Información General del Semillero de Investigación		
Facultad: **Ciencias Naturales e Ingeniería FCNI**		
Programa académico: **Tecnología en Gestión de Sistemas de Telecomunicaciones**	Grupo(s) de Investigación: **GNET**	
Nombre del semillero – Siglas **Semillero de investigación en Sistemas de Telecomunicaciones - SISTEL**		
Líneas de Investigación: **Programación en sistemas de comunicaciones**		

Áreas del saber *			
	1. Agronomía veterinaria y afines		5. Ciencias sociales y humanas
	2. Bellas artes		6. Economía, administración, contaduría y afines
	3. Ciencias de la educación		7. Matemáticas y ciencias naturales
	4. Ciencias de la salud	x	8. Ingenierías, arquitectura, urbanismo y afines

Al diligenciar este documento autorizo a UNIDADES TECNOLÓGICAS DE SANTANDER, ubicada en Calle de los Estudiantes 9-82 Ciudadela Real de Minas y con teléfono de contacto 6076917700, para que recolecte, almacene, use, circule y/o suprima mis datos personales. Lo anterior para dar cumplimiento a las finalidades incorporadas en la Política de Tratamiento de Información disponible en www.uts.edu.co, la cual declaro conocer y saber que en esta se especifican cuáles datos son sensibles. Así mismo, conozco que como titular me asisten los derechos a conocer, actualizar, rectificar y suprimir mis datos y revocar la autorización. Igualmente declaro que poseo autorización, de los otros titulares de datos que suministro, para que UNIDADES TECNOLÓGICAS DE SANTANDER les dé tratamiento conforme a las finalidades consignadas en la Política.

Información del Docente Líder

Nombre	Documento	Correo electrónico
Johan Leandro Téllez Garzón	1101753049	jtellez@correo.uts.edu.co

Información de los Autores

Nombre	Documento	Correo electrónico
Andrea Tatiana Acuña Prada	.1096959406	atacuna@uts.edu.co
Angie Alejandra Cadena Lizarazo	1005540911	aalejandracadena@uts.edu.co

Proyecto

1. Título del proyecto: Desarrollo de un prototipo de cerradura biométrica basada en reconocimiento de iris para una oficina del programa de Telecomunicaciones de las UTS	MODALIDAD DEL PROYECTO **			
	PA	PI	TI	Otra. ¿Cuál?
			x	
	Fecha de terminación del proyecto:			5/02/2024

2. Resumen del trabajo:

El prototipo de cerradura biométrica basada en el reconocimiento de iris, destinado a mejorar la seguridad y el control de acceso en una oficina del programa de Telecomunicaciones de las Unidades Tecnológicas de Santander (UTS).

El objetivo principal del proyecto es diseñar e implementar un sistema que utilice procesamiento de imágenes y algoritmos de reconocimiento biométrico para verificar la identidad de los usuarios. La solución tecnológica se basa en una Raspberry Pi 5, junto con una cámara especializada y técnicas avanzadas de análisis de patrones de iris, garantizando una identificación segura y eficiente.

La metodología empleada abarcó la captura y preprocesamiento de imágenes, la segmentación del iris, la extracción de características mediante filtros de Gabor y la verificación de identidad con la métrica de distancia de Hamming. Para la implementación, se utilizaron herramientas de desarrollo como Python, OpenCV y bases de datos embebidas, permitiendo una integración óptima entre hardware y software.

Los resultados obtenidos demostraron que el sistema tiene un alto nivel de precisión en la autenticación de usuarios, con una tasa reducida de falsos positivos y negativos. La validación del prototipo evidenció la viabilidad de aplicar este tipo de tecnología en entornos académicos y empresariales como una alternativa segura y eficiente a los métodos tradicionales de control de acceso.

Este proyecto no solo contribuye a la seguridad en los espacios físicos, sino que también fomenta el desarrollo e implementación de tecnologías biométricas en aplicaciones prácticas. Como trabajo futuro, se propone la optimización del algoritmo de reconocimiento, la integración con sistemas de gestión de acceso y la exploración de nuevas tecnologías de aprendizaje profundo para mejorar el rendimiento del sistema.

3. Objetivo general y objetivos específicos:

Desarrollar un prototipo de cerradura biométrica basada en reconocimiento de iris, integrado en una Raspberry Pi y conectado a un servidor local para el almacenamiento de datos biométricos, con el fin de controlar el acceso a una oficina del programa de telecomunicaciones de las UTS.

- Diseñar un algoritmo de reconocimiento de iris, basado en procesamiento de imágenes, para identificar y verificar usuarios que acceden a una de las oficinas del Programa de Telecomunicaciones de las UTS.
- Implementar el prototipo de cerradura biométrica utilizando una Raspberry Pi para la comunicación entre hardware y software, garantizando el funcionamiento autónomo del sistema de control de acceso.
- Establecer una conexión segura entre el prototipo de reconocimiento biométrico y un servidor local, utilizando protocolos de seguridad adecuados para garantizar la integridad de la información.
- Realizar pruebas exhaustivas del prototipo en situaciones simuladas y reales, evaluando su precisión, velocidad de respuesta y resistencia a posibles ataques o interferencias, con el fin de garantizar su viabilidad y efectividad como alternativa para el control de acceso a una de las oficinas del Programa de Telecomunicaciones de las UTS.

4. Análisis de resultados:

El prototipo de cerradura biométrica desarrollado integra diferentes etapas que garantizan un proceso completo y seguro de autenticación de usuarios. Dicho sistema inicia con la captura de la imagen del iris mediante la cámara Raspberry Pi, seguido del procesamiento a través de algoritmos de reconocimiento implementados en OpenCV. Posteriormente, se genera una señal de validación que activa la cerradura eléctrica en caso de que el usuario esté autorizado, complementándose con una plataforma web que centraliza el registro y control de accesos. Todo este flujo de funcionamiento se encuentra ilustrado en la Figura:



El desarrollo del sistema de reconocimiento de iris en la RaspberryPi 5 se llevó a cabo siguiendo una metodología estructurada que abarcó las siguientes etapas principales:

- Adquisición de imágenes de iris: Se utilizó el sensor de cámara Raspberry Pi V2 de 8MP para capturar imágenes del iris. Se desarrolló un script en Python con la librería OpenCV para controlar la cámara y ajustar enfoque, exposición y balance de blancos, buscando obtener imágenes de alta calidad y reducir el ruido.
- Preprocesamiento de imágenes: Se aplicaron técnicas para mejorar la calidad de las imágenes y facilitar las etapas posteriores. Esto incluyó eliminación de ruido con filtros de suavizado como el gaussiano y normalización para ajustar tamaño y orientación de las imágenes, asegurando un formato consistente para la comparación.(Zadnik & Žemva, 2021)
- Segmentación del iris y extracción de características: Se identificó y aisló la región del iris. Se implementó un algoritmo basado en transformadas de Hough para detectar bordes circulares del iris y la pupila. Luego, se aplicó un filtro de Gabor al iris segmentado para extraer características texturales únicas, almacenadas como un vector numérico.(Orozco-Rosas et al., 2012)
- Reconocimiento y comparación: Se utilizó el algoritmo de distancia de Hamming para comparar los vectores de características del iris capturado con los de la base de datos, calculando un puntaje de similitud. Si este puntaje superaba un umbral, se consideraba una coincidencia.(ARROYAVE GIRALDO, 2012) y (Coomonte Belmonte, 2006)
- Pruebas y validación: Se realizaron pruebas utilizando una base de datos propia compuesta por imágenes de iris capturadas a partir de los usuarios registrados en el sistema, con el fin de evaluar su precisión y rendimiento. Esta base de datos incluye múltiples imágenes por individuo tomadas en diferentes condiciones de iluminación y posición, lo que permitió analizar el comportamiento del sistema ante variaciones reales y medir su capacidad de identificación con mayor fiabilidad. Se calcularon métricas como la tasa de falsos positivos y la tasade falsos negativos. Se ajustaron parámetros como el umbral de similitud para optimizar el sistema.

.

1. Configuración e implementación del aplicativo web

1.1. Preparación del entorno

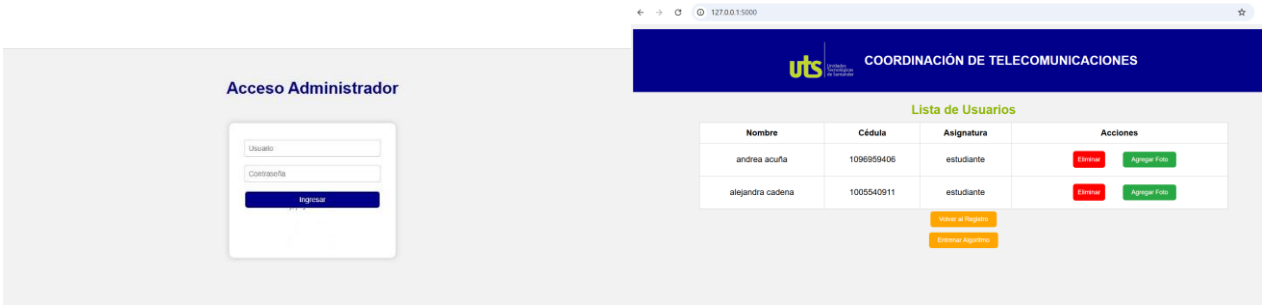
Se configuró una base de datos para la gestión de usuarios y un entorno gráfico web para la interacción con la aplicación. Se creó un entorno virtual para aislar dependencias y garantizar estabilidad en la ejecución. Se verificó la versión de Python, se generaron carpetas para usuarios y se habilitaron las funciones básicas de autenticación. Una vez completada la configuración inicial, se instaló la aplicación y se validó su funcionamiento a través del entorno web, confirmando la correcta gestión de usuarios registrados.

1.2. Interfaz web

La interfaz de acceso incluye un login de administrador, asegurando control exclusivo del sistema. El panel principal permite:

- Registrar usuarios (nombre, cédula, asignatura).
- Visualizar usuarios.
- Agregar fotografías.
- Entrenar el algoritmo biométrico.
- Descargar reportes.
- Cerrar sesión.

La vista de usuarios muestra registros en tabla con opciones de eliminación y gestión de fotos.



2. Entrenamiento del sistema de reconocimiento de iris

2.1. Organización del entorno

En la Raspberry Pi se estructuraron carpetas dedicadas para: entrenamientos, pruebas y registros del sistema. Esta organización permitió mantener un flujo ordenado de datos para el algoritmo.

2.2. Proceso de entrenamiento

El código (Anexo D y E) permitió cargar imágenes y entrenar la red. Durante el entrenamiento:

- El sistema activa un LED cuando el iris es reconocido correctamente.
- También activa un relé, simulando un sistema de apertura (cerradura electrónica).
- Se validó la integración entre hardware y software mediante pruebas reales.



3. Validación del funcionamiento

3.1. Pruebas con usuarios registrados

Tabla 1 Pruebas de ingreso individuo 1

prueba ingreso individuo 1			
intentos	positivo	negativo	tiempo de respuesta (s)
1	x		17,39
2	x		2,35
3	x		4,19
4	x		8,50
5	x		13,90
6	x		4,72
7	x		7,18
8	x		5,10
9	x		5,42
10	x		4,12
promedio de reconocimiento			7,287 segundos

Los promedios de reconocimiento permiten evaluar el desempeño general del sistema para cada individuo. Por ejemplo, en la Tabla 1, el individuo 1 alcanzó un tiempo promedio de respuesta de 7,287 segundos, mientras que en la Tabla 4 el

individuo 2 presentó un promedio mayor (20,751 segundos), lo que indica que factores como la iluminación, la calidad de la captura o las características del iris influyen en el tiempo de procesamiento.

Tabla 2 Pruebas con fotos individuo 1

prueba con fotos individuo 1			
intentos	positovo	negativo	tiempo de respuesta (s)
11		x	
12	x		3,99
13	x		3,97
14		x	
15	x		7

Tabla 3 Pruebas utilizando objetos de uso común individuo 1

prueba utilizando gafas			
intentos	positovo	negativo	tiempo de respuesta (s)
16	x		16,66
17		x	
18	x		9,52
19		x	
20		x	

Tabla 4 pruebas de ingreso individuo 2

prueba ingreso individuo 2			
intentos	positivo	negativo	tiempo de respuesta (s)
1	x		46,03
2	x		27,88
3	x		21,40
4	x		8,18
5	x		15,96
6	x		12,72
7	x		6,23
8	x		30,14
9	x		21,63
10	x		17,34
promedio de reconocimiento			20,751

Las pruebas con fotografías (Tablas 2 y 5) y con objetos de uso común como gafas (Tablas 3 y 6) se realizaron para verificar la robustez del sistema frente a intentos de suplantación o condiciones que dificultan la captura del iris. En estos casos, se evidenció que el sistema logra en varios intentos identificar el ingreso como inválido (negativo), aportando seguridad al evitar accesos no autorizados.

Tabla 5 Pruebas con fotos de individuo 2

prueba con fotos individuo 2			
intentos	positovo	negativo	tiempo de respuesta (s)
11	x		35,45
12		x	
13	x		8,84
14	x		9,39
15	x		6,75

Tabla 6 Pruebas usando objetos de uso común individuo 2

prueba utilizando gafas			
intentos	positovo	negativo	tiempo de respuesta (s)
16		x	
17		x	
18		x	
19		x	
20		x	

prueba ingreso no registrado			
intentos	positivo	negativo	Tiempo de respuesta (s)
1	x		
2	x		
3		x	
4		x	
5	x		8,63
6	x		3.32
7		x	
8		x	
9	x		3,80
10	x		5,68
11		x	

12	x		9,66
13		x	
14		x	
15		x	
16		x	
17	x		6,70
18	x		10,34
19	x		6,83
20		x	

Finalmente, en la prueba de ingreso de un individuo no registrado, se observan varios intentos en los que el sistema marcó positivo de manera errónea, pero también se registraron respuestas negativas correctas. Estos resultados resaltan la necesidad de seguir optimizando los algoritmos de reconocimiento para reducir la tasa de falsos positivos y garantizar una mayor precisión en la validación.

4. Análisis general de desempeño

Fortalezas identificadas

- El sistema reconoce de manera consistente a usuarios registrados.
- Interfaz web completa y funcional para la administración.
- Integración efectiva hardware–software.
- Buen manejo de información biométrica y estructura de almacenamiento.
- Capacidad para detectar algunos intentos de suplantación mediante fotos u objetos.

Limitaciones detectadas

- Tiempos variables de respuesta, afectados por condiciones ambientales y características morfológicas.
- Falsos positivos en pruebas con individuos no registrados y en casos puntuales de suplantación.
- Sensibilidad a elementos obstructores (gafas), que puede afectar exactitud.
- Oportunidades de mejora
- Optimización del algoritmo para aumentar precisión.
- Ampliación del dataset de entrenamiento para mejorar generalización.
- Implementación de filtros adicionales para detección de vida (anti-spoofing).
- Ajustes en iluminación y óptica para mejorar calidad de la captura.

5. Ensamblaje físico del prototipo

El ensamblaje físico del prototipo se realizó integrando una Raspberry Pi dentro de una caja plástica protectora, sobre una base perforada que permitió organizar y fijar los componentes de manera segura. A la tarjeta se conectó el módulo de cámara oficial mediante el puerto CSI utilizando un cable plano tipo flex, el cual fue instalado en la tapa de la caja para permitir la captura directa del iris sin exponer la placa. Adicionalmente, se incorporó un sistema de ventilación compuesto por un ventilador y disipador para evitar el sobrecalentamiento durante el procesamiento, así como una fuente de alimentación oficial de 27 W que garantiza un suministro estable de energía. El prototipo también incluye conexiones auxiliares, como un LED de verificación y el sistema de relé, quedando todos los elementos ordenados dentro de la caja, lo que facilita su mantenimiento y asegura un funcionamiento fiable del sistema biométrico.



El sistema desarrollado demuestra funcionalidad operativa en las etapas de registro, entrenamiento y validación del reconocimiento de iris. El aplicativo web ofrece una plataforma sólida para la gestión de usuarios y el control del sistema. Las pruebas indican un buen desempeño en el reconocimiento correcto de usuarios registrados, aunque se identifican áreas críticas a mejorar en términos de robustez frente a intentos de suplantación y reducción de falsos positivos.

El prototipo físico está correctamente ensamblado y permite una integración estable entre software y hardware, sentando las bases para mejoras futuras en precisión, seguridad y rendimiento.

5. Conclusiones:

El proyecto logró implementar un prototipo funcional de reconocimiento de iris integrando hardware y software, utilizando una Raspberry Pi 5 y su cámara oficial. Las técnicas empleadas —preprocesamiento con filtrado gaussiano, transformada de Hough, filtros de Gabor y métrica de Hamming— permitieron una segmentación y autenticación biométrica operativa. Sin embargo, el sistema presentó limitaciones bajo variaciones de iluminación y en usuarios con iris muy pigmentados, lo que aumentó las tasas de falsa aceptación (FAR) y falso rechazo (FRR). Además, la resolución limitada del sensor de la cámara afectó la calidad de la captura y redujo la precisión en la extracción de características. En conjunto, los resultados demuestran un prototipo funcional, pero aún distante del desempeño de sistemas comerciales, por lo que futuras versiones deberán reforzar tanto algoritmos como hardware para mejorar la confiabilidad y robustez.

6. Recomendaciones:

Se sugiere mejorar la calidad del sistema incorporando cámaras de mayor resolución, ópticas superiores e iluminación infrarroja. También se recomienda explorar hardware acelerador y técnicas avanzadas como super-resolución, segmentación profunda y detección de vitalidad. En el ámbito del software, es conveniente optimizar el procesamiento con herramientas más eficientes, diseñar una API RESTful para integración externa e incluir una interfaz de calibración. Finalmente, se aconseja reforzar la seguridad y privacidad de la base de datos biométrica e investigar métodos de machine unlearning para permitir el borrado responsable de datos.

7. Bibliografía:

Agullo Quiros, C. (2024). Reconocimiento Biométrico Ocular Basado en Deep Learning.

Alegre, E., & Fernánez-Robles, L. (2010). CAPÍTULO 8 SIFT (SCALE INVARIANT FEATURE TRANSFORM).

Alexander, J., Osorio, C., Francisco, M. S., Aguirre, A. M., & Escobar, J. A. M. (2010). Sistemas de seguridad basados en Biometría. Scientia et Technica, 3(46), 98–102. https://doi.org/10.22517/23447214.245

Al-Waisy, A. S., Qahwaji, R., Ipson, S., Al-Fahdawi, S., & Nagem, T. A. M. (2018a). A multi-biometric iris recognition system based on a deep learning approach. Pattern Analysis and Applications, 21(3), 783–802. https://doi.org/10.1007/S10044-017-0656-1/TABLES/8

Al-Waisy, A. S., Qahwaji, R., Ipson, S., Al-Fahdawi, S., & Nagem, T. A. M. (2018b). A multi-biometric iris recognition system based on a deep learning approach. Pattern Analysis and Applications, 21(3), 783–802. https://doi.org/10.1007/S10044-017-0656-1/TABLES/8

American Academy of Ophthalmology. (2020). Eye anatomy: Parts of the eye. https://www.aao.org/eye-health/anatomy

Aníbal, A., Moreno, V., Leonell, F., Reyes, M., & Mendoza De Los Santos, A. (2022). Benefits of biometric systems based on Iris reading. Perfiles de Ingeniería, 18(18), 51–66. https://doi.org/10.31381/PERFILESINGENIERIA.V18I18.5398

ARROYAVE GIRALDO, M. (2012). DETERMINACION DE LA INCIDENCIA DEL CALCULO DE LA ROI EN UN SISTEMA DE IDENTIFICACION BIOMETRICA DE LA PALMA DE LA MANO MEDIANTE LBP.

Avigilon.com. (2025). Guía de control de acceso biométrico y cerraduras seguras. https://www.avigilon.com/es/blog/biometric-access-control

aws.amazon.com. (2025). ¿Qué es Python? - Explicación del lenguaje Python - AWS. https://aws.amazon.com/es/what-is/python/

Chollet, F. (2018). Deep Learning with Python.

Coomonte Belmonte, R. (2006). Sistema de reconocimiento de personas mediante su patrón de iris basado en la transformada wavelet. 1–267.

Daugman, J. (1985). Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. Journal of the Optical Society of America A, 2(7), 1160-1169. https://doi.org/10.1364/JOSAA.2.001160

Daugman, J. (2004). How iris recognition works. IEEE Transactions on Circuits and Systems for Video Technology, 14(1), 21–30. https://doi.org/10.1109/TCSVT.2003.818350

Detección de caras con Python usando OpenCV Tutorial | DataCamp. (s/f). Recuperado el 22 de septiembre de 2025, de https://www.datacamp.com/es/tutorial/face-detection-python-opencv

Duda, R. O., & Hart, P. E. (1972). Use of the Hough Transformation to detect lines and curves in pictures. Communications of the ACM, 15(1), 11–15. https://doi.org/10.1145/361237.361242

Forrester, J. V., Dick, A. D., McMenamin, P. G., Roberts, F., & Pearlman, E. (2016). The Eye: Basic Sciences in Practice (4th ed.). Elsevier Health Sciences.

Gangwar, A., & Joshi, A. (2016). DeepIrisNet: Deep iris representation with applications in iris recognition and cross-sensor iris recognition. International Conference on Information Photonics, 2016-August, 2301–2305. https://doi.org/10.1109/ICIP.2016.7532769

GARCIA GUILLEN, D. M. (2011). CARACTERIZACION Y EXTRACCION DE CARACTERES DEL IRIS PARA APLICACIONES DE RECONOCIMIENTO.

Garcia Mireya, & Ramirez Alejandro. (2007). Avances en el reconocimiento del iris: perspectivas y oportunidades en la investigación de algoritmos biométricos. https://www.scielo.org.mx/scielo.php?pid=S1405-55462012000300002&script=sci_abstract

Grinberg, Miguel. (2018). Flask Web Development, 2nd Edition. 312. https://www.oreilly.com/library/view/flask-web-development/9781491991725/

H. Hamd, M., & K. Ahmed, S. (2018). Biometric System Design for Iris Recognition Using Intelligent Algorithms. International Journal of Modern Education and Computer Science, 10(3), 9–16. https://doi.org/10.5815/IJMECS.2018.03.02

Hamming, R. W. (1950). Error detecting and error correcting codes. Bell System Technical Journal, 29(2), 147–160. https://doi.org/10.1002/j.1538-7305.1950.tb00463.x

Jetson Nano Lleva el Poder de la IA Moderna a los Dispositivos en el Edge | NVIDIA. (2025). https://www.nvidia.com/es-la/autonomous-machines/embedded-systems/jetson-nano/product-development/

Jhamb, M., & Khera, V. K. (2010). Iris based human recognition system. 2010 IEEE International Conference on Computational Intelligence and Computing Research, 232–240. https://doi.org/10.1109/ICCIC.2010.5705766

Kaur, G., & Singh, M. (2021). Web-based biometric authentication system using iris recognition. International Journal of Computer Applications, 183(44), 1–7. https://doi.org/10.5120/ijca2021921734

Kumar, A., & Passi, A. (2010a). Comparison and combination of iris matchers for reliable personal authentication. Pattern Recognition, 43(3), 1016–1026. https://doi.org/10.1016/J.PATCOG.2009.08.016

Kumar, A., & Passi, A. (2010b). Comparison and combination of iris matchers for reliable personal authentication. Pattern Recognition, 43(3), 1016–1026. https://doi.org/10.1016/J.PATCOG.2009.08.016

Lengerke, O. (2020). PROYECTO EDUCATIVO INSTITUCIONAL.

Ley 1581 de 2012 - Gestor Normativo - Función Pública. (2025). https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=49981

Masek, L. (2003). Recognition of human iris patterns for biometric identification (Tesis de maestría). The University of Western Australia. https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=f8c785dc2f3cf2683de45fdab9e1e4c2

Mayo Clinic. (2023). Eye anatomy. Mayo Foundation for Medical Education and Research. https://www.mayoclinic.org

National Eye Institute. (2022). Parts of the eye. U.S. Department of Health & Human Services. https://www.nei.nih.gov

8. Anexos:

Anexo A

El presente anexo muestra el código fuente desarrollado en **Python utilizando el framework Flask**, el cual constituye el núcleo de la aplicación. Este módulo cumple la función de gestionar el registro, eliminación y consulta de usuarios, así como el almacenamiento de las fotografías asociadas a cada uno de ellos en la base de datos.

https://github.com/proyectoiris179/Proyecto_IRS/blob/d7631514e576d8834dcef7aabb46ffaf032db836/Codigo%20APP

Este Contiene el desarrollo de una **interfaz web en HTML, CSS y JavaScript** que cumple la función de gestionar el registro de usuarios, administrar las fotografías asociadas y facilitar la interacción con el proceso de entrenamiento del modelo de reconocimiento de iris.

https://github.com/proyectoiris179/Proyecto_IRS/blob/8360e0779e8a751007225b225a6d6d829f671acc/Front%20APP

Se incluye el código fuente correspondiente a la interfaz web para la captura de fotografías de los usuarios. Este módulo solicita acceso a la cámara del dispositivo, muestra la transmisión en tiempo real y permite realizar capturas que se almacenan en el sistema para su posterior procesamiento dentro del algoritmo de reconocimiento de iris. La interfaz, desarrollada en HTML, CSS y JavaScript, integra botones de control que facilitan tanto la toma de imágenes como el retorno al menú principal, garantizando una experiencia sencilla e intuitiva para el usuario.

https://github.com/proyectoiris179/Proyecto_IRS/blob/d7631514e576d8834dcef7aabb46ffaf032db836/Captura.py

Código correspondiente a la implementación del **aplicativo web en Flask**, que cumple la función de administrar los procesos de captura de imágenes, registro de usuarios, entrenamiento del modelo de reconocimiento de iris y control de accesos mediante la cámara integrada en la Raspberry Pi.

https://github.com/proyectoiris179/Proyecto_IRS/blob/8360e0779e8a751007225b225a6d6d829f671acc/captura%20tiempo%20real

Este anexo contiene el código fuente desarrollado en Python, cuyo propósito es entrenar y evaluar un modelo de clasificación de iris empleando técnicas de visión por computadora y aprendizaje automático.

https://github.com/proyectoiris179/Proyecto_IRS/blob/d7631514e576d8834dcef7aabb46ffaf032db836/Entrenador

*** Organización para la Cooperación y Desarrollo Económico (OCDE)**
**** PA: Proyecto de Aula; PI: Proyecto Integrador; TI: Trabajo de Investigación**