimiento facial del tutor



Imagen 2

Captura de pantalla del reconocimiento facial del programador, reconocimiento logrado incluso con muecas



Las imágenes anteriores hacen referencia al uso de CNN, esta tecnología es muy utilizada en este tipo de proyectos, pero dada las restricciones y capacidad de la "RaspBerry Pi 4" el sistema terminaba siendo relativamente lento generando respuestas tardías.

Dentro de la Imagen 3 podemos observar un “diccionario” de datos, entre los que podemos observar detalles como el “min_face_size” que se refiere el tamaño mínimo en el que las caras a reconocer se reducirían, “thresholds” o umbrales, que dictaminan los límites en los cuales funcionará el algoritmo e incluso “min_confidence” que se refiere al mínimo de parecido que se debe de obtener para adjudicar las imágenes a una etiqueta de la base de datos, “threshold_similaridad” el cual se trata de la variable que determina que tan similar puede ser la nueva imagen a los umbrales para poder determinar un resultado y “verbose” que simplemente es una opción que permite generar bitácoras de la ejecución; otras variables como “device” (que recibe una dirección de periférico) o “imagen” (la cual recibe una imagen que será la referencia para el reconocimiento) son bastante autoexplicativas.

Imagen 3

Fragmento del código donde podemos observar el diccionario de datos

```
In [3]: # Reconocimiento en imágenes
# =====
# Detectar si se dispone de GPU cuda
device = torch.device('cuda:0' if torch.cuda.is_available() else 'cpu')
print(f'Running on device: {device}')

# Identificar Las personas en La imagen
fig, ax = plt.subplots(figsize=(12, 7))
imagen = Image.open('3.jpg')

pipeline_deteccion_imagen(
    imagen = imagen,
    dic_referencia = dic_referencias,
    min_face_size = 20,
    thresholds = [0.6, 0.7, 0.7],
    min_confidence = 0.5,
    threshold_similaridad = 0.6,
    device = device,
    ax = ax,
    verbose = False
)

Running on device: cpu
```

Tras todo esto, al final se optó por empezar de nuevo y obtener un reconocimiento facial mediante Autoencoders y OpenCV (Librería de Visión de Código Abierto de Machine Learning) importando el módulo “cv2”.

Tras decidir finalmente la forma en la que se reconocerían los rostros se optó por utilizar fotos donde sólo la parte a reconocer fuese relevante especialmente ya que ello permitió obtener más detalles tras el procesado llevando a un mejor resultado final, tras ello se procedió a preparar los datos a utilizar para alimentar el sistema, así como generar las relaciones necesarias para lograr un correcto funcionamiento

del código especialmente debido a las limitantes de la placa controladora “Raspberry Pi 4” junto con el equipo de captura de imagen disponible.

3.2.3. PREPARACIÓN DEL EQUIPO RASPBERRY

Tras finalizar la fase anterior, se optó por un programa como servicio que corriera todo el tiempo, esto con tal de que en caso de fallo del sistema o de la red eléctrica se pudiera restablecer el servicio automáticamente en un lapso de un minuto todo esto gracias a las facilidades que presta el sistema operativo Raspbian, los detalles de cómo se logró esto pueden encontrarse en los Anexos 6 para la estructura del código y 7 para el archivo que permite se pueda lanzar el servicio, sumado a esto podemos observar en la Imagen 4 la estructura de carpetas utilizadas para el código, ordenadas de tal forma que se pueda saber en todo que datos se encuentran que carpetas ya sean las bitácoras de entrada o los rostros almacenados.

Imagen 4

Jerarquía de las carpetas del programa



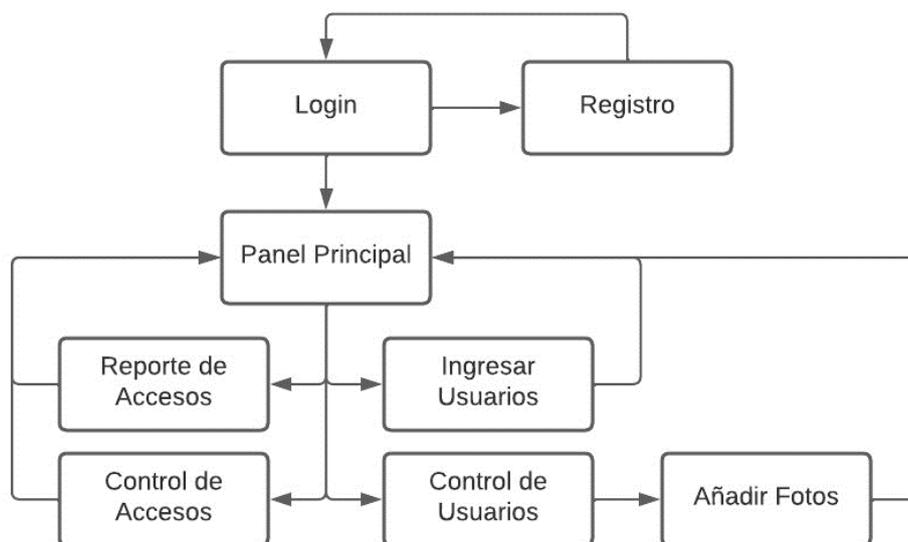
3.2.4. PROGRAMACIÓN DEL APLICATIVO WEB

Tras lograr tener un concepto funcional de lo que sería la parte de reconocimiento del sistema, se comenzó a trabajar en la parte administrativa del mismo, el lugar donde se otorgaría permisos de acceso y se añadirían nuevos usuarios siguiendo las directrices fundamentadas en la elicitación de requerimientos, para codificar los

formularios Web y la interfaz, mismos que permitieran cumplir los requisitos propuestos por lo cual se decidió utilizar el “Framework” de Laravel debido a las facilidades que ofrece como validación de identidades y un sistema integrado de roles de usuario así como las habilidades de recibir y enviar paquetes de información o poder ejecutar códigos sencillos en una ventana de comandos, aspecto íntegramente necesario dentro de los requisitos recabados. Como podemos observar en la Imagen 5 se optó por un flujo convencional, con un sistema de acceso a cuenta, registro en caso de no tener una, y tras ingresar exitosamente poder añadir, eliminar y realizar otras acciones básicas.

Imagen 5

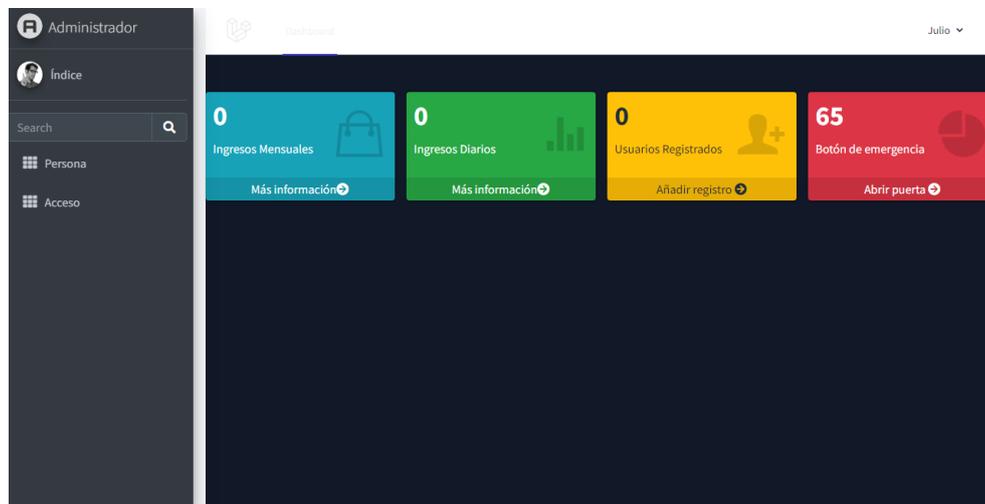
Flujo principal de las páginas



Tras ello se procedió a llevar estos conceptos a la programación, como podemos observar en la Imagen 6 donde se cumplió en su mayor medida el flujo, además de poder abrir la puerta mediante un fragmento de código en el “Botón de Emergencia” aprovechando la capacidad de Laravel para poder ejecutar comandos.

Imagen 6

Página principal del sistema



3.3. IMPLEMENTACIÓN

Finalmente se procedió a reutilizar el cableado pre-existente dentro de la UDIV con el objetivo expreso de reducir la planeación y aprovechar la infraestructura que el área proveía, realizando las instalaciones adecuadas para el correcto funcionamiento, conectando el equipo al fluido eléctrico, formando un sistema parecido al del diagrama expresado en la Imágenes 7 y 8 a continuación

Imagen 7

Diagrama del equipo

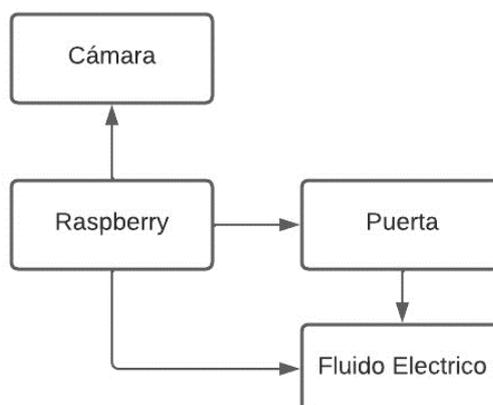


Imagen 8

Raspberry Pi 4 instalada (véase Anexo 8 para más imágenes del equipo instalado)



Tras ello, diversas pruebas fueron realizadas para confirmar el correcto guardado de los accesos generados (Anexo 9) en la base de datos mediante el uso de una API que recibiría un código numérico por parte del programa que corresponde a una persona dentro de la base de datos.

CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES

4.1. CONCLUSIONES

Luego de analizar los requisitos y los diferentes métodos de reconocimiento facial se define trabajar con una metodología basada en autoencoders para la codificación de los rostros de los usuarios debido a que esta estrategia es mucho más rápida y precisa en comparación a otras técnicas.

Los componentes esenciales del Sistema Biométrico, incluyendo la integración de autoencoders para la codificación de rostros, fueron desarrollados con éxito. Se implementaron técnicas de Deep Learning para la extracción y representación eficiente de las características biométricas, lo que contribuye a la precisión y robustez del sistema en la identificación y verificación de usuarios.

Finalmente, la instalación de un Sistema Biométrico basado en técnicas de Deep Learning, especialmente la utilización de autoencoders para la codificación de rostros, ha demostrado ser efectiva en cumplir con los objetivos establecidos. Este sistema no solo cumple con los requisitos de diseño, sino que también se integra de manera exitosa en la infraestructura existente, proporcionando un método seguro y eficiente para el control de acceso en la Unidad de Desarrollo Computacional de la ESPAM MFL. Estos resultados subrayan la viabilidad y utilidad de la aplicación de técnicas avanzadas de procesamiento de imágenes en entornos biométricos para la mejora de la seguridad y la gestión de accesos.

4.2. RECOMENDACIONES

Para lograr recopilar la mayor cantidad de información posible para realizar el correcto levantamiento de requerimientos de una manera satisfactoria, los autores recomiendan que se lleven a cabo diversos métodos de obtención de información de manera paralela o concurrente, todo lo que se pueda recabar u obtener puede ser vital para el desarrollo correcto del proyecto para no tener cambios de última hora o que de haberlos sean fácilmente arreglados, así como la satisfacción de la persona a la cual el proyecto podría ir dirigido.

Con respecto al desarrollo un programa y cualquier sub-proyecto adyacente los autores recomiendan el uso de diversas herramientas para poder acceder en todo momento al código o poder realizar un respaldo del mismo, ya sea mediante almacenamiento en la nube o repositorios compartidos para evitar perdida de datos o en caso de modificar algo que pueda poner en jaque la integridad del proyecto pueda, mediante el control de versiones, volver a una etapa anterior y en caso de estar trabajando en algo en que se necesita un alto poder computacional es necesario tener un ordenador que permita el acceso remoto, esto es para poder lidiar satisfactoriamente con eventos varios, como por ejemplo enfermedad, accidentes o viajes.

Finalmente, se recomienda que en el momento de concretar de manera física un proyecto de estas características se tome en cuenta todos los posibles escenarios en los que los usuarios finales puedan hacer uso del sistema, esto mediante pruebas exhaustivas antes de considerar finalizada su implementación, estén registrados o no mediante métodos alternativos de entrada.

BIBLIOGRAFÍA

Alber, M., Lapuschkin, S., Seegerer, P., Hägele, M., Schütt, K. T., Montavon, G., & Kindermans, P. J. (2019). INNvestigate neural networks! *J. Mach. Learn. Res.*, 20(93), 1-8.

Alom, M. Z., Taha, T. M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M. S., & Asari, V. K. (2019). A state-of-the-art survey on deep learning theory and architectures. *Electronics*, 8(3), 292.

Boles, A., & Rad, P. (2017, June). Voice biometrics: Deep learning-based voiceprint authentication system. In *2017 12th System of Systems Engineering Conference (SoSE)* (pp. 1-6). IEEE.

Brynjolfsson, E., & McAfee, A. (2017). Artificial intelligence, for real. *Harvard business review*, 1, 1-31.

Channegowda, A. B., & Prakash, H. N. (2021). Multimodal biometrics of fingerprint and signature recognition using multi-level feature fusion and deep learning techniques. *Indones. J. Electr. Eng. Comput. Sci.*, 22(1), 187.

Dahea, W., & Fadewar, H. S. (2018). Multimodal biometric system: A review. *International Journal of Research in Advanced Engineering and Technology*, 4(1), 25-31.

ESPAM MFL. (1999). Misión y Visión. Retrieved May 24, 2022, <http://www.espam.edu.ec/web/unidades/tthh.aspx#:~:text=Nuestra%20Visión,beneficio%20de%20la%20comunidad%20politécnica>.

ESPAM MFL. (2003). Carrera de Computación. Retrieved May 24, 2022, from <http://www.espam.edu.ec/web/oferta/grado/computacion.aspx>

ESPAM MFL. (2019). Estatuto. Retrieved May 24, 2022, from <http://espam.edu.ec/recursos/sitio/espam/ESTATUTOESPAMMFL20190910-CES.pdf>

Fuentes-Doria, D. D., Toscano-Hernández, A. E., Malvaceda-Espinoza, E., Díaz Ballesteros, J. L., & Díaz Pertuz, L. (2020). Metodología de la investigación: Conceptos, herramientas y ejercicios prácticos en las ciencias administrativas y contables.

Gavrilova, M. L., Ahmed, F., Azam, S., Paul, P. P., Rahman, W., Sultana, M., & Zohra, F. T. (2017). Emerging trends in security system design using the concept of social behavioural biometrics. *Information Fusion for Cyber-Security Analytics*, 229-251.

Gawande, U., & Golhar, Y. (2018). Biometric security system: a rigorous review of unimodal and multimodal biometrics techniques. *International Journal of Biometrics*, 10(2), 142-175.

Goh, G. B., Hodas, N. O., & Vishnu, A. (2017). Deep learning for computational chemistry. *Journal of computational chemistry*, 38(16), 1291-1307.

Hu, Y., Kuang, W., Qin, Z., Li, K., Zhang, J., Gao, Y., ... & Li, K. (2021). Artificial intelligence security: threats and countermeasures. *ACM Computing Surveys (CSUR)*, 55(1), 1-36.

Huber, S., Wiemer, H., Schneider, D., & Ihlenfeldt, S. (2019). DMME: Data mining methodology for engineering applications—a holistic extension to the CRISP-DM model. *Procedia Cirp*, 79, 403-408.

Malik, P., Pathania, M., & Rathaur, V. K. (2019). Overview of artificial intelligence in medicine. *Journal of family medicine and primary care*, 8(7), 2328.

Martínez-Plumed, F., Contreras-Ochando, L., Ferri, C., Hernández-Orallo, J., Kull, M., Lachiche, N., ... & Flach, P. (2019). CRISP-DM twenty years later: From data mining processes to data science trajectories. *IEEE Transactions on Knowledge and Data Engineering*, 33(8), 3048-3061.

Mintz, Y., & Brodie, R. (2019). Introduction to artificial intelligence in medicine. *Minimally Invasive Therapy & Allied Technologies*, 28(2), 73-81.

Ortiz, N., Hernández, R. D., Jimenez, R., Mauledeoux, M., & Avilés, O.

(2018). Survey of biometric pattern recognition via machine learning techniques. *Contemporary Engineering Sciences*, 11(34), 1677-1694.

Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M. P., & Iyengar, S. S. (2018). A survey on deep learning: Algorithms, techniques, and applications. *ACM Computing Surveys (CSUR)*, 51(5), 1-36.

Rodríguez, A. S. (2021). Estándares IEEE sobre el análisis y diseño de software en proyectos de control de información. *Nextia*, (11), 9-26.

Rondo, G. (2021). Inteligencia Artificial en la Seguridad de TI. *INF-FCPN-PGI Revista PGI*, (8), 99–101. Recuperado a partir de https://ojs.umsa.bo/ojs/index.php/inf_fcpn_pgi/article/view/59

Salazar, J. C., Casallas, Á. T., Linares, J. C., Lozano, A., & Valbuena, Y. L. (2018). Scrum versus XP: similitudes y diferencias. *Tecnología Investigación y Academia*, 6(2), 29-37.

Severo, E., Laroca, R., Bezerra, C. S., Zanlorensi, L. A., Weingaertner, D., Moreira, G., & Menotti, D. (2018, July). A benchmark for iris location and a deep learning detector evaluation. In *2018 international joint conference on neural networks (IJCNN)* (pp. 1-7). IEEE.

Silva, A., Ledezma, E., Castorena, J., Domínguez, A., & Riojas, A. (2018). Utilidad del Lenguaje Unificado de Modelado (UML) en el desarrollo de software profesional dentro del sector empresarial y educativo. *CienciaCierta revista de divulgación científica*, 56.

Turner III, D. W., & Hagstrom-Schmidt, N. (2022). Qualitative interview design. *Howdy or Hello? Technical and Professional Communication*.

Yang, W., Wang, S., Hu, J., Zheng, G., & Valli, C. (2019). Security and accuracy of fingerprint-based biometrics: A review. *Symmetry*, 11(2), 141.

ANEXOS

ANEXO 1. PREGUNTAS DE LA ENTREVISTA

Pregunta: ¿Qué papel desempeña dentro de la UDC?

Respuesta: Responsable de las Unidades de Docencia, Investigación y Vinculación.

Pregunta: ¿Ha tenido inconvenientes con terceros accediendo a la UDC sin ningún tipo de permiso?

Respuesta: Sí

Pregunta: ¿Ha tenido experiencias negativas debido a pérdidas? ¿Equipo de trabajo u objetos personales?

Respuesta: Sí

Pregunta: ¿Cómo catalogaría la seguridad de la UDC?

Respuesta: Poca seguridad.

Pregunta: ¿Cree usted que muchos de los problemas referente a pérdidas de material personal o de la propia oficina se deba a la falta del control de acceso?

Respuesta: Sí.

Pregunta: ¿Está de acuerdo usted que las cámaras registren la entrada y salida de todo el personal y terceros?

Respuesta: Por supuesto que sí.

Pregunta: ¿Considera usted que la información que se maneja aquí es sensible? De ser así ¿Que niveles de sensibilidad se maneja aquí?

Respuesta: Sí, del 1 al 10 le doy un 5.

Pregunta: ¿Consideraría útil que se maneje una bitácora de ingreso y salida de todo quien entre a los predios de la UDC?

Respuesta: Sí

Pregunta: ¿Está usted de acuerdo que esto se logre mediante el uso de técnicas de Machine Learning y Redes Neuronales Convolutivas?

Respuesta: Como carrera de Computación debemos estar a la vanguardia de la innovación y tecnología así que si estoy totalmente de acuerdo.

Pregunta: ¿Desea que los datos nuevos se ingresen por personal de la UDIV? ¿Qué opciones cree serían necesarias para poder manejar a gusto el programa?

Respuesta: Sí, que la experiencia de usuario sea muy buena, una interfaz amigable y sobre todo funcional.

Pregunta: ¿Preferiría una aplicación que este centralizada en un ordenador o un aplicativo web para abrir las opciones en cualquier parte de la red? ¿Por qué?

Respuesta: En un aplicativo web, pero a donde se vaya a guardar la información sea seguro y porque me inclino por esta opción la respuesta es sencilla quisiera que a través de una notificación me advierta que persona ajena a la Carrera de computación ingresa a la UDC.

Pregunta: ¿Está consciente de que el sistema necesita estar constantemente conectado a corriente para funcionar correctamente?

Respuesta: Es que así debería de ser, de ser posible conectarlo a un banco de baterías

ANEXO 2. ELICITACIÓN DE REQUERIMIENTOS

2.1. REQUERIMIENTOS FUNCIONALES

Tabla 3

Requisito funcional "01"

Identificación del requerimiento:	RF-01
Nombre del requerimiento:	Autenticación de usuario
Características	Los administradores deberán de identificarse para acceder al sistema.
Descripción	El sistema puede ser consultado por los administradores
Requerimientos no funcionales:	RNF-01, RNF-02, RNF-04, RNF-06
Prioridad:	Alta

Tabla 4

Requisito funcional "02"

Identificación del requerimiento:	RF-02
Nombre del requerimiento:	Registro de personas
Características	Los administradores podrán registrar personas en el sistema para ser reconocidas. Los datos suministrados deberán ser: Nombres, Apellidos, Teléfono celular y Cédula
Descripción	El sistema puede ser consultado por los administradores
Requerimientos no funcionales:	RNF-01, RNF-02, RNF-04, RNF-06
Prioridad:	Alta

Tabla 5

Requisito funcional "03"

Identificación del requerimiento:	RF-03
Nombre del requerimiento:	Modificación
Características	Los administradores podrán modificar los datos de las personas en el sistema.
Descripción	Modificación de los datos personales de las personas, o incluso otorgarles el estatus de administrador
Requerimientos no funcionales:	RNF-01, RNF-02, RNF-04
Prioridad:	Media

Tabla 6

Requisito funcional "04"

Identificación del requerimiento:	RF-04
Nombre del requerimiento:	Consulta de información de acceso
Características	El sistema ofrecerá al administrador información general acerca de los ingresos generados
Descripción	Muestra los datos entre dos fechas
Requerimientos no funcionales:	RNF-01, RNF-02
Prioridad:	Baja

Tabla 7

Requisito funcional "05"

Identificación del requerimiento:	RF-05
Nombre del requerimiento:	Gestión de reportes
Características	El sistema permitirá la generación de reportes
Descripción	Permitir que el sistema genere de manera autónoma o bajo pedido de reportes mensuales sobre el acceso
Requerimientos no funcionales:	RNF-01, RNF-02
Prioridad:	Alta

Tabla 8

Requisito funcional "06"

Identificación del requerimiento:	RF-06
Nombre del requerimiento:	Control de Acceso
Características	Permitir acceso mediante reconocimiento facial
Descripción	El sistema permite el acceso a gente que se encuentra dentro del sistema, pero evitar el ingreso de terceros
Requerimientos no funcionales:	RNF-04, RNF-06
Prioridad:	Alta

2.2. Requerimientos No Funcionales

Tabla 9

Requisito no funcional "01"

Identificación del requerimiento:	RNF-01
Nombre del requerimiento:	Interfaz Gráfica de Usuario
Características	Interfaz sencilla y de fácil manejo
Descripción	El programa debe de tener una interfaz cuyo uso sea intuitiva y sencilla
Prioridad:	Alta

Tabla 10

Requisito no funcional "02"

Identificación del requerimiento:	RNF-02
Nombre del requerimiento:	Uso sencillo del sistema
Características	La interfaz gráfica de usuaria presentará un sistema de ayuda dentro de sus opciones para facilitar el manejo del programa y sistema
Descripción	La interfaz poseerá un sistema de ayudas para facilitar el aprendizaje de su uso
Prioridad:	Alta

Tabla 11

Requisito no funcional "03"

Identificación del requerimiento:	RNF-03
Nombre del requerimiento:	Diseño de la interfaz ESPAM-MFL
Características	Diseño similar a la web de la institución
Descripción	El programa debe de tener una interfaz cuyo diseño se parezca al de la web de la ESPAM-MFL
Prioridad:	Baja

Tabla 12

Requisito no funcional "04"

Identificación del requerimiento:	RNF-04
Nombre del requerimiento:	Desempeño óptimo
Características	Desempeño en cuanto a la confiabilidad de los datos almacenados
Descripción	La información almacenada (registros) pueden ser consultados y actualizados de manera simultánea sin que se afecte el tiempo de respuesta
Prioridad:	Alta

Tabla 13*Requisito no funcional "05"*

Identificación del requerimiento:	RNF-05
Nombre del requerimiento:	Confiabilidad continua
Características	Sistema funcional continuo
Descripción	Funcional las veinticuatro horas los siete días de la semana para procesar siempre los datos.
Prioridad:	Alta

Tabla 14*Requisito no funcional "06"*

Identificación del requerimiento:	RNF-06
Nombre del requerimiento:	Seguridad de información
Características	Información asegurada en los procesos del sistema
Descripción	Garantizar la seguridad del sistema en cuanto a información, contraseñas, datos personales y fechas de acceso
Prioridad:	Alta

ANEXO 3. INFORME SOBRE LOS COMPONENTES DE UN SISTEMA BIOMÉTRICO

En el presente informe se hablará de los componentes de un sistema biométrico, tanto físicos como de software, un sistema de estas características contiene un conjunto de componentes que se define como un sistema de enrolamiento que se encarga de capturar las biometrías de un usuario, validarlas, transformarlas y guardarlas para verificaciones posteriores (Mora & Sanz, 2021).

Usualmente se encuentra conformado por cuatro componentes principales, los cuales son: un sensor o módulo de adquisición para obtener los datos biométricos, un módulo extractor de características para reducir las características a datos comprensibles para el sistema, módulo de encaje para la búsqueda de patrones en los datos recibidos que encajen con los datos dentro del sistema y un módulo de toma de decisiones para ver si la entidad que accede al sistema es rechazada o aceptada (Obaidat et al, 2019) sin embargo la autora Nieves, en el año 2021, reduce estos componentes a tres mecanismos: autenticación, autorización y trazabilidad. Sin embargo, los autores Cedeño y Párraga en 2019 divide los componentes necesarios en dos partes:

- Software: Una aplicación que una al cliente con el servidor y una base de datos
- Hardware: Una placa, usualmente un microcontrolador, que funciona como puente entre el resto del sistema físico y el software; el módulo de adquisición o sensor, un cable de conexión del estándar necesario para la placa, relays y finalmente algún tipo de cerradura eléctrica

Finalmente, tras los datos recopilados se concluye que los componentes necesarios para lograr el sistema biométrico es un sistema que capte los datos, usualmente de manera única pero también los hay de manera continua, un sistema que lleve la información al sistema, una base de datos que consulte el sistema, y una cerradura electrónica que reciba una señal del sistema para permitir el acceso

BIBLIOGRAFÍA

Cedeño Navarrete, J. R., & Párraga Vera, C. L. (2017). Sistema biométrico de control de acceso para el laboratorio de cómputo de la Unidad Educativa Francisco González Álava (Bachelor's thesis, Calceta: Espam).

Nieves Villamar, K. X. (2021). Restructuración de un Sistema Biométrico para fortalecer la seguridad y control de acceso en el decanato de la Facultad De Ciencias Técnicas (Bachelor's thesis, Jipijapa. UNESUM).

Mora, M. M. C., & Sanz, C. J. (2021). SISTEMAS BIOMÉTRICOS: ¿SEGURIDAD O PRIVACIDAD? En Aspectos sociales en la seguridad ciudadana (pp. 147–158). Dykinson.

Obaidat, M. S., Rana, S. P., Maitra, T., Giri, D., & Dutta, S. (2019). Biometric security and internet of things (IoT). *Biometric-based physical and cybersecurity systems*, 477-509.

ANEXO 4. FOTOS DE LAS PRUEBAS DE LA INTELIGENCIA ARTIFICIAL

Imagen 9

Reconocimiento facial de uno de los autores

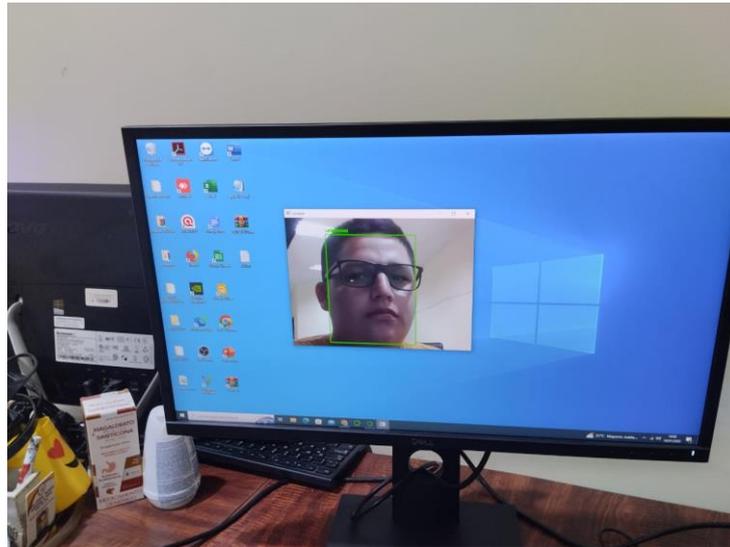


Imagen 10

Reconocimiento facial del tutor de tesis



Imagen 11

Tutor de tesis probando el reconocimiento facial

**Imagen 12**

Reconocimiento facial del programador y autor de tesis



Imagen 13

Reconocimiento facial del tutor de tesis



ANEXO 5. INFORME DE PROGRAMACIÓN DE LA INTELIGENCIA ARTIFICIAL

Imagen 14

Inicio de la prueba

```
In [7]: # Detectar si se dispone de GPU cuda
# =====
device = torch.device('cuda:0' if torch.cuda.is_available() else 'cpu')
print(f'Running on device: {device}')

# Crear diccionario de referencia para cada persona
# =====
dic_referencias = crear_diccionario_referencias(
    folder_path = '../data/test_images',
    min_face_size = 40,
    min_confidence = 0.9,
    device = device,
    verbose = True
)

Running on device: cpu
Identidad: angelina_jolie --- Imágenes referencia: 1
Identidad: BelenVillao --- Imágenes referencia: 200
Identidad: bradley_cooper --- Imágenes referencia: 1
Identidad: edwin --- Imágenes referencia: 1
Identidad: julioPonce --- Imágenes referencia: 200
Identidad: kate_siegel --- Imágenes referencia: 1
Identidad: paul_rudd --- Imágenes referencia: 1
Identidad: shea_whigham --- Imágenes referencia: 1
```

Imagen 15

Detección mediante la webcam

```
In [4]: import cv2
```

```
In [5]: pipeline_deteccion_webcam(
    dic_referencia = dic_referencias,
    threshold_similaridad = 0.4,
)
```

```
In [ ]:
```

ANEXO 6. ESTRUCTURA DEL CÓDIGO

6.1. CÓDIGO PRINCIPAL PARA LA CÁMARA

```
#Acceso y configuración de la puerta
```

```
RELAY = 17
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BCM)
GPIO.setup(RELAY, GPIO.OUT)
GPIO.output(RELAY,GPIO.LOW)
```

```
#Dirección de acceso las imágenes para reconocimiento
```

```
url='/home/udc/Desktop/TesisFinal/data/images'
```

```
#Definición de variables globales y dispositivo de captura de imagen
```

```
cap = cv2.VideoCapture('/dev/video0')
flag=0
nombre_temp=0
actual1 = time.time_ns()
```

```
#Verificación de la disponibilidad del dispositivo de captura de imagen, de no encontrarse disponible el programa se cierra
```

```
if not cap.isOpened():
    print("No se puede acceder a la camara")
    exit()
```

```
#Inicialización del código librería, véase Anexo 6.2
```

```
objeto = FaceDeteccion.FaceDeteccion(url)
known_face_encodings, known_face_names = objeto.get_faceEncodings()
print('System Ready')
```

```
#Inicio del bucle principal
```

```
while cap.isOpened():
    frame_exist, frame = cap.read()
```

```
nombres = objeto.detect_faces(frame, known_face_encodings,
known_face_names)
```

#Si se detectan caras se genera una dirección que en caso de no existir para guardar la informacion

```
if len(nombres)>0:
```

```
    current_datetime = datetime.now()
```

```
    date_only = current_datetime.date()
```

```
    month_only = current_datetime.month
```

```
    directory_path
```

```
=
```

```
"/home/udc/Desktop/TesisFinal/reports/DailyLog/{}/".format(month_only)
```

```
if not os.path.exists(directory_path):
```

```
    os.makedirs(directory_path)
```

```
file_path = os.path.join(directory_path, "{}-log.txt".format(date_only))
```

```
if flag==0:
```

```
    print(nombres)
```

```
    with open(file_path, 'a') as file:
```

```
        for nombre in nombres:
```

```
            print(nombre)
```

```
            file.write("{} -- {} \n".format(nombre, current_datetime))
```

#Tras guardar los datos de manera satisfactoria se envía la señal a la puerta para cortar momentáneamente la corriente y permitir el acceso

```
GPIO.output(RELAY,GPIO.HIGH)
```

```
time.sleep(2)
```

```
GPIO.output(RELAY,GPIO.LOW)
```

```
nombre_temp=nombres
```

```
actual1 = time.time_ns()
```

```
flag=1
```

```
else:
```

```
if nombres!=nombre_temp and nombres!=["Unknown"]:
```

```
    print(nombres)
```

```
    with open(file_path, 'a') as file:
```

```

    for nombre in nombres:
        print(nombre)
        file.write("{} -- {} \n".format(nombre, current_datetime))
GPIO.output(RELAY,GPIO.HIGH)
time.sleep(1)
GPIO.output(RELAY,GPIO.LOW)
nombre_temp=nombres
nombres=[]
flag=0

```

#Revalorización de variables

```

    actual2 = time.time_ns()
    if((actual2-actual1)/1e9) > 10:
        nombre_temp=[]
        actual1 = time.time_ns()

```

6.2. CÓDIGO LIBRERÍA PARA EL RECONOCIMIENTO FACIAL

#Definición de la clase

class FaceDeteccion:

```

    images_url = "
    def __init__(self, samples_url):
        self.images_url = samples_url

```

#Función para obtener los encodings e identidades

```

    def get_faceEncodings(self):
        folders = glob.glob(self.images_url + "/*")
        known_face_encodings=[]
        known_face_names=[]
        for folder in folders:
            if platform.system() in ['Linux', 'Darwin']:
                identidad = folder.split("/")[-1]

```

```

else:
    identidad = folder.split("\\")[-1]
    path_imagenes = glob.glob(folder + "/*.jpg")
    path_imagenes.extend(glob.glob(folder + "/*.jpeg"))
    path_imagenes.extend(glob.glob(folder + "/*.tif"))
    path_imagenes.extend(glob.glob(folder + "/*.png"))
    subject_image = face_recognition.load_image_file(path_imagenes[0])
    face_encoding = face_recognition.face_encodings(subject_image)[0]
    known_face_encodings.append(face_encoding)
    known_face_names.append(folder.split('/')[-1])
return known_face_encodings, known_face_names

```

#Función para detectar y reconocer los rostros, en caso de no registrar la cara se devuelve el nombre "Unknown" o desconocido, sin embargo, no refleja en los datos finales

```

def detect_faces(self, frame, known_face_encodings, known_face_names):
    small_frame = cv2.resize(frame,(0,0),None, 0.25, 0.25)
    rgb_small_frame = cv2.cvtColor(small_frame, cv2.COLOR_BGR2RGB)
    face_locations = face_recognition.face_locations(rgb_small_frame)
    face_encodings = face_recognition.face_encodings(rgb_small_frame,
    face_locations)
    face_names = []
    for face_encoding in face_encodings:
        matches = face_recognition.compare_faces(known_face_encodings,
    face_encoding)
        name = "Unknown"
        face_distances = face_recognition.face_distance(known_face_encodings,
    face_encoding)
        best_match_index = np.argmin(face_distances)
        if (face_distances[best_match_index]<=0.45):
            if matches[best_match_index]:

```

```
        name = known_face_names[best_match_index]
    face_names.append(name)
return face_names
```

ANEXO 7. SERVICIO DAEMON

#Descripción general del servicio

[Unit]

Description=Reconocimiento Facial

After=network.target

#Definición de las características del servicio, entre ellos podemos encontrar que siempre se reiniciará el servicio en caso de caerse, así como la dirección desde donde proviene el código a ejecutar

[Service]

Type=simple

Environment="DISPLAY=:0"

ExecStart=/bin/lxterminal -e "sudo
/usr/bin/python3/home/udc/Desktop/TesisFinal/main/feedcamara.py"

Restart=always

User=udc

Group=udc

#En que perfiles se ejecuta el servicio

[Install]

WantedBy=multi-user.target

ANEXO 8. ESTRUCTURA DEL CABLEADO

Imagen 16

Caja de cables donde se encuentra la conexión a la puerta con cerradura electromagnética

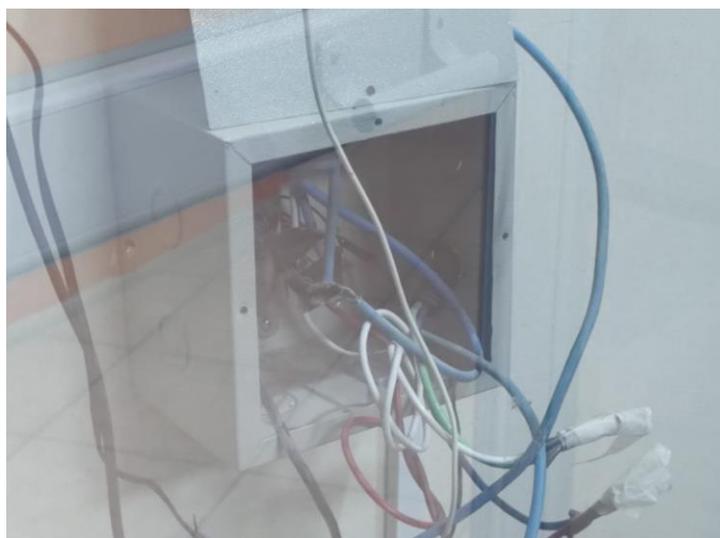


Imagen 17

Toma a corriente de la cerradura



Imagen 18

Cerradura electromagnética



ANEXO 9. ARCHIVOS DE BITÁCORA GENERADOS

11.1. IMÁGENES

Imagen 19

Carpetas donde se guardan los archivos, enumeradas por mes

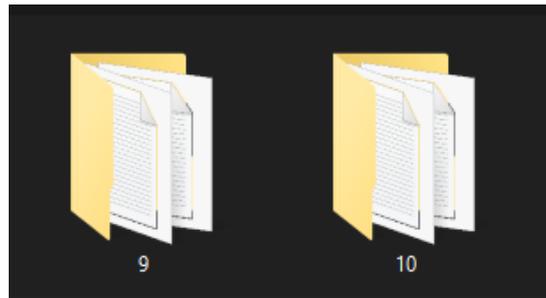
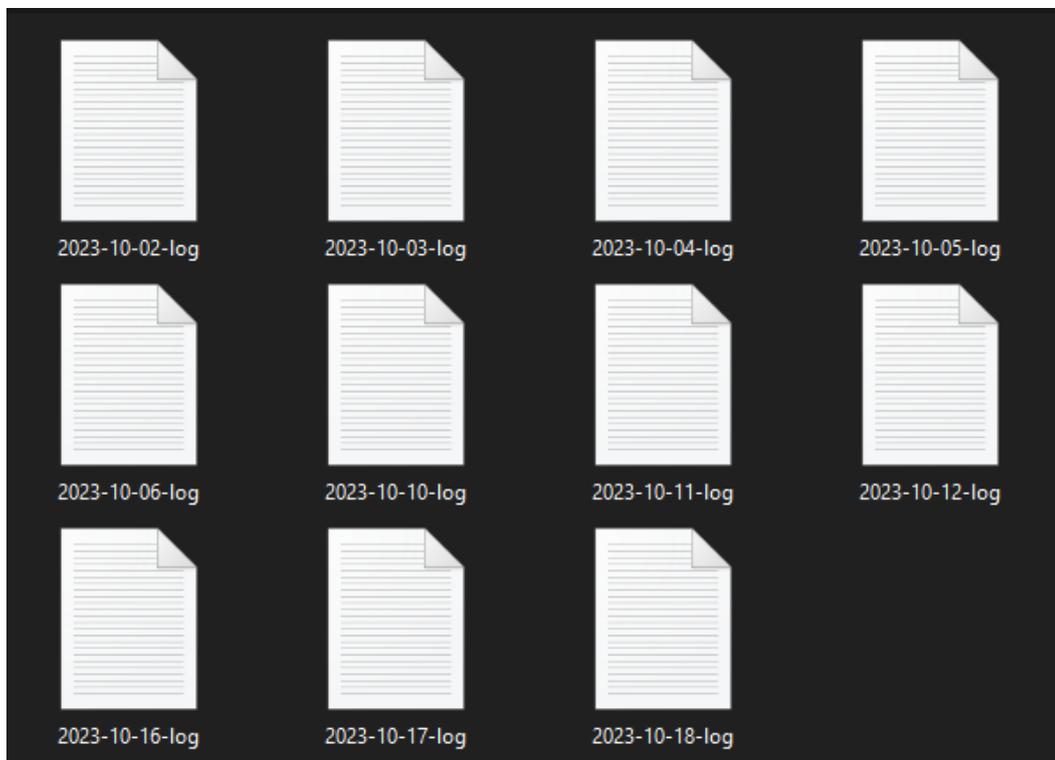


Imagen 20

Archivos generados



11.2. EJEMPLO DE BITÁCORA GENERADA

"jimmy_loor -- 2023-10-03 08:15:24.697252
jimmy_loor -- 2023-10-03 08:27:24.319572
victor_pinargote -- 2023-10-03 08:27:51.010164
victor_pinargote -- 2023-10-03 08:30:48.257056
victor_pinargote -- 2023-10-03 08:46:00.202489
edwin_vera -- 2023-10-03 08:46:22.295165
victor_pinargote -- 2023-10-03 08:57:29.386982
victor_pinargote -- 2023-10-03 09:09:27.286694
victor_pinargote -- 2023-10-03 09:32:40.693071
victor_pinargote -- 2023-10-03 09:33:12.409090
isabel_villamar -- 2023-10-03 09:39:07.645314
isabel_villamar -- 2023-10-03 09:39:08.750774
victor_pinargote -- 2023-10-03 09:39:47.564347
jimmy_loor -- 2023-10-03 09:42:46.518952
edwin_vera -- 2023-10-03 09:45:37.133202
victor_pinargote -- 2023-10-03 10:06:39.706205
manuel_macias -- 2023-10-03 10:06:49.423957
william_rivera -- 2023-10-03 10:07:03.008220
victor_pinargote -- 2023-10-03 10:07:08.322961
victor_pinargote -- 2023-10-03 10:07:48.294924
jimmy_loor -- 2023-10-03 10:07:57.292933
victor_pinargote -- 2023-10-03 10:40:08.156285
victor_pinargote -- 2023-10-03 10:40:36.863521
victor_pinargote -- 2023-10-03 10:40:38.086625
jimmy_loor -- 2023-10-03 10:47:31.774129
william_rivera -- 2023-10-03 10:50:48.562084
manuel_macias -- 2023-10-03 10:51:21.936612
edwin_vera -- 2023-10-03 12:09:06.546165

edwin_vera -- 2023-10-03 12:09:07.705239
joffre_moreira -- 2023-10-03 12:09:48.889103
joffre_moreira -- 2023-10-03 12:09:50.004201
jimmy_loor -- 2023-10-03 12:14:18.774906
jimmy_loor -- 2023-10-03 12:14:19.893048
edwin_vera -- 2023-10-03 12:26:00.115929
edwin_vera -- 2023-10-03 12:26:01.234085
belen_villao -- 2023-10-03 12:37:15.802174
belen_villao -- 2023-10-03 12:37:16.916281
edwin_vera -- 2023-10-03 12:37:41.387770
edwin_vera -- 2023-10-03 12:37:42.501350
manuel_macias -- 2023-10-03 12:43:39.572902
manuel_macias -- 2023-10-03 12:43:40.729070
edwin_vera -- 2023-10-03 13:14:26.542521
edwin_vera -- 2023-10-03 13:14:27.662950
belen_villao -- 2023-10-03 13:44:09.478382
belen_villao -- 2023-10-03 13:44:10.722265
jimmy_loor -- 2023-10-03 13:51:39.916355
jimmy_loor -- 2023-10-03 13:51:41.052051
victor_pinargote -- 2023-10-03 14:24:15.538704
victor_pinargote -- 2023-10-03 14:24:16.655985
jimmy_loor -- 2023-10-03 15:12:17.993869
jimmy_loor -- 2023-10-03 15:12:19.180846
manuel_macias -- 2023-10-03 15:19:25.495711
manuel_macias -- 2023-10-03 15:19:26.616311
jimmy_loor -- 2023-10-03 15:42:11.858913
jimmy_loor -- 2023-10-03 15:42:13.148709
manuel_macias -- 2023-10-03 16:11:41.574340
manuel_macias -- 2023-10-03 16:11:43.020354
ligia_zambrano -- 2023-10-03 16:39:59.617950

belen_villao -- 2023-10-03 16:44:55.281853

belen_villao -- 2023-10-03 16:48:28.342891

isabel_villamar -- 2023-10-03 16:53:10.499886

manuel_macias -- 2023-10-03 18:17:28.315841

manuel_macias -- 2023-10-03 18:17:29.492798“