

**Software para el control de acceso a través de reconocimiento facial
mediante Machine Learning que apoye la seguridad en la universidad
popular del cesar seccional Aguachica**

Ciro José Perrony Parra

Universidad Popular Del Cesar

Facultad De Ingenierías Y Tecnológicas

Programa Ingeniería De Sistemas

Línea De Investigación Ingeniería Del Software

Seccional Aguachica, Cesar

2024

**Software Para El Control De Acceso A Través De Reconocimiento Facial
Mediante Machine Learning Que Apoye La Seguridad En La Universidad
Popular Del Cesar Seccional Aguachica**

Ciro José Perrony Parra

**Proyecto De Grado Para Optar El Título De Ingeniero De Sistemas, Modalidad
Presencial.**

DIRECTOR

Luis Manuel Palmera

Codirector

Yohn Timy López

Universidad Popular Del Cesar

Facultad De Ingenierías Y Tecnológicas

Programa Ingeniería De Sistemas

Línea De Investigación Ingeniería Del Software

Seccional Aguachica, Cesar

2024

Nota de aceptación

Nota de aceptación:

Luis Manuel Palmera Quintero
Director

Yohn Timy López
Co-Director

Luis Octavio Ramírez
Evaluador 1

Darwin Navarro Pino
Evaluador 2

Aguachica, Día _____ Mes _____ Año _____

Dedicatoria

A Dios, por ser mi guía y fortaleza en cada paso de este camino.

A mis padres, por su amor incondicional y su constante apoyo, que me han inspirado a alcanzar mis sueños.

A mi esposa Aisquel, por ser mi compañera, mi motivación y mi mayor fuente de amor. Gracias por estar a mi lado en cada desafío.

Y a mi hijo Aarón, por recordarme cada día la importancia de perseverar y luchar por un futuro mejor.

Agradecimientos

Quiero expresar mi más profundo agradecimiento a Dios, quien ha sido mi principal apoyo y motivador, impulsándome a seguir adelante sin rendirme.

A mis padres, quienes han sido pilares fundamentales a lo largo de este proceso. Este logro es también un tributo a mi esposa Aisquel; gracias a su amor y aliento, he podido concluir con éxito mi proyecto de grado, el cual en un principio parecía un desafío insuperable.

Estoy muy agradecido con mi universidad por brindarme la oportunidad de formarme, y quiero extender un especial agradecimiento a mi director de proyecto, Luis Palmera, por su valiosa guía durante este proceso.

Desarrollar esta tesis no ha sido una tarea sencilla, pero puedo afirmar que he disfrutado cada momento de investigación, cada etapa y cada proyecto que se ha llevado a cabo. Este es un momento muy especial que espero perdure en el tiempo, no solo en la mente de quienes agradezco, sino también en el corazón de aquellos que invirtieron su tiempo en revisar mi trabajo. A todos ellos, les agradezco con todo mi ser.

TABLA DE CONTENIDO

1.	PROBLEMA.....	15
1.1	FORMULACIÓN DEL PROBLEMA	17
1.2	JUSTIFICACIÓN	18
1.3	OBJETIVOS	21
1.3.1	Objetivo General	21
1.3.2	Objetivos específicos	21
1.4	DELIMITACIÓN.....	22
1.4.2	Temporal.....	22
1.4.3	Espacial.	22
1.4.4	Contextual	22
2.	MARCO REFERENCIAL	27
2.1	MARCO HISTÓRICO	27
2.1.1	Internacional:	27
2.1.2	Nacional:.....	28
2.1.3	Local:	28
2.2	MARCO TEÓRICO	29
2.3	MARCO LEGAL	31
2.3.3	Ley 1581 de 2012.....	33
2.4	MARCO CONCEPTUAL	34
3.	DISEÑO METODOLÓGICO	38
3.1	TIPO Y ENFOQUE DE INVESTIGACIÓN	38
3.2	POBLACIÓN	38

3.3	INSTRUMENTOS DE RECOLECCIÓN DE DATOS	38
3.4	METODOLOGÍA DE DESARROLLO DE SOFTWARE	39
4.	ESQUEMA TEMÁTICO	41
4.2	PROPUESTA DE DISEÑO DEL SOFTWARE DE RECONOCIMIENTO FACIAL UTILIZANDO ALGORITMOS, LIBRERÍAS BASADAS EN IA Y TÉCNICAS DE MACHINE LEARNING	52
4.2.1	Diseño de navegación administrador	52
4.2.2	Diseño de navegación estudiante	53
4.2.3	Diseño de navegación personal de seguridad	53
4.2.4	Diseño del modelo entidad-relación	55
	Tecnologías para un software creado en Laravel	57
4.2.5	Modelado del sistema.....	58
4.2.6	Casos de uso del sistema	59
4.2.7	Diagramas UML	60
4.2.8	Descripción de casos de uso.....	63
4.2.9	Código de la Inteligencia Artificial (IA).....	72
4.3	CONSTRUIR E INTEGRAR A UNA INTERFAZ DE USUARIO AL SOFTWARE DE RECONOCIMIENTO FACIAL SEGÚN EL DISEÑO ESTABLECIDO.	76
4.4	REALIZAR MEDIANTE UNA PRUEBA PILOTO EN LA UNIVERSIDAD POPULAR DEL CESAR SECCIONAL AGUACHICA LA FUNCIONALIDAD Y ACEPTACIÓN DEL SOFTWARE DE RECONOCIMIENTO FACIAL.	118
5.	CONCLUSIONES	128
6.	RECOMENDACIONES.....	129
7.	BIBLIOGRAFÍA	130

**MANUAL DE USUARIO: SOFTWARE PARA EL CONTROL DE ACCESO A
TRAVÉS DE RECONOCIMIENTO FACIAL MEDIANTE MACHINE LEARNING**

138

Lista de figuras

Figura 1. Inteligencia Artificial	30
Figura 2. Navegación ruta del administrador	52
Figura 3. Navegación ruta estudiante	53
Figura 4. Navegación ruta personal de seguridad	54
Figura 5. Registro de Usuarios	54
Figura 6. Autenticación de Usuarios	55
Figura 7. Modelo Entidad-Relación.....	56
Figura 8. Login Administrador	60
Figura 9. Descripción del caso de uso administrador	60
Figura 10. Caso de uso de estudiante	61
Figura 11. Panel de navegación estudiante	62
Figura 12. Caso de uso Personal de seguridad	62
Figura 13. Panel de personal de seguridad	63
Figura 14. Librerías Python.....	73
Figura 15. Cargar imágenes	73
Figura 16. Entrenamiento de la IA	74
Figura 17. Código reconocimiento facial.....	74
Figura 18. Burndown inicial.....	89
Figura 19. Burndown semana 1	90
Figura 20. Diseño de vista del Login.....	91
Figura 21. Burndown semana 2	92
Figura 22. Panel de control Administrador	93

Figura 23. Burndown semana 3	95
Figura 24. Vista panel de inicio	96
Figura 25. Burndown semana 3	97
Figura 26. Diseño vista Crear perfiles	97
Figura 27. Diseño vista perfiles	98
Figura 28. Burndown semana 4	99
Figura 29. Burndown final semana 4	100
Figura 30. Diseño vista crear de usuarios	101
Figura 31. Diseño vista gestión de usuarios	101
Figura 32. Diseño de vista crear Roles	102
Figura 33. Diseño vista Gestión de Roles	102
Figura 34. Burndown semana 5	104
Figura 35. Diseño de vista subir imágenes	105
Figura 36. Burndown semana 6	106
Figura 37. Diseño vista Entrenar IA	106
Figura 38. Burndown semana 7	108
Figura 39. Burndown semana 8	110
Figura 40. Diseño vista optimización del procesamiento de imágenes	110
Figura 41. Burndown semana 9	112
Figura 42. Recolección y preparación del conjunto de datos	112
Figura 43. Burndown semana 10	114
Figura 44. Burndown semana 11	115
Figura 45. Burndown final	117

Figura 46. Reporte de pruebas 1	119
Figura 47. Reporte de pruebas 2	120
Figura 48. Reporte de pruebas 4	120
Figura 49. Reporte de pruebas 3	121
Figura 50. Socialización software parte 1.....	122
Figura 51. Socialización software parte 2.....	123
Figura 52. Socialización software parte 3.....	124
Figura 53. Socialización software parte 4.....	125
Figura 54. Socialización software parte 5.....	126
Figura 55. Socialización software parte 6.....	127

Lista de tablas

Tabla 1. Requisitos funcionales Administrador	48
Tabla 2. Requisitos Funcionales Estudiantes	49
Tabla 3. Requisitos Funcionales Personal de Seguridad.....	49
Tabla 4. Especificaciones tecnológicas	57
Tabla 5. Descripción del caso de uso Login	61
Tabla 6. Descripción de caso de uso administrador	63
Tabla 7. Descripción caso de uso Editar usuario	64
Tabla 8. Descripción caso de uso eliminar usuario	64
Tabla 9. Descripción caso de uso listar usuarios	65
Tabla 10. Descripción caso de uso Asignar Roles.....	65
Tabla 11. Descripción caso de uso crear rol	66
Tabla 12. Descripción de caso de uso editar rol	66
Tabla 13. Descripción de caso de uso ver rol	67
Tabla 14. Descripción caso de uso Asignar permisos a rol	67
Tabla 15. Descripción caso de uso Crear perfil	67
Tabla 16. Descripción caso de uso editar perfil	68
Tabla 17. Descripción caso de uso eliminar perfil.....	68
Tabla 18. Descripción caso de uso ver perfil	69
Tabla 19. Descripción de caso de uso Cargar imágenes a perfil.....	69
Tabla 20. Descripción caso de uso Asociar imágenes a perfil.....	70
Tabla 21. Descripción caso de uso Eliminar imágenes a perfil	70
Tabla 22. Descripción caso de uso Iniciar entrenamiento.....	71

Tabla 23. Descripción de caso de uso Iniciar entrenamiento.....	71
Tabla 24. Librerías usadas para el diseño del software.....	75
Tabla 25. Historias de usuario	77
Tabla 26. Tarea de Historias de Usuario Login.....	78
Tabla 27. Tarea de Historias de Usuario funcionalidades.....	79
Tabla 28. Tarea de Historias de Usuario editar datos.....	79
Tabla 29. Tarea de Historias de Usuario panel administrador	80
Tabla 30. Tarea de Historias de Usuario editar datos.....	80
Tabla 31. Tarea de Historias de Entrenamiento de la IA	81
Tabla 32. Tarea de Historias de generador de informes.....	81
Tabla 33. Tarea de Historias de Reunión Stakeholders.....	81
Tabla 34. Tarea de Historias de Monitoreo y estadísticas	82
Tabla 35. Sprint 1 Login.....	83
Tabla 36. Sprint 2 – gestión de imágenes faciales.....	84
Tabla 37. Sprint 3 – Entrenamiento del algoritmo de reconocimiento.....	84
Tabla 38. Sprint 4 - Gestionar alertas	85
Tabla 39. Revisión de cumplimiento de sprint 1.....	85
Tabla 40. Revisión de cumplimiento de sprint 2.....	86
Tabla 41. Revisión de cumplimiento sprint 3	86
Tabla 42. Tareas iniciales	87
Tabla 43. tareas semana 1	90
Tabla 44. tareas semana 2	92
Tabla 45. tareas semana 3	94

Tabla 46. tareas semana 3	96
Tabla 46. tareas semana 4	99
Tabla 48. tareas finalizadas semana 4	100
Tabla 49. tareas semana 5	103
Tabla 50. tareas semana 6	105
Tabla 51. tareas semana 7	107
Tabla 52. tareas semana 8	109
Tabla 53. Tareas semana 9	111
Tabla 54. Tareas semana 10	113
Tabla 55. Tareas semana 11	115
Tabla 56. Taskboard final	116

1. Problema

Las nuevas tecnologías de la información y la comunicación han irrumpido con fuerza y se ha extendido cada vez más en diferentes ámbitos de la sociedad (Domingo, 2021). Entre estas nuevas tecnologías está la inteligencia artificial (IA), que “es la habilidad de los ordenadores para hacer actividades que normalmente requieren inteligencia humana” (Rouhiainen, 2018).

El uso de la IA nos provee de diferentes herramientas para implementarlas según el entorno que corresponda. En la seguridad ciudadana, toma un enfoque preventivo, brindando técnicas como reconocimiento biométrico (huella dactilar, retina), reconocimiento de imágenes (procesado de rostros, extracción de gestos y rasgos distintivos, etc.), y muchas otras más.

Debido a los altos índices delictivos y de criminalidad que presenta el país, Colombia ocupa el segundo lugar en el ranking mundial de crimen organizado (Becerra, 2022), lo que induce a pensar que la posibilidad de que algún ciudadano sea víctima del actuar criminal sea muy alta. Cifras como la tasa de criminalidad de Colombia, donde por cada 100.000 habitantes se presentan 2.547 hechos criminales (CEJ, 2022), y donde una población carcelaria presenta una sobrepoblación del 19.12% de su capacidad máxima (CEJ, 2022); denotan la grave realidad de inseguridad que se vive en el país, y a esto puede sumarse que solo en el 2021, 5.004 criminales obtuvieron su libertad de centros penitenciarios y carcelarios mediante la modalidad de libertad por vencimiento de términos (CEJ, 2022), mayormente debido a la falta de pruebas que reafirmen su cometido delictivo.

La Universidad Popular del Cesar – Seccional Aguachica ha enfrentado una creciente problemática de inseguridad y robos, que afecta no solo a los estudiantes, sino también a los administrativos y docentes. Este fenómeno, que se manifiesta de diversas maneras, desde el hurto de pertenencias personales hasta actos más violentos, pone en riesgo la integridad física y emocional de la comunidad universitaria. Basado en lo expuesto por Cerezo y Enríquez, (2024), otro aspecto de

la inseguridad es el vandalismo, que incluye daños a la infraestructura como lo son grafitis, destrucción de mobiliario y equipo, lo cual genera costos adicionales para las instituciones y afecta el ambiente de aprendizaje.

La desigualdad económica y la falta de oportunidades pueden llevar a algunos individuos a recurrir al crimen como medio de subsistencia. Las universidades, con su concentración de jóvenes y la percepción de que estos portan objetos de valor, se convierten en objetivos atractivos para los delincuentes. Muñoz y Ponces (2024) expresan que muchas universidades carecen de una infraestructura adecuada para garantizar la seguridad en sus campus. La falta de iluminación en áreas exteriores, la escasez de cámaras de vigilancia y la insuficiencia de personal de seguridad contribuyen a la vulnerabilidad de la comunidad universitaria.

La falta de coordinación efectiva entre la universidad y las autoridades locales de seguridad pública puede dificultar la implementación de medidas preventivas y la respuesta rápida ante incidentes delictivos. En algunos casos, la percepción de que los delitos no serán castigados adecuadamente puede incentivar a los delincuentes. La impunidad y la falta de consecuencias legales para los perpetradores aumentan la incidencia de robos y actos violentos, (La Madrid y Ruiz, 2023).

Las víctimas de robos y violencia pueden experimentar estrés postraumático, ansiedad y miedo constante, afectando su bienestar mental y emocional. Este impacto puede ser especialmente severo para los estudiantes jóvenes que están lejos de sus familias. Un entorno inseguro puede afectar negativamente el ambiente académico. Los estudiantes pueden sentirse menos motivados y concentrados en sus estudios, y los docentes y administrativos pueden trabajar bajo constante temor, lo que reduce la calidad de la educación y los servicios ofrecidos, (Jiménez et al., 2023).

Con lo anteriormente expuesto, es claro que la universidad carece de mecanismos y herramientas efectivas para llevar el control y el registro de las personas que entran y salen de la institución, indiferentemente si están vinculadas (estudiantes, docentes, administrativos) o no (visitantes y el público en general). En la actualidad,

el ingreso del personal a las instalaciones no tiene ninguna seguridad, (Ver Anexo I) solo se depende de la memoria del vigilante que esté en la entrada de la institución para llevar dicho registro, además de las planillas de asistencia que se toman en las clases magistrales (esto aparte de obsoleto solo cubre a los estudiantes y docentes). De esta manera, lo mencionado anteriormente, lleva al hecho de que la universidad no está exenta del actuar criminal, y que por el contrario el hecho de no contar con un mecanismo que permita llevar un registro de las personas que ingresan a la institución, causa que la posibilidad de que pueda ocurrir algún acto delincuenciales y no se dé con los responsables, sea muy alta.

1.1 Formulación Del Problema

¿De qué manera se puede mejorar la seguridad en las instalaciones de la Universidad Popular del Cesar Seccional Aguachica mediante la implementación de herramientas de reconocimiento facial utilizando técnica de Machine Learning?

1.2 Justificación

La necesidad de aumentar la seguridad siempre ha sido motivo de preocupación general, y la tecnología ha jugado un papel fundamental en encontrar nuevas formas de asegurar la vida y los bienes de las personas. El permitir el ingreso de cualquier persona dentro de la institución educativa, es un factor de inseguridad, más cuando no se lleva registro del ingreso y/o salida de las personas en la institución, de acuerdo con, (Portilla, 2018, p. 9). Por otra parte, González y González, (2024), expresan que el reconocimiento facial, potenciado por algoritmos de Machine Learning, ofrece una solución robusta y eficiente para el control de acceso. Esta tecnología permite identificar a las personas de manera rápida y precisa, basándose en características biométricas únicas y prácticamente imposibles de falsificar. En un mundo donde la seguridad es cada vez más crucial, la implementación de un software de control de acceso mediante reconocimiento facial representa un avance significativo.

Como se menciona en la descripción del problema, la Universidad no cuenta con herramientas efectivas para llevar un control y el registro de las personas que interactúan con las instalaciones, es por ello que se hace necesario crear un sistema de control de acceso que permita reconocer a las personas evitando así las actuales desventajas que presenta el proceso de ingreso/salida de las personas. El contar con un sistema de control de acceso va a permitir a la universidad controlar el acceso a la instalación, solicitando datos de todas las personas que ingresan, se propone entonces realizar este sistema usando detección facial (reconocimiento de imágenes) porque de esta manera, los datos solo serán recolectados una sola vez y después el reconocimiento será realizado automáticamente por el sistema.

Uno de los principales beneficios del reconocimiento facial es su alta precisión. Los algoritmos de reconocimiento facial, son capaces de analizar y comparar características faciales con una exactitud impresionante, (Pilay y Alban, 2024). De la misma manera, Juanazo y Lamboggia, (2024), explica que a “través del

entrenamiento con grandes volúmenes de datos, estos algoritmos pueden mejorar continuamente su capacidad para reconocer rostros, minimizando la posibilidad de errores”, (p. 26). Esto resulta en un sistema de control de acceso altamente fiable, capaz de identificar correctamente a las personas autorizadas y detectar intrusos potenciales. De esta manera, la institución se beneficiará de un sistema de apoyo a los vigilantes quienes tendrán más tiempo de cumplir con otras actividades, y podrán llevar un mejor control del acceso sobre quien ingresa/sale de las instalaciones, como lo expresa, (Guallasamin y Guallichico, 2024).

De acuerdo a lo anterior, se propone un software de reconocimiento facial que pueda generar informes, de las personas que ingresan y salen de la universidad, de tal manera que si ocurre algún hecho desfavorable se pueda verificar dichos informes en pro de determinar posibles responsables. Además, se ejercerá cierto rigor en la entrada de la universidad, porque el vigilante encargado tendrá la tarea de registrar a las personas (una vez) y adicionalmente conocer el propósito de su visita a la institución, y no como ahora que solo cumplen con la tarea de abrir y cerrar la puerta a cualquiera que entre o salga de la institución.

Hoy en día la Inteligencia Artificial (IA), abre un sin número de nuevas oportunidades que permiten optimizar procesos dentro de las empresas, a través del “Reconocimiento de imágenes, patrones, facial, biométrico, etc.”, el cual consisten en el procesamiento de imágenes y datos con el objetivo de comprobar la veracidad en las identidades o verificación de objetos para mejorar los servicios que se prestan dentro de una organización, (García y Barona, 2024). Por tal motivo, la importancia de este proyecto de investigación el cual puede servir de apoyo para mejorar la seguridad al momento de acceder a la Universidad Popular del Cesar, teniendo un control de acceso e identificación de personas a través del reconocimiento facial.

La implementación de un software de control de acceso mediante reconocimiento facial agiliza significativamente el proceso de identificación. A diferencia de los métodos tradicionales que requieren la presentación de tarjetas o la introducción de códigos, el reconocimiento facial puede realizarse en cuestión de segundos. Esta

rapidez es especialmente beneficiosa en entornos donde se requiere el acceso de un gran número de personas, como en universidades, corporaciones y eventos masivos. La eficiencia del sistema reduce las filas y tiempos de espera, mejorando la experiencia de los usuarios. Las características faciales son únicas para cada individuo, lo que hace extremadamente difícil la falsificación o suplantación. Esta capacidad de monitoreo continuo y en tiempo real fortalece las medidas de seguridad y reduce el riesgo de acceso no autorizado.

La adaptación a nuevas tecnologías puede encontrar resistencia por parte de los usuarios. Es esencial proporcionar capacitación y concienciación sobre los beneficios del sistema para facilitar su aceptación. La demostración de la efectividad y seguridad del reconocimiento facial puede ayudar a disipar dudas y promover una adopción más amplia. Los avances tecnológicos y la disminución de los costos de hardware y software están haciendo que esta tecnología sea más accesible. Por esta razón, la inversión en seguridad avanzada puede justificar los costos a largo plazo, considerando la reducción de riesgos y la mejora en la eficiencia operativa dentro de la Universidad Popular del Cesar – Seccional Aguachica.

1.3 Objetivos

1.3.1 Objetivo General

Desarrollar un software de control de acceso a través de reconocimiento facial mediante machine learning para el acceso de estudiantes a la universidad popular del Cesar seccional Aguachica.

1.3.2 Objetivos específicos.

- Definir los requerimientos funcionales y no funcionales.
- Diseñar software de reconocimiento facial utilizando algoritmos, librerías basadas en IA y técnicas de Machine Learning.
- Construir e integrar a una interfaz de usuario al software de reconocimiento facial según el diseño establecido.
- Realizar una prueba piloto en la universidad popular del Cesar seccional Aguachica la funcionalidad y aceptación del software de reconocimiento facial.

1.4 Delimitación

1.4.2 Temporal.

Se determinará un lapso de 6 meses a partir de la aprobación del formato 2 para la terminación de esta investigación.

1.4.3 Espacial.

El desarrollo de este estudio se efectuará en la Universidad Popular del Cesar, Seccional Aguachica; en el municipio de Aguachica en el departamento del Cesar.

1.4.4 Contextual

La población que será objeto de estudio abarcará a los estudiantes, docentes y personal administrativo de la Universidad Popular del Cesar Seccional Aguachica.

Historia de la Universidad Popular

Según la información proporcionada por la Universidad Popular del Cesar, UPC Seccional - Aguachica, (2020), la universidad fue creada en el año 1976 a partir de la transformación del Instituto Tecnológico Universitario del Cesar, se transforma en Universidad como establecimiento público con personería jurídica, teniendo como sede principal la ciudad de Valledupar, cuyo objetivo principal será la investigación y la docencia a través de programas que han de conducir a la obtención de licenciaturas, grados profesionales y títulos académicos.

En 1992, se realizó un proyecto importante para consultar a los residentes de los Municipios del Sur del Departamento del Cesar sobre la posibilidad de establecer una Universidad en la región. Como resultado de este estudio de factibilidad realizado por los profesores Simón Martínez Guarnes y Carmen Patricia Guerrero, se determinó que el Municipio de Aguachica tenía las condiciones más favorables para el establecimiento de la Universidad.

A partir de la iniciativa del Dr. Vicente Vaños Galvis, representante de exrectores de la Universidad Popular del Cesar en el Consejo Superior Universitario, se propuso en septiembre de 1995 establecer programas de Administración de Empresas y Contaduría Pública en el Municipio de Aguachica como extensiones de la

universidad. Esta propuesta se basó en las necesidades sociales, políticas, académicas y económicas de la región, y en los beneficios que estos programas podrían ofrecer.

Misión

La Universidad Popular del Cesar, como institución de educación superior oficial del orden nacional, forma personas responsables social y culturalmente; con una educación de calidad, integral e inclusiva, rigor científico y tecnológico; mediante las diferentes modalidades y metodologías de educación, a través de programas pertinentes al contexto, dentro de la diversidad de campos disciplinares, en un marco de libertad de pensamiento; que consolide la construcción de saberes, para contribuir a la solución de problemas y conflictos, en un ambiente sostenible, con visibilidad nacional e internacional (Molina, 2020).

Visión

En el año 2025, la Universidad Popular del Cesar será una Institución de Educación Superior de alta calidad, incluyente y transformadora; comprometida en el desarrollo sustentable de la Región, con visibilidad nacional y alcance internacional (Molina, 2020).

1.4.4.1 Aspecto Geográfico.

Aguachica es un municipio ubicado al sur Departamento del Cesar, Colombia, y la segunda ciudad en población e importancia después de la capital, Valledupar. Se encuentra en una posición geográfica estratégica, siendo paso obligado entre el norte y el interior del país; interconectada por carretera con la Troncal del Magdalena (Ruta Nacional 45) y la Troncal Central (Ruta Nacional 45^a), además del transporte férreo, aéreo y en forma paralela la vía fluvial del río Magdalena. (EcuRed, Municipio Aguachica, 2018)

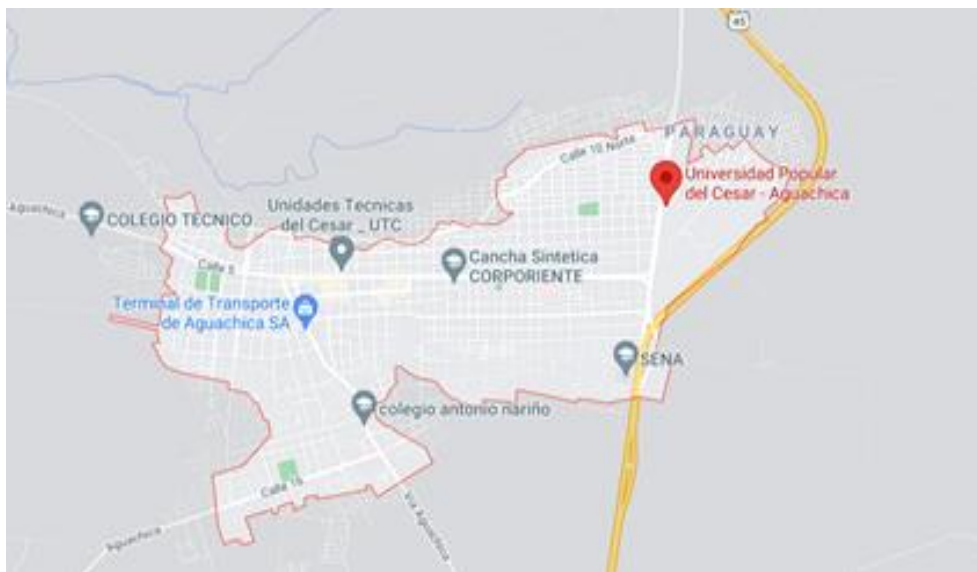
Límites del municipio:

Por el Norte con el municipio de La Gloria (Cesar), El Carmen (Norte de Santander). Por el este con Río de Oro (Cesar). Por el Sur con Río de Oro, San Martín (Cesar) y Puerto Wilches (Santander). Por el Oeste con Gamarra (Cesar) y Morales (Bolívar). Extensión total: 876.26 Km². Altitud de la cabecera municipal (metros sobre el nivel del mar): 50 y 200.

Temperatura media: 28°C° C

Distancia de referencia: Dista de Valledupar, la capital a 301 Km

El acceso al municipio se puede lograr desde la Costa Caribe colombiana a través de la vía que conecta Santa Marta con Aguachica, uniendo los departamentos de Magdalena y Cesar. Por otro lado, desde el sur del país se puede acceder mediante la Troncal del Magdalena Medio, o a través de la vía que conecta Bogotá, Bucaramanga y Aguachica, enlazando los departamentos de Cundinamarca, Boyacá, Santander y Cesar. Además, se cuenta con la carretera Aguachica-Ocaña, que comunica el municipio con el nororiente, y la carretera Aguachica-Gamarra, que conecta el municipio con el río Magdalena y la región occidental. (Alcaldia de Aguachica, 2022).



Fuente: tomado de (Google Maps, 2022)

1.4.4.2 Aspecto Histórico.

El actual territorio de Aguachica se empezó a consolidar en los primeros veinte años del siglo XVIII a partir de la hacienda de San Roque de propiedad de Don Antón García de Bonilla, localizada al oriente de la actual vía cuarenta, hacia la planta del acueducto municipal. Por razones asociadas a una los primeros pobladores de este asentamiento debieron trasladarse más abajo, alrededor del actual Parque San Roque. El primer núcleo poblacional era un incipiente núcleo de habitación en propiedades que hacia 1722 pertenecía a Don Casimiro Ramos de Barahoja, articulado al flujo de mercancías y población de Gamarra a Ocaña. Unas décadas después, el 16 de agosto de 1748, mediante concesión Realenga de los terrenos de Aguachica viejo y San Francisco hecho a favor de Don José Lázaro de Rivera se realizó un acto de fundación o reconocimiento de la parroquia, sus fundaciones o refundaciones fueron reconfirmadas por la administración del Virrey José Alfonso Pizarro, entre 1749 y 1753. Tanto en la fundación de Aguachica como la de san Francisco, hacia 1753 se inició la construcción de algunas casas. Se acepta como fecha de fundación, el 16 de agosto de 1748, habiendo sido elevado a la categoría municipal en virtud de la ordenanza número 40 de 1914 (Alcaldía de Aguachica, 2022).

1.4.4.3 Aspecto Cultural.

En el aspecto de Fiestas, Aguachica es un pueblo festivo ciento por ciento, distinguiéndose las siguientes fiestas: Carnavales, Semana Santa, Fiesta de la Cruz, Corpus Christi, Novena de Nuestra Señora del Carmen (Celebración que realizan los peseros y conductores), Novena al Patrono San Roque, Cumpleaños de Aguachica 16 de agosto; Día del gestor Cultural (Festival Folclórico), Novena y Misa de Aguinaldos. Dentro de su cultura también se encuentran algunas leyendas: Las Animas Benditas, El Nazareno, El Perro negro del Cristo y del Agüil (EcuRed, Ecured, 2022).

1.4.4.4 Aspecto Económico

La economía de Aguachica se basa principalmente en el sector agropecuario, la agroindustria y el comercio. Esta diversidad ha impulsado el desarrollo de una

variedad de servicios de apoyo, como asesoría agrotécnica, servicios financieros, transporte y otros servicios empresariales y personales dirigidos a los diferentes sectores económicos y a la población local y regional. Estos servicios complementarios juegan un papel fundamental en el impulso y crecimiento de los distintos sectores económicos de la zona.(Alcaldía de Aguachica, 2022).

1.4.4.5 Historia de la Universidad

La Universidad Popular del Cesar fue fundada en 1976, cuando el Instituto Tecnológico Universitario del Cesar se convirtió en una universidad pública con personalidad jurídica. La sede principal se encuentra en la ciudad de Valledupar. Su principal objetivo es la investigación y la enseñanza, a través de programas académicos que conducen a la obtención de licenciaturas, grados profesionales y títulos académicos. La universidad se enfoca en brindar una educación de calidad y fomentar el desarrollo intelectual y profesional de sus estudiantes.

2. Marco Referencial

En este capítulo se expone la base conceptual y teórica que sustenta el desarrollo de la presente investigación. En la sección 2.1 se abordan los trabajos históricos vinculados al proyecto. La sección 2.2 examina la fundamentación teórica de un sistema de reconocimiento facial. La sección 2.3 analiza el marco legal del proyecto. Por último, la sección 2.4 expone los conceptos clave relevantes para el proyecto.

2.1 Marco Histórico

Aquí se describen los referentes históricos a nivel internacional, nacional y local que han impactado en la sociedad mediante sistemas de control de acceso mediante reconocimiento facial. A nivel internacional, expertos han mejorado la precisión y confiabilidad de esta tecnología. A nivel nacional, países como China y Estados Unidos han implementado sistemas en aeropuertos y entidades gubernamentales. A nivel local, empresas, centros comerciales, universidades y hospitales han adoptado esta tecnología. Sin embargo, el uso del reconocimiento facial plantea preocupaciones de privacidad y se han establecido regulaciones para proteger los datos personales y los derechos individuales.

2.1.1 Internacional:

En Ecuador, Valencia y Brito, presentan un trabajo de investigación titulado: “Implementación de un Control de Acceso con Reconocimiento Facial Mediante OpenCV para la Sala de Profesores de la FIE” en la Escuela Superior Politécnica de Chimborazo. En donde desarrollan un sistema de control de acceso basado en técnicas de reconocimiento facial la librería OpenCV del lenguaje de programación Python. El sistema está diseñado, de forma que, nos permite aplicar nuevas tecnologías mejorando la seguridad de ingreso del personal. Aplicando el método deductivo se diseñó un sistema de reconocimiento facial basado en un método estadístico que calcula los valores y vectores propios de la matriz de covarianza de intensidad de color en el espacio de grises. El sistema concede el acceso a la

persona, si esta se encuentra dentro del horario preestablecido por el administrador, además se proporciona una clave personal en caso de falla en la detección facial, si el usuario se encuentra en el horario establecido se acciona la salida por relé del sistema. De acuerdo con (Valencia Rodríguez & Brito Poveda, 2015).

2.1.2 Nacional:

En este ámbito se hará una revisión de los referentes a nivel nacional que se han desarrollado. Alonso, y Castaño, presentan un trabajo de investigación titulado: “Sistema de reconocimiento facial para control de acceso a viviendas” en la Universidad Católica de Colombia. Donde se presenta el desarrollo de un sistema de reconocimiento facial el cual permitirá el control de acceso a una casa. Debido a la alta inseguridad que se evidencia en la ciudad, por este hecho, se vio la necesidad de crear una solución que disminuya estos porcentajes de hurto. Con este sistema no solo se pretende brindar seguridad y confort al usuario si no incentivar el uso herramientas tecnológicas como el reconocimiento facial, y dispositivos que están en el mercado, para implementar un control de acceso para viviendas seguro y de un bajo costo. De acuerdo con (Alonso Sierra & Castaño Saavedra, 2019).

Gómez, denota en el desarrollo de su trabajo de investigación titulado: “Aplicación móvil de control de acceso por medio de reconocimiento facial”, en la Universidad de Antioquia, en donde se propone la creación de una solución biométrica y de lectura de cédula colombiana que se desarrolla con la empresa Idelity, la cual es una empresa enfocada en la investigación y la prestación de servicios de reconocimiento biométrico. De acuerdo con (Gómez Alzate, 2022).

2.1.3 Local:

A nivel local actualmente en la Universidad Popular del Cesar – Seccional se está llevando a cabo un proyecto de investigación titulado: “Sistema de Reconocimiento de Placas Vehiculares Mediante Técnicas De Machine Learning Para El Control De Acceso De Vehículos A La Universidad Popular Del Cesar Seccional Aguachica”, en el cual se muestra el desarrollo de un sistema de reconocimiento de placas

vehiculares mediante machine learning, el cual permitirá llevar un control de acceso de los vehículos que entran y salen de la institución.

2.2 Marco Teórico

Esta sección se divide en las diferentes teorías basadas en inteligencia artificial, que componen la fundamentación teórica para el desarrollo del presente trabajo de investigación.

2.2.2 Inteligencia Artificial

La inteligencia artificial (IA), es una disciplina que se encarga de imitar las características y el comportamiento inteligente de los seres humanos en sistemas computacionales (Gestión, 2018). En el mundo de la IA, existen dos grandes tipos en los cuales esta se divide: la IA convencional y, la IA computacional. Para el presente trabajo de investigación, se va a enfatizar solo en la IA convencional.

2.2.2.1 IA (Inteligencia Artificial) Convencional:

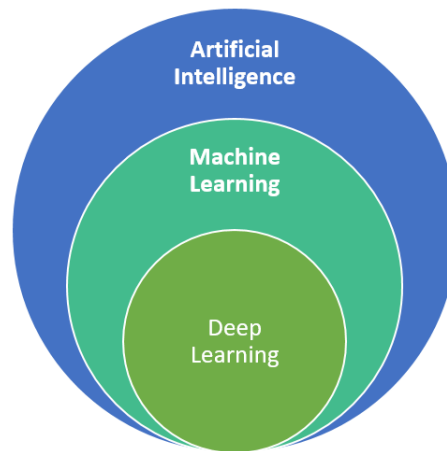
La IA convencional o también conocida como IA simbólico-deductiva, es una forma de IA que se basa en el análisis tanto formal, como estadístico sobre el comportamiento humano en diferentes situaciones o tipos de problemas. Sirve de apoyo y ayuda a la toma de decisiones en situaciones complejas, además de proponer la solución a determinados problemas, ya que cuenta con autonomía propia (no necesariamente necesita de la intervención humana para funcionar) y puede autorregularse y controlarse para mejorar (Gestión, 2018).

Adicionalmente, la IA como factor primordial está impulsando el desarrollo de la de la industria y está siendo el factor clave para que se dé una nueva revolución industrial denominada Industria 4.0, en la cual predominan factores como la automatización y autosuficiencia de las máquinas de producción en donde el valor agregado lo genera en este caso, la misma IA.

La IA se ha ampliado constantemente a lo largo del tiempo, generando nuevas disciplinas más específicas como: machine learning y Deep learning (hay muchas

otras más, pero para el desarrollo del presente trabajo de investigación, solo nos interesa estas dos).

Figura 1. Inteligencia Artificial



Fuente: Tomado de (IESC, 2018)

Por otro lado, según Arthur (2019) en su trabajo de investigación, explica que el aprendizaje automático en el campo de estudio da a las computadoras la capacidad de aprender sin estar explícitamente programado. Aprendizaje automático (Machine Learning en inglés) se usa para enseñar a las máquinas cómo manejar los datos más eficientemente. A veces, después de ver los datos, no podemos interpretar la información extraída de los datos. En ese caso, aplicamos el aprendizaje automático (Technical Report, 2019). Existen 3 tipos distintos de aprendizaje automático, como lo son: aprendizaje supervisado, no supervisado, semi supervisado, por refuerzo, activo, evolutivo y de conceptos.

2.2.2.2 Tipos De Aprendizaje Automático

Aprendizaje Supervisado: El aprendizaje supervisado es cuando el algoritmo produce una entre las entradas y salidas deseadas del sistema. Constituye un algoritmo de aprendizaje basado en ejemplos donde el nuevo conocimiento es

inducido a partir de una serie de ejemplos y contraejemplos (Chaviano Arteaga, 2015).

Aprendizaje No Supervisado: El aprendizaje no supervisado se da cuando todo el modelado de los datos pasa en las entradas del sistema, y no se tiene idea de las posibles salidas o las etiquetas/categorías de los datos de salida. Constituye un tipo de aprendizaje por observación y descubrimiento, donde el sistema de aprendizaje analiza una serie de entidades y determina que algunas tienen características comunes, por lo que pueden ser agrupadas formando un concepto (Chaviano Arteaga, 2015).

Aprendizaje por Refuerzo: Empleado para diseñar y desarrollar comportamientos colectivos por medio de la generación automática de comportamientos individuales que los diferentes agentes aprenden a través de la interacción con su entorno, seleccionando para ello aquellas acciones que permitan obtener la máxima recompensa global posible (Mousavi, 2018).

Deep Learning: El aprendizaje profundo o Deep Learning es una subrama del aprendizaje automático, y se basa principalmente en realizar análisis y procesar la información mediante diferentes capas que van captando cada vez características más complejas de los datos. Así que, entre más capas, mejor será el análisis, y de allí proviene el término Deep learning. (Gonzales Muñiz, 2018).

El aprendizaje profundo se divide en diferentes ramas como redes neuronales, redes neuronales convolucionales o redes neuronales recurrentes. Estas arquitecturas son utilizadas en campos como la visión por computador, reconocimiento del habla o reconocimiento de dígitos manuscritos, reconocimiento de rostros, entre otros. (Oliva Rodriguez, 2018).

2.3 Marco Legal

2.3.1 Derechos de autor. Los derechos de autor a través de la ley 23 De 1982 (Congreso de la república de Colombia, 1982), por medio de la cual se regula los

derechos morales y patrimoniales que la Ley concede a los autores (los derechos de autor), por el solo hecho de la creación de una obra literaria, artística o científica, esté publicada o inédita, según esta ley se establece los siguientes artículos:

Artículo 1. Los creadores de obras literarias, científicas y artísticas recibirán protección para sus obras de acuerdo con lo establecido en la presente ley y, en la medida en que sea compatible, según el derecho común. Además, esta ley también protege los derechos conexos de los intérpretes, ejecutantes, productores de programas y organismos de radiodifusión en relación con los derechos de autor.

Artículo 2. Los derechos de autor se aplican a las obras científicas, literarias y artísticas, que incluyen todas las creaciones del espíritu en los campos científico, literario y artístico, sin importar el modo o forma de expresión ni su destino. Estas obras abarcan libros, escritos, conferencias, obras dramáticas, coreográficas, musicales, cinematográficas, visuales (dibujo, pintura, arquitectura, etc.), fotográficas, ilustraciones, mapas, planos y cualquier otra producción relacionada con la ciencia, literatura o arte que pueda ser reproducida o definida mediante cualquier forma de impresión, reproducción, grabación, entre otros medios conocidos o por conocer. Esta descripción fue complementada por la Ley 44 de 1993.

Artículo 3. Los derechos de autor comprenden para sus titulares las facultades exclusivas:

- A. De disponer de su obra a título gratuito u oneroso bajo las condiciones lícitas que su libre criterio les dicte.
- B. De aprovecharla, con fines de lucro o sin él, por medio de la imprenta, grabado, copias, molde, fonograma, fotografía, película cinematografía, videograma, y por la ejecución, recitación, representación, traducción, adaptación, exhibición, transmisión, o cualquier otro medio de reproducción, multiplicación, o difusión conocida o por conocer.

C. De ejercer las prerrogativas, aseguradas por esta Ley en defensa de su "derecho moral", como se estipula en el Capítulo II, Sección Segunda, artículo 30 de esta Ley. Adiciona mediante la Ley 44 de 1993.

2.3.2 Ley 1273 de 2009. Mediante esta ley se introducen modificaciones al Código Penal, estableciendo un nuevo bien jurídico protegido denominado "protección de la información y los datos", y se asegura la integridad de los sistemas que utilizan tecnologías de la información y las comunicaciones, entre otras disposiciones. Además, se agregan dos capítulos al Código Penal colombiano: el primero trata sobre los delitos contra la confidencialidad, integridad y disponibilidad de los datos y sistemas informáticos, y el segundo aborda los delitos informáticos y otras infracciones relacionadas. (Congreso de Colombia, 2009).

2.3.3 Ley 1581 de 2012. Por la cual se dictan disposiciones generales para la protección de datos personales y el decreto 1377 de 2013 por el cual se reglamenta parcialmente la Ley 1581 de 2012. Cuyo objetivo es desarrollar el derecho constitucional que tienen todas las personas a conocer, actualizar y rectificar las informaciones que se hayan recogido sobre ellas en bases de datos o archivos, y los demás derechos, libertades y garantías constitucionales a que se refiere el artículo 15 de la Constitución Política; así como el derecho a la información consagrado en el artículo 20 de la misma. Los principios y disposiciones contenidas en la presente ley serán aplicables a los datos personales registrados en cualquier base de datos que los haga susceptibles de tratamiento por entidades de naturaleza pública o privada, que se encuentren dentro del territorio colombiano o cuando el encargado o responsable del tratamiento no establecido en el territorio nacional le sea aplicable la legislación colombiana en virtud de normas y tratados internacionales (Congreso de la Republica, 2012).

2.4 Marco Conceptual

Con el fin de conocer la terminología empleada en el desarrollo del presente trabajo de investigación, se define la siguiente lista con términos y expresiones relevantes:

IA (Inteligencia Artificial): La IA comprende a la capacidad de las máquinas para utilizar algoritmos, aprender de los datos, procesar los datos y luego usarlos para resolver problemas tal cual como lo haría un ser humano. A diferencia de las personas, la IA no necesita descansar y tampoco se ve afectada por factores que, si afectan a los seres humanos, como las emociones, las noticias, el nivel social y económico, entre muchas otras; por lo que el rendimiento va a ser mejor en comparación con los seres humanos (Rouhiainen, 2018).

Biometría: La biometría son indicadores biológicos o características físicas que se pueden utilizar para identificar a una persona. El análisis de huellas dactilares, el reconocimiento facial y el escaneo de retina también son opciones populares, así como todo tipo de biometría (Kaspersky, 2022).

Librería: son conjuntos de archivos de código que se utilizan para desarrollar software. Su objetivo es facilitar la programación, al proporcionar funcionalidades comunes, que ya han sido resueltas previamente por otros programadores. Estas librerías pueden estar orientadas específicamente a facilitar la solución a un tipo de problema (Gob, 2022).

Sistema de control de acceso: Un sistema de control de acceso se refiere a un mecanismo que brinda acceso a datos o recursos en función de una identidad actualmente autenticada. Básicamente, encontramos que los sistemas de control de acceso vienen en muchas formas y son adecuados para diferentes aplicaciones (TecnoSeguro, s.f.)

Redes neuronales artificiales: Una red neuronal es un método de la inteligencia artificial que enseña a las computadoras a procesar datos de una manera que está inspirada en la forma en que lo hace el cerebro humano. Se trata de un tipo de proceso de machine learning llamado aprendizaje profundo, que utiliza los nodos o

las neuronas interconectados en una estructura de capas que se parece al cerebro humano. Crea un sistema adaptable que las computadoras utilizan para aprender de sus errores y mejorar continuamente. De esta forma, las redes neuronales artificiales intentan resolver problemas complicados, como la realización de resúmenes de documentos o el reconocimiento de rostros, con mayor precisión. (Amazon, 2023)

Python: Python es un lenguaje de programación de alto nivel que se utiliza para desarrollar aplicaciones de todo tipo. A diferencia de otros lenguajes como Java o .NET, se trata de un lenguaje interpretado, es decir, que no es necesario compilarlo para ejecutar las aplicaciones escritas en Python, sino que se ejecutan directamente por el ordenador utilizando un programa denominado interpretador, por lo que no es necesario “traducirlo” a lenguaje máquina (Santander Universidades, 2021).

OpenCV: Las siglas OpenCV provienen de los términos anglosajones “Open Source Computer Vision Library”. OpenCV es una biblioteca especializada en visión artificial iniciada por Intel en 1999, que busca el procesamiento de imágenes en tiempo real y que además sus implementaciones se encuentran libres de patentes (Caballero Barriga, 2017).

RF RNF: Son las abreviaciones para Requisitos Funcionales y Requisitos no Funcionales. Estos son las características o los servicios que debe proveer un sistema para que pueda funcionar correctamente. Los requisitos funciones son los requisitos indispensables con los cuales sin ellos el sistema no sería funcional. Estos requisitos se pueden dividir en los requisitos del usuario, requisitos del sistema y requisitos del cliente. Los requisitos no funcionales son requisitos más que todos operativos, aunque no interfieren directamente con el funcionamiento del sistema, si son necesarios para que la aplicación perdure en el tiempo como, por ejemplo: que la aplicación cumpla con las leyes establecidas, que tenga colores acordes con los productos ofertados, que tenga multilenguaje para las personas de diferentes regiones etc. (Navas-Triana, 2021)

Base de datos: Una base de datos se refiere a un conjunto de datos interrelacionados, almacenados y organizados de manera sistemática, con el fin de facilitar su conservación, búsqueda y aprovechamiento. El propósito principal de las bases de datos es brindar una mayor facilidad de uso y acceso a la información. Por esta razón, se utilizan ampliamente en diversos sectores como empresas, entidades públicas, investigación científica y bibliotecas, entre otros. Su implementación permite gestionar y organizar los datos de manera eficiente, lo que facilita la búsqueda, recuperación y utilización de la información de manera efectiva. (Significados, 2023).

Modelo entidad-relación: Es un tipo de modelo de datos de alto nivel que normalmente se emplea para la construcción de las bases de datos relacionales. El modelo entidad-relación muestra la estructura de la base de datos empleando cualquier tipo de herramientas conceptuales. Este modelo es uno de los más utilizados en la actualidad para modelar problemas realizar y administrar datos de forma dinámica (Millán, 2020).

Visual Studio Code: Es un editor de código fuente independiente que se ejecuta en Windows, macOS y Linux. La elección principal para desarrolladores web y JavaScript, con extensiones para admitir casi cualquier lenguaje de programación (Code, 2022)

Pip: El nombre es un acrónimo recursivo que se puede interpretar como Pip Instalador de Paquetes o Pip Instalador Python. Este es un sistema de gestión de paquetes sencillo utilizado para la instalación y administración de paquetes en el lenguaje de programación Python (Pypi, 2022)

CLI: Console Line Interface (al español Interfaz de línea de consola), es una interfaz de consola que se caracteriza porque su funcionalidad se basa en líneas de comandos. Es un programa que permite a los usuarios escribir comandos en formato de texto para enviar instrucciones a una computadora para que esta realice tareas específicas (Hostinger, 2022).

Reconocimiento facial: es una tecnología biométrica que utiliza características faciales para identificar o verificar la identidad de una persona. Este proceso involucra la captura de una imagen o secuencia de imágenes del rostro de un individuo y el análisis de sus características distintivas, tales como la estructura de los huesos, la forma y posición de los ojos, nariz, boca, y otros contornos faciales, (García y Barona, 2024).

3. Diseño Metodológico

3.1 Tipo Y Enfoque De Investigación

Para el desarrollo de la investigación, se procedió a realizar una investigación de tipo aplicada con enfoque descriptivo. La investigación aplicada permite usar la ciencia a los problemas de la sociedad y las empresas. Para hacerlo, se nutre de lo que se aprende de la investigación básica, de la que toma los conocimientos necesarios. Su propósito es solucionar situaciones que se presentan en la realidad. Por eso, su punto de vista es claro, analizar y estudiar dichos problemas para encontrar soluciones de acuerdo con (Arias, 2021). Se recurrió o se tuvo en cuenta el tipo de investigación aplicada, porque, nos permitirá tratar situaciones o problemas reales que enfrenta el proceso de aprendizaje de la mano de las TIC.

3.2 Población

La población inicialmente será aquella que está conformada por docentes, estudiantes y administrativos de la Universidad Popular del Cesar Seccional Aguachica. La importancia al delimitar una población estadística implica establecer límites específicos para el estudio. Esto incluye definir el grupo de individuos o elementos de la muestra, el problema a abordar y el periodo de tiempo considerado. Es esencial para garantizar la viabilidad y enfoque adecuado de la investigación, evitando la inclusión de elementos irrelevantes. Además, facilita el análisis de los datos en un marco temporal definido, asegurando resultados relevantes y precisos, de acuerdo con (Masters, 2023).

3.3 Instrumentos de Recolección de Datos

Se utilizó la observación directa ya que es un método de recolección de datos que radica esencialmente en prestar atención al objeto de estudio dentro de una situación particular, este método nos permitirá evaluar el comportamiento por un periodo de tiempo continuo para si arrojar resultados efectivos, además de darnos resultados tanto objetivos como subjetivos, (OkDiario, 2019).

3.4 Metodología de desarrollo de software

La metodología de desarrollo de software utilizada para mejorar la eficiencia en el proceso de desarrollo de software, se han vuelto cada vez más populares en el mundo del desarrollo de software. En estas metodologías, el proceso de desarrollo es iterativo e incremental. Los equipos de desarrollo pueden adaptarse a los cambios y a las necesidades de los interesados. Como lo es Scrum, siendo un marco de trabajo ágil que se enfoca en la colaboración, la adaptabilidad y la entrega iterativa del software. Se basa en principios como la transparencia, la inspección y la adaptación, se organiza en sprints, que son ciclos de trabajo cortos y enfocados, para que el equipo de desarrollo se autoorganice y trabaje en incrementos de funcionalidad del producto. (Scrum, 2023).

Extreme Programming (XP):

Extreme Programming es una metodología ágil que se centra en la calidad del software y la satisfacción del cliente. Promueve prácticas como la retroalimentación continua, la simplicidad, las pruebas automatizadas y la programación en parejas. XP busca adaptarse a los cambios de manera rápida y efectiva. (Alliance, 2023).

Kanban:

Kanban es una metodología ágil que se basa en el flujo continuo de trabajo. Utiliza un tablero Kanban para visualizar las tareas y limitar la cantidad de trabajo en progreso. Kanban se enfoca en optimizar el flujo de trabajo, identificar cuellos de botella y mejorar continuamente el proceso. (Martins, 2022).

Lean Software Development:

Lean Software Development se basa en los principios del sistema de producción Lean. Busca eliminar el desperdicio, maximizar el valor para el cliente y fomentar la colaboración. Se centra en la entrega temprana, la retroalimentación constante y la mejora continua del proceso de desarrollo. (Mendieta, 2022).

Dynamic Systems Development Method (DSDM):

DSDM es una metodología ágil que se enfoca en la entrega rápida de software. Utiliza principios como la participación activa del usuario, la entrega iterativa y el desarrollo incremental. DSDM proporciona un marco para la gestión de proyectos ágiles y se adapta a diferentes entornos y requisitos. (informacion, 2018).

Para este proyecto de investigación se utilizará la metodología de desarrollo de software Scrum debido a que es la más acorde al proyecto de investigación por su facilidad de implementación y esta es importante porque es beneficiosa en entornos complejos, caracterizados por plazos de entrega muy cortos y requisitos cambiantes o poco definidos. Es especialmente útil para abordar proyectos en los que se necesita adaptabilidad y se enfrentan desafíos constantes en la definición de los objetivos. SCRUM ofrece una estructura que permite responder rápidamente a los cambios y entregar resultados de manera iterativa, garantizando la satisfacción de las necesidades del cliente, (Esan, 2017).

4. Esquema Temático

En este capítulo describe los pasos seguidos para crear la aplicación web destinada al control y acceso a través de reconocimiento facial, para garantizar la seguridad de la Universidad Popular del Cesar Seccional Aguachica, centrándose en el estudio de Machine Learning. La investigación se llevó a cabo utilizando la metodología ágil de desarrollo Scrum. Se analizarán los procesos y resultados obtenidos en cada fase de implementación definida por esta metodología de desarrollo de software. El objetivo es lograr una página mucho más eficiente, integral y funcional que se adapte a los requerimientos del usuario que permita un reconocimiento óptimo y garantice los estándares básicos de seguridad en la institución.

4.1 Definir los requerimientos funcionales y no funcionales.

4.1.1 Análisis de requerimientos del sistema

Implica una evaluación detallada de las necesidades y expectativas específicas relacionadas con el desarrollo del software para el reconocimiento. Durante esta fase, se plantean los requisitos funcionales y no funcionales que guiarán el diseño, desarrollo y despliegue del sistema. Esto incluye la identificación y especificación de los objetivos de la aplicación, así como las funciones y capacidades requeridas para cumplir con esos objetivos. Se debe considerar la recopilación y preparación de datos necesarios para entrenar y validar el modelo del algoritmo, así como los requisitos de hardware y software para implementar y ejecutar eficientemente el entorno de producción.

4.1.1.1 Establecer ámbito y alcance del software

En esta fase crucial del proyecto, se delimita con precisión el alcance y los objetivos del sistema de reconocimiento facial. El propósito principal de este software es permitir la identificación precisa y eficiente de estudiantes universitarios a través del análisis de sus rostros. Para lograr este objetivo, el sistema se diseñará con una serie de funciones fundamentales:

- **Captura de imágenes faciales:** El software será capaz de adquirir imágenes de los rostros de los estudiantes utilizando cámaras disponibles en dispositivos móviles.
- **Comparación con base de datos de estudiantes autorizados:** Una vez que se capture la imagen facial, el sistema comparará la información obtenida con una base de datos previamente establecida que contiene las imágenes faciales y los datos de identificación de los estudiantes autorizados.
- **Autorización de acceso a instalaciones universitarias:** Basándose en los resultados de la comparación, el sistema determinará si el estudiante tiene acceso permitido a las instalaciones universitarias. En caso afirmativo, se le otorgará acceso; de lo contrario, se le denegará.

El enfoque principal del sistema estará en garantizar la precisión y la seguridad durante todo el proceso de reconocimiento facial. Se implementarán medidas de seguridad robustas para proteger la integridad de los datos biométricos de los estudiantes y para prevenir el acceso no autorizado al sistema. Además, se prestará especial atención al diseño de una interfaz de usuario intuitiva y amigable, tanto para estudiantes como para el personal de seguridad, con el objetivo de facilitar su uso y maximizar la eficiencia en el proceso de identificación y acceso a las instalaciones universitarias.

4.1.1.2 Identificación de los roles de usuario

Esta etapa implica la identificación de los diferentes roles de usuario que interactuarán con la aplicación de reconocimiento facial, así como la definición de sus privilegios y funciones dentro del sistema. Se han identificado los siguientes roles:

- **Administradores:** Tienen acceso completo al sistema y pueden administrar usuarios, agregar o eliminar estudiantes autorizados, supervisar el funcionamiento del sistema y generar informes sobre el uso y rendimiento
- **Estudiantes:** Los estudiantes son los usuarios finales del sistema. Su función principal es utilizar la aplicación para acceder a las instalaciones universitarias mediante el reconocimiento facial. Tienen privilegios limitados y solo pueden realizar acciones relacionadas con su propia identificación y acceso.
- **Personal de seguridad:** Este rol implica supervisar y mantener el sistema de reconocimiento facial. El personal de seguridad puede monitorear las actividades de acceso, recibir alertas en caso de anomalías, y proporcionar asistencia técnica en caso de problemas con el sistema.

Cada uno de estos roles tendrá diferentes niveles de acceso y privilegios dentro del sistema, asegurando que cada usuario pueda desempeñar sus funciones de manera efectiva y segura.

4.1.2 Diseño del aplicativo web y móvil

Se centra en la planificación y la estructuración de la aplicación tanto para su versión web como móvil. Se consideran diversos aspectos de diseño, incluyendo la experiencia del usuario (UX), la interfaz de usuario (UI), la arquitectura de la aplicación y la integración de características específicas. El objetivo es crear una experiencia de usuario intuitiva y atractiva, asegurando la coherencia entre las versiones web y móvil.

Se comienza definiendo los requisitos funcionales y no funcionales del sistema, los cuales servirán como base para el diseño. Luego, se exploran y seleccionan las arquitecturas de software más adecuadas para cada plataforma, teniendo en cuenta consideraciones como la escalabilidad, el rendimiento y la seguridad.

Al mismo tiempo, se desarrolla un diseño de interfaz de usuario (UI) que garantiza la accesibilidad, la usabilidad y la coherencia visual en ambas plataformas. Se incluyen elementos como diseño de pantallas, navegación,

interacciones y feedback del usuario. Por último, se establece la arquitectura de la aplicación, definiendo cómo se estructurarán los componentes y cómo se comunicarán entre sí. Se considera la separación de preocupaciones, el modularidad y la reutilización de código para facilitar el mantenimiento y la evolución futura del aplicativo.

4.1.2.1 Arquitecturas de referencia para una aplicación Web

En esta fase del diseño, se examinan diversas arquitecturas de software que pueden ser idóneas para la aplicación web y móvil de reconocimiento facial. Se considerarán cuidadosamente los siguientes aspectos:

- **Escalabilidad:** Se evaluarán las arquitecturas que permitan una escalabilidad eficiente del sistema para manejar un aumento en el número de usuarios y transacciones sin comprometer el rendimiento.
- **Rendimiento:** Se buscarán arquitecturas que garanticen un rendimiento óptimo, minimizando el tiempo de respuesta y maximizando la velocidad de procesamiento de las solicitudes.
- **Seguridad:** Se priorizarán las arquitecturas que integren medidas de seguridad sólidas para proteger los datos de los usuarios y garantizar la confidencialidad e integridad de la información sensible, como los datos biométricos utilizados en el reconocimiento facial.

Se llevará a cabo un análisis exhaustivo de las diferentes opciones arquitectónicas disponibles, considerando aspectos como la arquitectura monolítica, basada en microservicios, en la nube o serverless. Esta evaluación permitirá seleccionar la arquitectura más adecuada que cumpla con los requisitos de escalabilidad, rendimiento y seguridad de la aplicación web y móvil de reconocimiento facial.

Arquitecturas como referente

- **Arquitectura Monolítica:**

En esta arquitectura, toda la aplicación se desarrolla como un único y gran componente. Es adecuada para aplicaciones de tamaño pequeño a mediano con

requisitos simples y donde la escalabilidad no es una preocupación inmediata. Aunque puede ser más fácil de desarrollar e implementar inicialmente, puede volverse difícil de mantener y escalar a medida que la aplicación crece.

- **Arquitectura basada en Microservicios:**

Esta arquitectura divide la aplicación en pequeños servicios independientes que se comunican entre sí a través de interfaces bien definidas. Cada microservicio se encarga de una tarea específica, lo que facilita la escalabilidad y el despliegue independiente de cada componente. Es especialmente útil para aplicaciones complejas que requieren una alta escalabilidad y flexibilidad.

- **Arquitectura Serverless:**

En esta arquitectura, la infraestructura subyacente es gestionada por un proveedor de servicios en la nube, lo que permite a los desarrolladores centrarse únicamente en escribir y desplegar código. Es útil para cargas de trabajo intermitentes o impredecibles, ya que permite escalar automáticamente según la demanda y reduce los costos operativos.

- **Arquitectura de Contenedores (por ejemplo, Kubernetes):**

Esta arquitectura utiliza contenedores para encapsular y distribuir las aplicaciones y sus dependencias. Proporciona un entorno consistente y aislado para ejecutar la aplicación, lo que facilita su despliegue y escalabilidad. Kubernetes es una plataforma de orquestación de contenedores popular que facilita la gestión de grandes despliegues de contenedores en entornos de producción.

- **Arquitectura Distribuida:**

En esta arquitectura, las diferentes partes de la aplicación se ejecutan en múltiples dispositivos y se comunican entre sí a través de una red. Esta distribución de

componentes permite una mayor flexibilidad y escalabilidad en el desarrollo y operación de la aplicación. Cada componente puede ser desarrollado, desplegado y gestionado de forma independiente, lo que facilita la gestión de la carga de trabajo y la escalabilidad del sistema. La arquitectura distribuida es especialmente útil para aplicaciones que requieren un alto rendimiento, tolerancia a fallos y acceso distribuido a recursos.

Lenguajes de programación más utilizados para software administrativo:

Los lenguajes de programación más utilizados para desarrollar software administrativo son:

1. **PHP:** Un lenguaje de scripting ampliamente utilizado para el desarrollo web back-end, especialmente en la creación de sitios web dinámicos. Su simplicidad, flexibilidad y amplia comunidad lo convierten en una opción popular para CMS como WordPress, Joomla y Drupal.
2. **Python:** Un lenguaje de programación versátil y fácil de aprender, cada vez más utilizado en el desarrollo web back-end. Su legibilidad, escalabilidad y bibliotecas lo hacen adecuado para CMS como Django y Pyramid.
3. **Java:** Un lenguaje de programación robusto y confiable, popular para el desarrollo de aplicaciones empresariales. Su seguridad, rendimiento y portabilidad lo convierten en una buena opción para CMS como Alfresco y Liferay.
4. **JavaScript:** Un lenguaje esencial para el desarrollo web front-end, pero también se utiliza en el desarrollo back-end con frameworks como Node.js. Su versatilidad y comunidad activa lo hacen adecuado para CMS como Gatsby y Contentful.
5. **C#:** Un lenguaje de programación desarrollado por Microsoft, popular para el desarrollo de aplicaciones web y de escritorio con el framework .NET. Su escalabilidad, seguridad y facilidad de uso lo convierten en una buena opción para CMS como Umbraco y Orchard CMS.

Lenguajes de programación más utilizados para Inteligencia Artificial:

Los lenguajes de programación más utilizados para desarrollar aplicaciones de inteligencia artificial (IA) son:

1. **Python:** Un lenguaje de programación versátil y fácil de aprender, con una gran comunidad de desarrolladores en el ámbito de la IA. Sus bibliotecas especializadas como NumPy, TensorFlow y PyTorch lo convierten en una herramienta poderosa para el aprendizaje automático, el procesamiento del lenguaje natural y la visión artificial.
2. **R:** Un lenguaje de programación estadístico específicamente diseñado para el análisis de datos y la visualización. Su amplia gama de paquetes estadísticos y gráficos lo hace ideal para tareas de IA como el aprendizaje automático, la minería de datos y el análisis predictivo.
3. **Java:** Un lenguaje de programación robusto y escalable, popular para el desarrollo de grandes sistemas de IA. Su seguridad, rendimiento y portabilidad lo convierten en una buena opción para aplicaciones de IA que requieren un alto nivel de rendimiento y confiabilidad.
4. **C/C++:** Lenguajes de programación de bajo nivel que ofrecen un control preciso sobre el hardware y la memoria. Son utilizados para desarrollar aplicaciones de IA que requieren un alto rendimiento y eficiencia, como la robótica y los sistemas embebidos.
5. **JavaScript:** Un lenguaje de programación versátil que se utiliza tanto en el desarrollo web front-end como back-end. Con frameworks como TensorFlow.js y PyTorch.js, JavaScript permite desarrollar aplicaciones de IA que se ejecutan en el navegador web, como sistemas de reconocimiento facial y chatbots.

4.1.3 Requisitos Funcionales

Tabla 1. Requisitos funcionales Administrador

Requisito	Descripción del requisito
RFA1	El administrador tiene la opción de iniciar sesión en el sistema.
RFA2	El administrador tiene la capacidad de designar un rol a los usuarios que se han registrado.
RFA3	El administrador puede generar una lista de los usuarios.
RFA4	El administrador tiene la habilidad de realizar búsquedas de usuarios.
RFA5	El administrador puede modificar los permisos de cada rol en el sistema.
RFA6	El administrador tiene la capacidad de mostrar todos los registros actuales.
RFA7	El administrador debe ser capaz de enviar las imágenes recopiladas al servidor Python para su procesamiento.
RFA8	El administrador debe tener la capacidad de seleccionar imágenes específicas para entrenar el algoritmo de reconocimiento facial.
RFA9	El administrador puede registrar nuevas cuentas de usuario con privilegios de administrador.
RFA9	El administrador puede cerrar sesión.

Fuente: elaboración propia.

Tabla 2. Requisitos Funcionales Estudiantes

Requisito	Descripción del requisito
RFE1	El sistema verifica la identidad y crea un perfil único para el estudiante.
RFE2	El sistema detecta el rostro y realiza la comparación en tiempo real.
RFE3	La administración revisa y aprueba o deniega la solicitud.
RFE4	El sistema actualiza los permisos de acceso del estudiante en caso de aprobación.
RFE5	El sistema debe verificar la identidad del estudiante comparando la imagen capturada con los registros existentes en la base de datos universitaria.
RFE6	El sistema debe comparar el rostro detectado con las imágenes almacenadas en la base de datos en tiempo real para permitir o denegar el acceso.
RFE7	El sistema debe registrar todas las entradas y salidas del estudiante, almacenando la fecha, hora y ubicación de cada evento.

Fuente: elaboración propia.

Tabla 3. Requisitos Funcionales Personal de Seguridad

Requisito	Descripción del requisito
RFPS1	El sistema debe permitir al personal de seguridad monitorear en tiempo real los puntos de acceso mediante una interfaz que muestre las imágenes capturadas por las cámaras de reconocimiento facial y el estado de cada intento de acceso (permitido o denegado).
RFPS2	El sistema debe generar alertas inmediatas en caso de detectar intentos de acceso no autorizados o comportamientos sospechosos, notificando al personal de seguridad mediante sonidos, mensajes en pantalla y notificaciones móviles.

RFPS3	El personal de seguridad debe poder acceder a los perfiles de los usuarios en caso de necesitar verificar manualmente la identidad de una persona. Esto incluye la posibilidad de visualizar las imágenes faciales registradas y los datos personales relevantes.
RFPS4	El personal de seguridad debe poder registrar y documentar incidentes de seguridad relacionados con el control de acceso, incluyendo detalles como fecha, hora, descripción del incidente y acciones tomadas.
RFPS5	El sistema debe permitir la generación de informes detallados de incidentes, los cuales pueden ser revisados por la administración para tomar decisiones informadas sobre mejoras en la seguridad.

Fuente: elaboración propia.

4.1.4 Requisitos no funcionales

Tabla 2. Requisitos no funcionales del software administrativo

Requisito	Descripción del requisito
RFA1	El sistema debe garantizar la seguridad de los datos de imágenes y la información de los usuarios, utilizando técnicas de cifrado y autenticaciones robustas.
RFA2	El software debe ser capaz de manejar grandes volúmenes de imágenes y procesarlas eficientemente en un tiempo razonable.
RFA3	El sistema debe ser escalable para adaptarse a un mayor número de usuarios y a un aumento en la cantidad de imágenes procesadas
RFA4	La interfaz de usuario debe ser fácil de usar y estar diseñada de manera que los usuarios puedan realizar sus tareas de manera eficiente y sin dificultades.
RFA5	El sistema debe estar disponible en todo momento para que el administrador pueda acceder a él y realizar sus funciones cuando sea necesario.

RFA6	El software debe ser compatible con una variedad de dispositivos y sistemas operativos para garantizar su accesibilidad y utilidad.
RFA7	Mediante una codificación organizada, una documentación explícita y el empleo adecuado de herramientas para depuración y mantenimiento continuo, se busca facilitar la tarea de mantener y actualizar el sistema

Fuente: elaboración propia.

Tabla 3. Requisitos no funcionales del algoritmo de la IA en Python

Requisito	Descripción del requisito
RFA1	El sistema debe tener una alta precisión en la identificación y verificación de rostros para minimizar los falsos positivos y negativos.
RFA2	El sistema debe ser capaz de procesar imágenes y videos en tiempo real de manera eficiente, con tiempos de respuesta rápidos.
RFA3	El sistema debe poder manejar grandes volúmenes de datos y usuarios a medida que el sistema crece.
RFA4	El sistema debe ser resistente a variaciones en la iluminación, ángulos de la cara, expresiones faciales y cambios en la apariencia debido a factores como el maquillaje o el uso de accesorios.
RFA5	El sistema debe ser resistente a intentos de manipulación, como el uso de imágenes o videos falsificados
RFA6	El sistema debe ser fácil de mantener y actualizar, con una estructura de código clara y documentación completa.
RFA7	El sistema debe ser compatible con una variedad de hardware y sistemas operativos.
RFA8	El Sistema debe optimizar el uso de recursos de hardware para minimizar el consumo de energía, especialmente en dispositivos móviles o sistemas embebidos.

Fuente: elaboración propia.

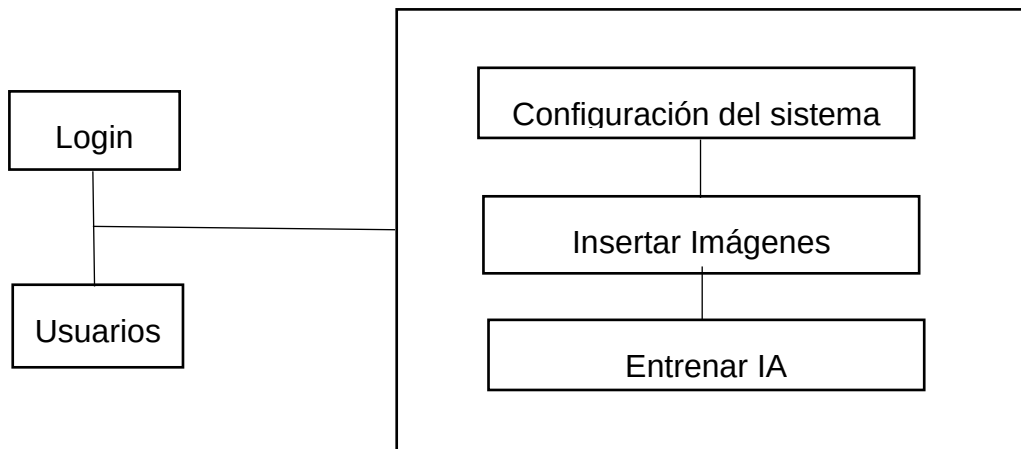
4.2 Propuesta de diseño del software de reconocimiento facial utilizando algoritmos, librerías basadas en IA y técnicas de Machine Learning

El diseño del software de reconocimiento facial está orientado a cumplir con la de requisitos funcionales que se establecieron para garantizar su eficacia y fiabilidad. Estos requisitos incluyen, entre otros, la precisión en la identificación, la velocidad de procesamiento, la capacidad de manejar grandes volúmenes de datos y la facilidad de integración con otros sistemas de seguridad.

4.2.1 Diseño de navegación administrador

En el diseño de la navegación del administrador del software de reconocimiento facial, se muestra la ruta de navegación que permitirá a los administradores gestionar y supervisar todas las funciones del sistema de manera integral. Este diseño incluirá un panel de control centralizado donde se podrán monitorear en tiempo real los accesos autorizados y no autorizados, revisar el estado de los dispositivos y cámaras de reconocimiento facial, y gestionar las bases de datos de usuarios.

Figura 2. Navegación ruta del administrador

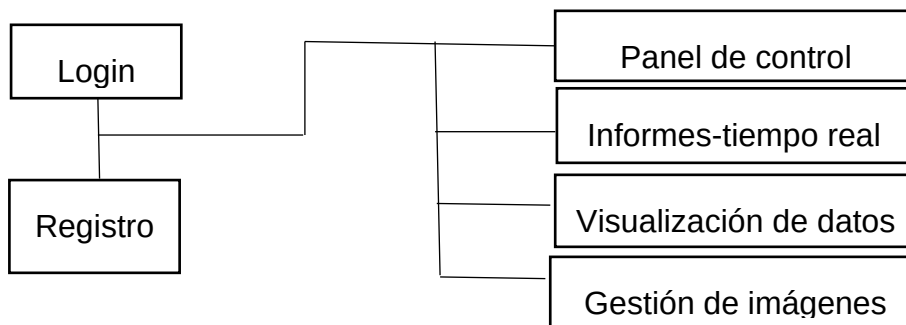


Fuente: elaboración propia.

4.2.2 Diseño de navegación estudiante

Los estudiantes tendrán acceso a su historial de accesos, pudiendo consultar fechas y horas de entrada y salida. Se integrarán notificaciones en tiempo real para informar sobre el estado de sus accesos y cualquier alerta relevante. Este diseño aprovechará algoritmos avanzados y técnicas de machine learning para asegurar una experiencia de usuario fluida y confiable, manteniendo la seguridad y privacidad de los datos biométricos en todo momento.

Figura 3. Navegación ruta estudiante

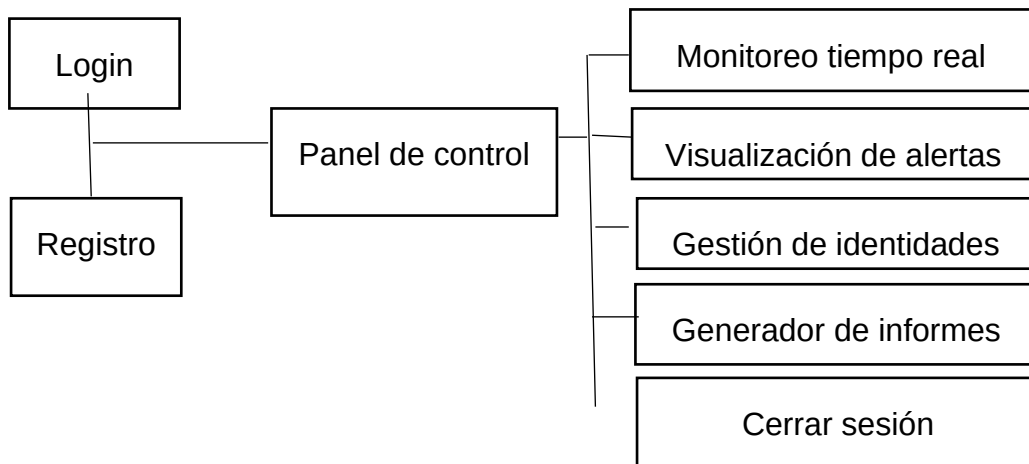


Fuente: elaboración propia.

4.2.3 Diseño de navegación personal de seguridad

Esta interfaz incluirá un panel de control centralizado desde el cual se podrán monitorear en tiempo real los puntos de acceso y las actividades de reconocimiento facial, visualizar alertas de acceso no autorizado y revisar registros detallados de eventos de seguridad. Por otra parte, el personal de seguridad tendrá acceso a herramientas para la verificación manual de identidades, la gestión de permisos de acceso y la configuración de políticas de seguridad. Se integrarán funcionalidades avanzadas basadas en algoritmos de reconocimiento de imágenes y técnicas de machine learning para detectar patrones sospechosos, que permitan generar informes automáticos sobre incidentes y actividades inusuales. Este diseño garantizará que el personal de seguridad pueda responder de manera rápida y precisa a cualquier amenaza, manteniendo la integridad y seguridad del entorno controlado.

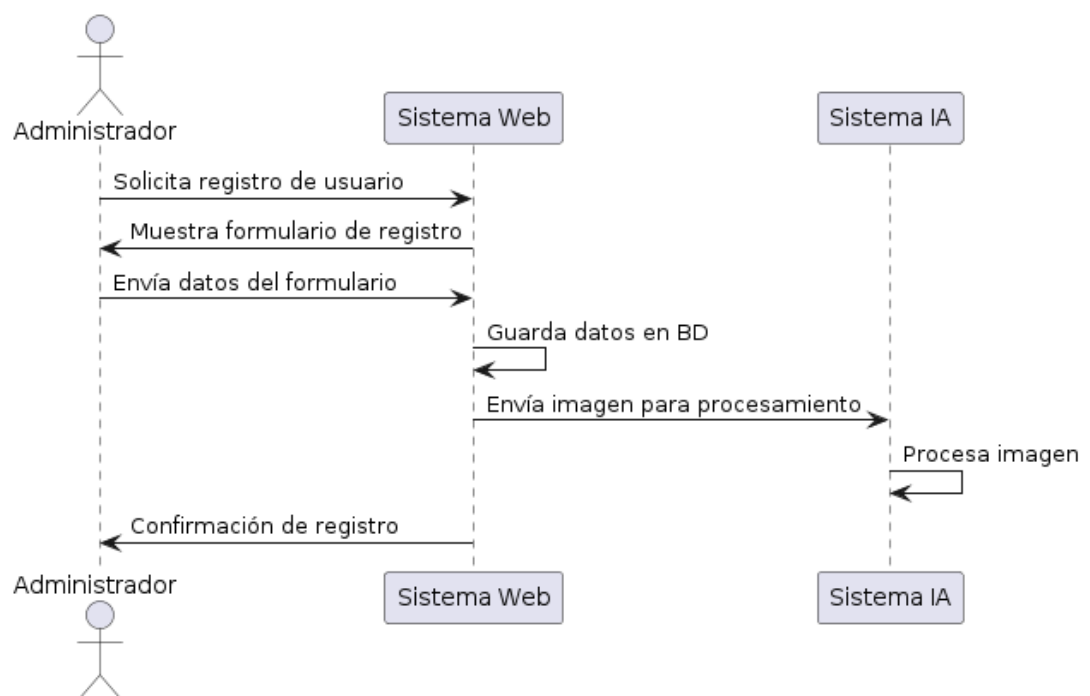
Figura 4. Navegación ruta personal de seguridad



Fuente: elaboración propia

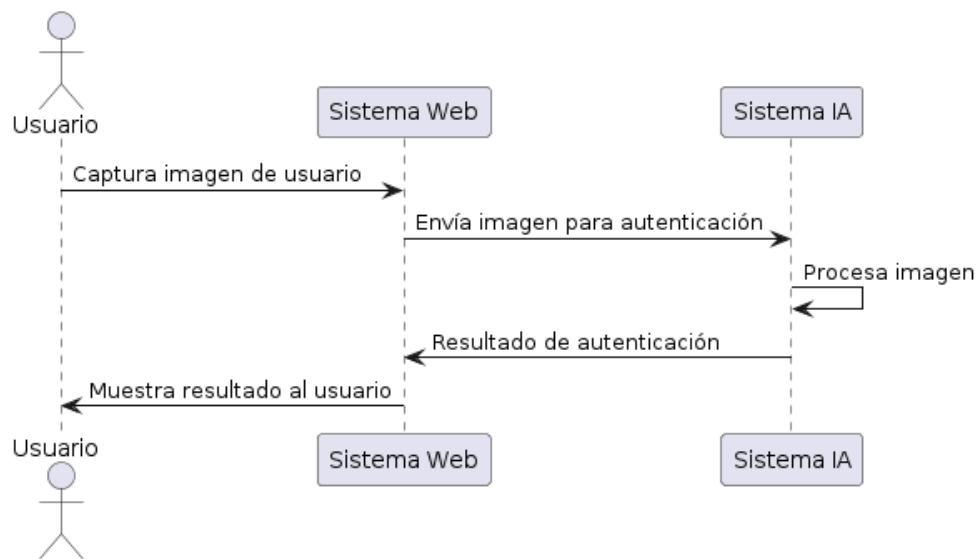
4.2.3.1 Diagrama de secuencia

Figura 5. Registro de Usuarios



Fuente: elaboración propia.

Figura 6. Autenticación de Usuarios

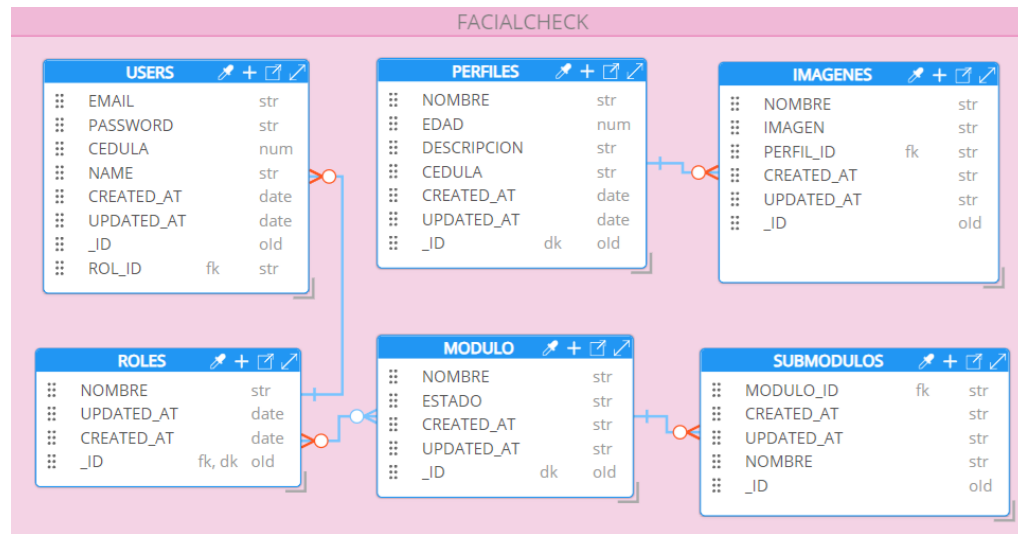


Fuente: elaboración propia.

4.2.4 Diseño del modelo entidad-relación

Para el desarrollo del software de reconocimiento facial, se planteó un modelo entidad-relación (ER) basado en los requisitos funcionales de seguridad previamente establecidos. Este modelo se estructura y organiza de manera eficiente como base de datos del sistema, definiendo claramente las entidades clave, como usuarios, perfiles de reconocimiento facial, registros de acceso y alertas de seguridad, así como las relaciones entre ellas. Este diseño permitirá una gestión de los datos operacionales, asegurando que el sistema funcione de manera eficiente y segura, y que pueda escalarse para satisfacer futuras necesidades de seguridad y rendimiento.

Figura 7. Modelo Entidad-Relación



Fuente: elaboración propia.

Para la implementación del software se plantea un enfoque moderno y eficiente, en el que se decidió utilizar el modelo entidad-relación (ER) previamente planteado en combinación con MongoDB y Laravel. Una base de datos NoSQL, permitirá almacenar y gestionar datos de manera flexible y escalable, lo cual es crucial para manejar grandes volúmenes de datos y registros de acceso en tiempo real. Su capacidad para manejar documentos JSON permite que el modelo ER se traduzca fácilmente en una estructura de datos dinámica y no rígida, facilitando la integración con los algoritmos de reconocimiento de patrones. A pesar de la naturaleza no relacional de MongoDB, se logra implementar relaciones entre los datos utilizando el ORM Eloquent de Laravel.

4.2.4.1 Especificaciones de las tecnologías

Debido a la necesidad de implementar reconocimiento facial, se desarrolló una solución integral utilizando diversas tecnologías especializadas. Para la inteligencia artificial y el procesamiento de imágenes, empleando Python junto con la biblioteca OpenCV, lo que nos permite realizar el reconocimiento facial de manera eficiente y precisa. Por otro lado, el sistema de administración ha sido desarrollado utilizando

el framework Laravel, conocido por su robustez y escalabilidad en el desarrollo de aplicaciones web.

Tecnologías para un software creado en Laravel

Tabla 4. Especificaciones tecnológicas

Tecnología	Descripción
Laravel Framework	Laravel es un framework de desarrollo web de código abierto para aplicaciones PHP. Está diseñado para facilitar y agilizar el proceso de desarrollo proporcionando una estructura y una sintaxis elegante.
Composer	Composer es un gestor de dependencias para PHP. Permite declarar las bibliotecas de las que depende un proyecto y las gestiona (instalación, actualización).
PHP	PHP es un lenguaje de programación de propósito general especialmente adecuado para el desarrollo web. Se ejecuta en el servidor y se utiliza para crear contenido dinámico en las páginas web.
MongoDB	MongoDB es una base de datos NoSQL orientada a documentos que almacena datos en formato JSON. Es altamente escalable y adecuada para manejar grandes volúmenes de datos.
Nginx	Nginx es un servidor web y proxy inverso de alto rendimiento, conocido por su escalabilidad y eficiencia en el manejo de cargas de trabajo concurrentes.

CSS3	CSS3 es la última evolución del lenguaje de hojas de estilo en cascada (CSS), utilizado para diseñar y presentar el contenido en la web.
Phyton	Python es un lenguaje de programación de alto nivel, interpretado y de propósito general, conocido por su legibilidad y simplicidad.
OpenCV	OpenCV (Open Source Computer Vision Library) es una biblioteca de software de visión artificial de código abierto, diseñada para el procesamiento de imágenes y análisis de video.
JavaScript	JavaScript es un lenguaje de programación interpretado, orientado a objetos y diseñado para crear contenido dinámico en las páginas web. Se ejecuta en el navegador del cliente.

Fuente: Elaboración propia

4.2.5 Modelado del sistema

El sistema está compuesto por dos componentes principales: un software de inteligencia artificial (IA) desarrollado en Python utilizando la biblioteca OpenCV, y una plataforma administrativa construida con Laravel. El software de IA se encarga de procesar y analizar las imágenes faciales de los usuarios, permitiendo su reconocimiento de manera rápida y precisa. La plataforma administrativa, por su parte, permite a los administradores cargar y gestionar las imágenes de los usuarios que serán utilizadas para entrenar el modelo de IA. A través de una interfaz intuitiva, los administradores pueden agregar, actualizar y eliminar imágenes, facilitando la administración de la base de datos de rostros y garantizando que el sistema de reconocimiento facial se mantenga actualizado y efectivo.

El flujo del sistema es sencillo y eficiente, el administrador carga las imágenes a través del panel de control, donde se almacenan de forma segura y se preparan para el procesamiento. Una vez cargadas, estas imágenes son enviadas al módulo de IA en Python con OpenCV, que las analiza y las integra en el modelo de reconocimiento facial. Cuando los usuarios intentan ser reconocidos, sus rostros son capturados y comparados con la base de datos existente, permitiendo una identificación precisa y rápida.

4.2.5.1 Casos de usos

Los casos de uso describen las interacciones entre los usuarios y el sistema para alcanzar objetivos específicos. Cada caso de uso se enfoca en una funcionalidad principal del sistema, describiendo las acciones realizadas por los usuarios y las respuestas del sistema. A continuación, se presentan los casos de uso para las funcionalidades principales del sistema de reconocimiento facial.

4.2.5.2 Descripción de casos de uso

Es un enfoque utilizado en el análisis y diseño de sistemas para representar las interacciones usuario-sistema a través de pasos ilustrativos y con nombres que conforman la interacción. Consiste en un flujo de casos de uso primario que detalla la secuencia de actividades clave y escenarios alternativos que actúan como ejemplos de excepciones, junto con postcondiciones (resultados esperados después de una finalización exitosa) y precondiciones (requisitos necesarios antes de la ejecución).

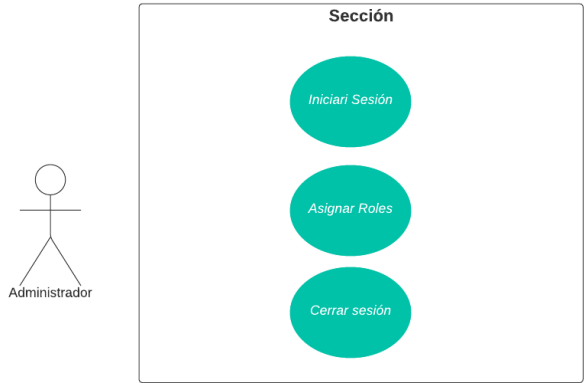
4.2.6 Casos de uso del sistema

Los casos de uso en esta sección demuestran las interacciones de roles establecidos con la plataforma cuando se utilizan acciones o funciones. Estos permiten a los usuarios utilizar eficazmente el sistema, por lo tanto, se proporciona un esquema de cada una de las estructuras de estos diagramas.

4.2.7 Diagramas UML

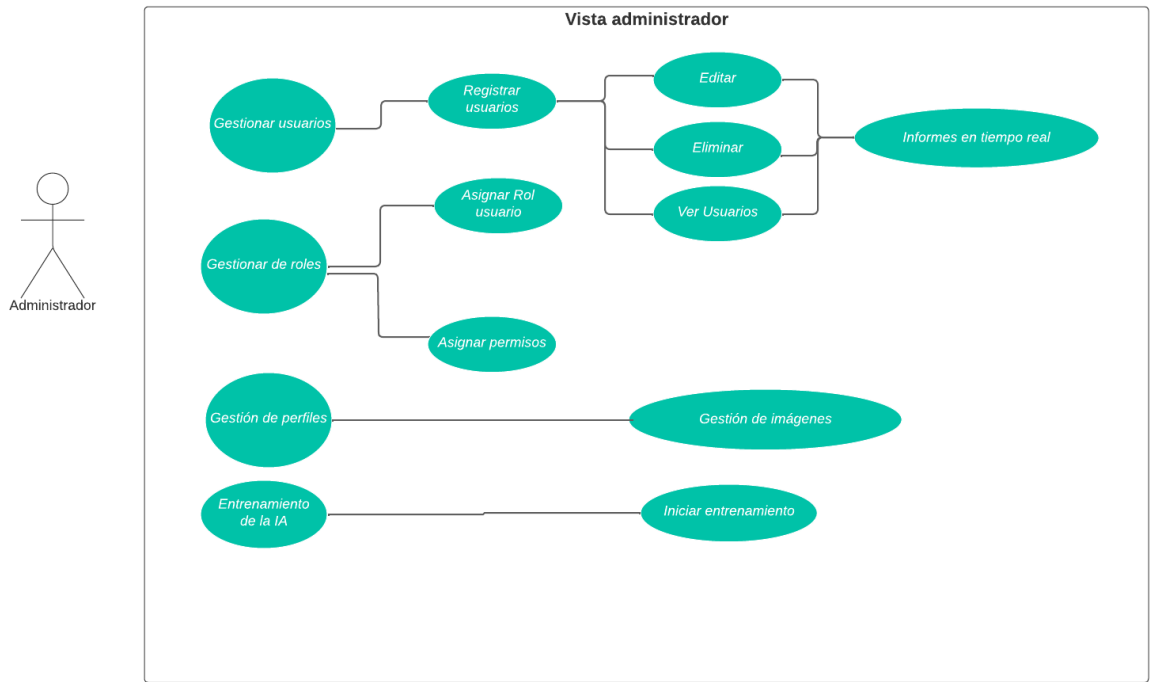
4.2.7.1 Caso de uso administrador

Figura 8. Login Administrador



Fuente: elaboración propia.

Figura 9. Descripción del caso de uso administrador



Fuente: elaboración propia.

Tabla 5. Descripción del caso de uso Login

ID Caso de uso	CU1 Login Administrador
Actores	Administrador
Propósito	Controlar el acceso a las personas. Solo las personas registradas podrán ingresar al sistema.
Condiciones	Se debe estar registrado en el sistema.

Fuente: autor del proyecto.

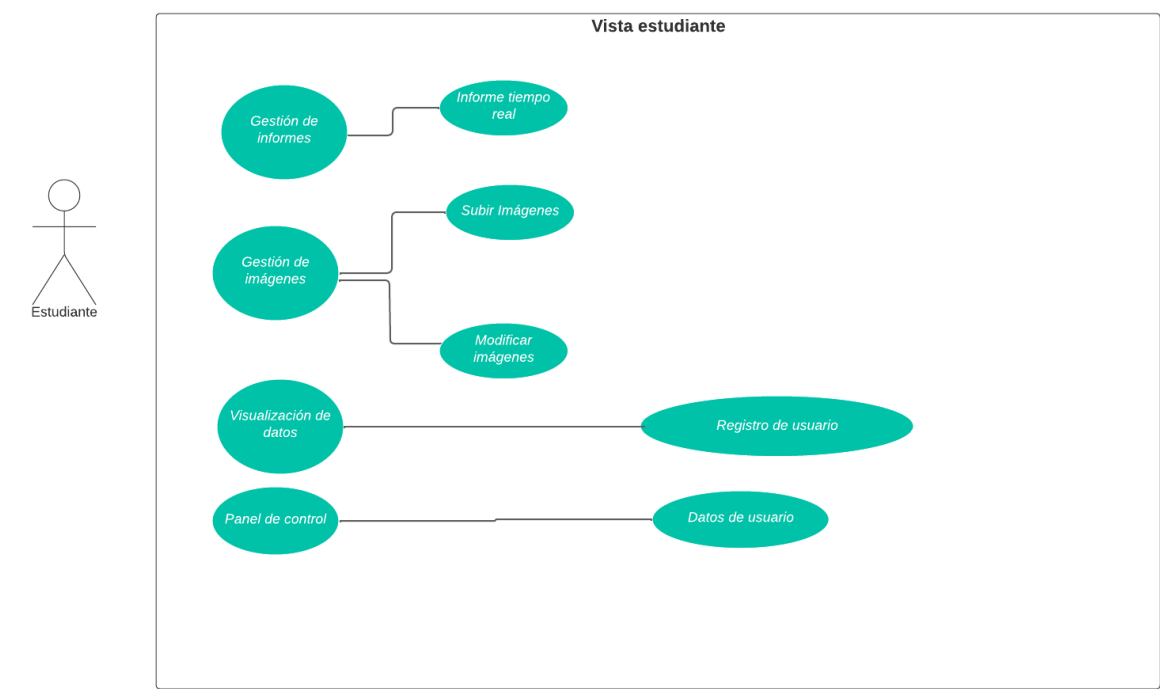
4.2.7.2 Caso de uso de Estudiante

Figura 10. Caso de uso de estudiante



Fuente: elaboración propia.

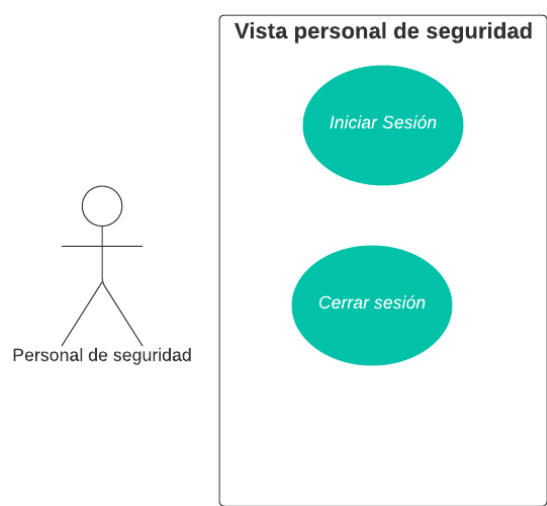
Figura 11. Panel de navegación estudiante



Fuente: elaboración propia.

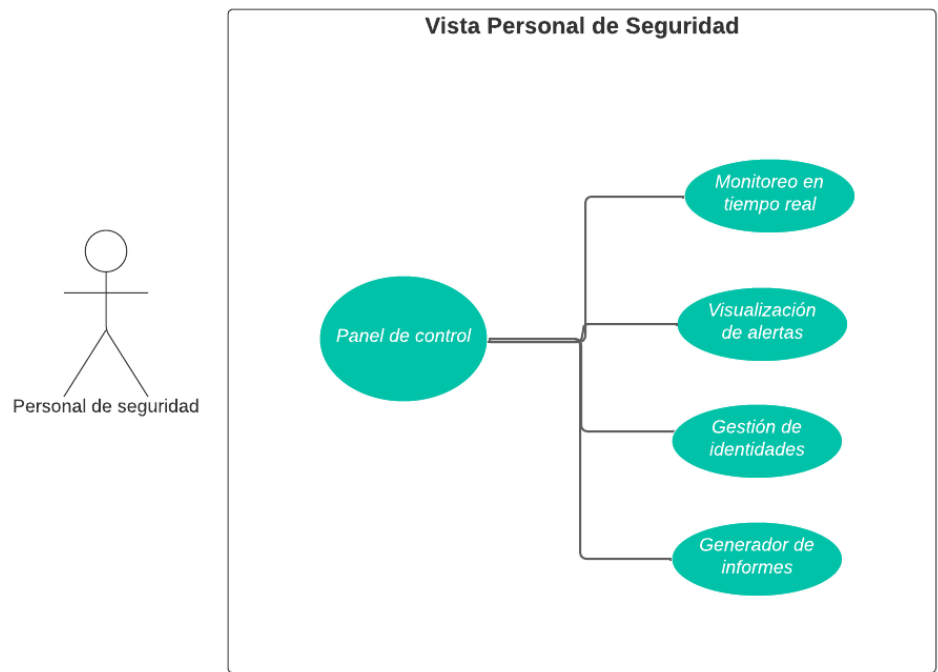
4.2.7.3 Caso de uso personal de seguridad

Figura 12. Caso de uso Personal de seguridad



Fuente: elaboración propia.

Figura 13. Panel de personal de seguridad



Fuente: elaboración propia.

4.2.8 Descripción de casos de uso

Cada caso de uso se describe en detalle, incluyendo los actores involucrados, el flujo principal de eventos, las precondiciones y las postcondiciones. Esto asegura que cada interacción del usuario con el sistema esté claramente definida.

Tabla 6. Descripción de caso de uso administrador

UML	Fecha:	Código: CU1
	Nombre:	Crear Usuario
	Descripción:	Gestión de Usuarios
	Actores:	Administrador
Precondiciones: El administrador debe estar autenticado en el sistema.		
Flujo Principal:		
- El administrador selecciona la opción "Usuarios" en el menú de navegación.		
- El sistema muestra las opciones para crear usuarios.		

Crear Usuario:

- El administrador selecciona "Crear Usuario".
- El sistema muestra un formulario para ingresar los datos del usuario (Documento, Email, Primer Nombre, Primer Apellido, contraseña, Rol).
- El administrador completa el formulario y selecciona "Guardar".
- El sistema valida los datos ingresados y guarda el usuario en la base de datos.
- El sistema muestra un mensaje de éxito.

Postcondiciones: Los cambios realizados de creación se reflejan en la base de datos y en la lista de usuarios disponibles

Fuente: elaboración propia.

Tabla 7. Descripción caso de uso Editar usuario

UML	Fecha:	Código: CU2
	Nombre:	Editar Usuario
	Descripción:	Gestión de Usuarios
	Actores:	Administrador

Precondiciones: El administrador debe estar autenticado en el sistema.

Flujo Principal:

- El administrador selecciona la opción "Usuarios" en el menú de navegación.
- El sistema muestra las opciones para editar usuarios.

Editar Usuario:

- El administrador selecciona un usuario de la lista.
- El sistema muestra los detalles del usuario en un formulario editable.
- El administrador actualiza los datos y selecciona "Guardar".
- El sistema valida los cambios y actualiza el usuario en la base de datos.
- El sistema muestra un mensaje de éxito.

Postcondiciones: Los cambios realizados de edición se reflejan en la base de datos y en la lista de usuarios disponibles

Fuente: elaboración propia.

Tabla 8. Descripción caso de uso eliminar usuario

UML	Fecha:	Código: CU3
	Nombre:	Eliminar Usuario
	Descripción:	Gestión de Usuarios
	Actores:	Administrador

Precondiciones: El administrador debe estar autenticado en el sistema.

Flujo Principal:

- El administrador selecciona la opción "Usuarios" en el menú de navegación.
- El sistema muestra las opciones para eliminar usuarios.

Eliminar Usuario:

- El administrador selecciona un usuario de la lista.
- El sistema muestra una confirmación de eliminación.
- El administrador confirma la eliminación.
- El sistema elimina el usuario de la base de datos.
- El sistema muestra un mensaje de éxito.

Postcondiciones: Los cambios realizados de eliminación se reflejan en la base de datos y en la lista de usuarios disponibles

Fuente: elaboración propia.

Tabla 9. Descripción caso de uso listar usuarios

UML	Fecha:	Código: CU4
	Nombre:	Ver Usuario
	Descripción:	Gestión de Usuarios
	Actores:	Administrador

Precondiciones: El administrador debe estar autenticado en el sistema.

Flujo Principal:

- El administrador selecciona la opción "Usuarios" en el menú de navegación.
 - El sistema muestra las opciones para ver usuarios.
-

Ver Usuario:

El administrador selecciona un usuario de la lista.

El sistema muestra los detalles del usuario en una vista de solo lectura.

Postcondiciones: Los cambios realizados de vista se reflejan en la base de datos y en la lista de usuarios disponibles

Fuente: elaboración propia.

Tabla 10. Descripción caso de uso Asignar Roles

UML	Fecha:	Código: CU5
	Nombre:	Asignar Rol a Usuario
	Descripción:	Gestión de Usuarios
	Actores:	Administrador

Precondiciones: El administrador debe estar autenticado en el sistema.

Flujo Principal:

- El administrador selecciona la opción "Usuarios" en el menú de navegación.
 - El sistema muestra las opciones para Asignar Rol a Usuario.
-

Asignar Rol a usuario:

El administrador da un rol al usuario dependiendo los permisos que este tendrá en el sistema.

Postcondiciones: Los cambios realizados de asignación de rol a usuario se reflejan en la base de datos y en la lista de usuarios disponibles

Fuente: elaboración propia.

Tabla 11. Descripción caso de uso crear rol

UML	Fecha:	Código: CU6
	Nombre:	Crear Rol
	Descripción:	Gestión de Roles
	Actores:	Administrador
Precondiciones: El administrador debe estar autenticado en el sistema.		
Flujo Principal:		
- El administrador selecciona la opción "Roles" en el menú de navegación.		
- El sistema muestra las opciones para crear roles.		
Crear Rol:		
- El administrador selecciona "Crear Rol".		
- El sistema muestra un formulario para ingresar los datos del rol (Nombre del Rol, Descripción, Submódulos).		
- El administrador completa el formulario y selecciona "Guardar".		
- El sistema valida los datos ingresados y guarda el rol en la base de datos.		
- El sistema muestra un mensaje de éxito.		
Postcondiciones: Los cambios realizados de creación se reflejan en la base de datos y en la lista de roles disponibles.		

Fuente: elaboración propia.

Tabla 12. Descripción de caso de uso editar rol

UML	Fecha:	Código: CU7
	Nombre:	Editar Rol
	Descripción:	Gestión de Roles
	Actores:	Administrador
Precondiciones: El administrador debe estar autenticado en el sistema.		
Flujo Principal:		
- El administrador selecciona la opción "Roles" en el menú de navegación.		
- El sistema muestra las opciones para editar roles.		
Editar Rol:		
- El administrador selecciona un rol de la lista.		
- El sistema muestra los detalles del rol en un formulario editable.		
- El administrador actualiza los datos y selecciona "Guardar".		
- El sistema valida los cambios y actualiza el rol en la base de datos.		
- El sistema muestra un mensaje de éxito.		
Postcondiciones: Los cambios realizados de eliminación se reflejan en la base de datos y en la lista de roles disponibles.		

Fuente: elaboración propia.

Tabla 13. Descripción de caso de uso ver rol

UML	Fecha:	Código: CU8
	Nombre:	Ver Rol
	Descripción:	Gestión de Roles
	Actores:	Administrador
Precondiciones: El administrador debe estar autenticado en el sistema.		
Flujo Principal:		
- El administrador selecciona la opción "Roles" en el menú de navegación.		
- El sistema muestra las opciones para ver roles.		
Ver Rol:		
- El administrador selecciona un rol de la lista.		
- El sistema muestra los detalles del rol en una vista de solo lectura.		
Postcondiciones: Los cambios realizados de vista se reflejan en la base de datos y en la lista de roles disponibles.		

Fuente: elaboración propia.

Tabla 14. Descripción caso de uso Asignar permisos a rol

UML	Fecha:	Código: CU9
	Nombre:	Asignar permisos a rol
	Descripción:	Gestión de Roles
	Actores:	Administrador
Precondiciones: El administrador debe estar autenticado en el sistema.		
Flujo Principal:		
- El administrador selecciona la opción "Roles" en el menú de navegación.		
- El sistema muestra las opciones para Asignar permisos a roles.		
Asignar Permisos a Rol		
El administrador asigna los módulos que puede visualizar un rol al momento de crearlo.		
Postcondiciones: Los cambios realizados de asignación de permisos a roles se reflejan en la base de datos y en la lista de roles disponibles.		

Fuente: elaboración propia.

Tabla 15. Descripción caso de uso Crear perfil

UML	Fecha:	Código: CU10
	Nombre:	Crear perfil
	Descripción:	Gestión de Perfiles
	Actores:	Administrador

Precondiciones: El administrador debe estar autenticado en el sistema.

Flujo Principal:

- El administrador selecciona la opción "Perfiles" en el menú de navegación.
 - El sistema muestra las opciones para crear perfiles.
-

Crear Perfil:

- El administrador selecciona "Crear Perfil".
 - El sistema muestra un formulario para ingresar los datos del perfil (Nombre, Cédula, Edad, Descripción, Email).
 - El administrador completa el formulario y selecciona "Guardar".
 - El sistema valida los datos ingresados y guarda el perfil en la base de datos.
 - El sistema muestra un mensaje de éxito.
-

Postcondiciones: Los cambios realizados de creación se reflejan en la base de datos y en la lista de perfiles disponibles.

Fuente: elaboración propia.

Tabla 16. Descripción caso de uso editar perfil

UML	Fecha:		Código: CU11
	Nombre:	Editar perfil	
	Descripción:	Gestión de Perfiles	
	Actores:	Administrador	

Precondiciones: El administrador debe estar autenticado en el sistema.

Flujo Principal:

- El administrador selecciona la opción "Perfiles" en el menú de navegación.
 - El sistema muestra las opciones para editar perfiles.
-

Editar Perfil:

- El administrador selecciona un perfil de la lista.
 - El sistema muestra los detalles del perfil en un formulario editable.
 - El administrador actualiza los datos y selecciona "Guardar".
 - El sistema valida los cambios y actualiza el perfil en la base de datos.
 - El sistema muestra un mensaje de éxito.
-

Postcondiciones: Los cambios realizados de edición se reflejan en la base de datos y en la lista de perfiles disponibles.

Fuente: elaboración propia.

Tabla 17. Descripción caso de uso eliminar perfil

UML	Fecha:		Código: CU12
	Nombre:	Eliminar perfil	

Descripción:	Gestión de Perfiles
Actores:	Administrador
Precondiciones: El administrador debe estar autenticado en el sistema.	
Flujo Principal:	
- El administrador selecciona la opción "Perfiles" en el menú de navegación.	
- El sistema muestra las opciones para eliminar perfiles.	
Eliminar Perfil:	
- El administrador selecciona un perfil de la lista.	
- El sistema muestra una confirmación de eliminación.	
- El administrador confirma la eliminación.	
- El sistema elimina el perfil de la base de datos.	
- El sistema muestra un mensaje de éxito.	
Postcondiciones: Los cambios realizados de eliminación se reflejan en la base de datos y en la lista de perfiles disponibles.	

Fuente: elaboración propia.

Tabla 18. Descripción caso de uso ver perfil

UML	Fecha:	Código: CU13
	Nombre:	ver perfil
	Descripción:	Gestión de Perfiles
	Actores:	Administrador
Precondiciones: El administrador debe estar autenticado en el sistema.		
Flujo Principal:		
- El administrador selecciona la opción "Perfiles" en el menú de navegación.		
- El sistema muestra las opciones para ver perfiles.		
Ver Perfil:		
- El administrador selecciona un perfil de la lista.		
- El sistema muestra los detalles del perfil en una vista de solo lectura.		
Postcondiciones: Los cambios realizados de vista se reflejan en la base de datos y en la lista de perfiles disponibles.		

Fuente: elaboración propia.

Tabla 19. Descripción de caso de uso Cargar imágenes a perfil

UML	Fecha:	Código: CU14
	Nombre:	Cargar Imágenes a perfil
	Descripción:	Gestión de Imágenes
	Actores:	Administrador
Precondiciones:		
- El administrador debe estar autenticado en el sistema.		
- Debe existir al menos un perfil creado.		
Flujo Principal:		
- El administrador selecciona la opción "Imágenes" en el menú de navegación.		

- El sistema muestra las opciones para cargar imágenes.

Cargar Imágenes:

- El administrador selecciona "Cargar Imágenes".
 - El sistema muestra una interfaz para seleccionar y cargar imágenes desde el dispositivo.
 - El administrador selecciona una imagen y selecciona "Cargar".
 - El sistema valida las imágenes y las guarda en el servidor.
 - El sistema muestra un mensaje de éxito.
-

Postcondiciones: Las imágenes cargadas se reflejan en el sistema.

Fuente: elaboración propia.

Tabla 20. Descripción caso de uso Asociar imágenes a perfil

UML	Fecha:	Código: CU15
	Nombre:	Asociar imágenes a perfil
	Descripción:	Gestión de Imágenes
	Actores:	Administrador
Precondiciones:		
- El administrador debe estar autenticado en el sistema. - Debe existir al menos un perfil creado.		
Flujo Principal:		
- El administrador selecciona la opción "Imágenes" en el menú de navegación. - El sistema muestra las opciones para asociar imágenes.		
Asociar Imágenes a Perfil:		
- El administrador selecciona un perfil desde el select que aparece en el formulario. - El sistema muestra una interfaz para asociar una imagen al perfil seleccionado. - El administrador selecciona las imágenes y confirma la asociación. - El sistema guarda la asociación en la base de datos. - El sistema muestra un mensaje de éxito.		
Postcondiciones: Las imágenes asociadas se reflejan en el sistema.		

Fuente: elaboración propia.

Tabla 21. Descripción caso de uso Eliminar imágenes a perfil

UML	Fecha:	Código: CU16
	Nombre:	Eliminar imágenes a perfil
	Descripción:	Gestión de Imágenes
	Actores:	Administrador
Precondiciones:		
- El administrador debe estar autenticado en el sistema. - Debe existir al menos un perfil creado.		
Flujo Principal:		
El administrador selecciona la opción "Imágenes" en el menú de navegación.		

El sistema muestra las opciones para eliminar imágenes.
Eliminar Imágenes:
El administrador selecciona una imagen de la lista.
El sistema muestra una confirmación de eliminación.
El administrador confirma la eliminación.
El sistema elimina la imagen del servidor y la base de datos.
El sistema muestra un mensaje de éxito.
Postcondiciones: Las imágenes eliminadas se eliminan del servidor y la base de datos.

Fuente: elaboración propia.

Tabla 22. Descripción caso de uso Iniciar entrenamiento

UML	Fecha:	Código: CU17
	Nombre:	Iniciar entrenamiento
	Descripción:	Entrenamiento de la IA
	Actores:	Administrador
Precondiciones:		
- El administrador debe estar autenticado en el sistema.		
- Deben existir perfiles con imágenes asociadas.		
Flujo Principal:		
- El administrador selecciona la opción "Entrenar" en el menú de navegación.		
- El sistema muestra un botón para iniciar el entrenamiento.		
Iniciar Entrenamiento:		
- El administrador selecciona "Iniciar Entrenamiento".		
- El sistema inicia el proceso de entrenamiento de la IA utilizando las imágenes enviadas.		
- Una vez finalizado el entrenamiento se muestra una confirmación de éxito		
Postcondiciones:		
- El modelo de IA es entrenado con las imágenes disponibles.		
- El sistema almacena el modelo entrenado para su uso en el reconocimiento facial.		

Fuente: elaboración propia.

Tabla 23. Descripción de caso de uso Iniciar entrenamiento

UML	Fecha:	Código: CU18
	Nombre:	Iniciar entrenamiento
	Descripción:	Reconocimiento Facial
	Actores:	Usuarios, Sistema
Precondiciones:		
- El sistema de reconocimiento facial debe estar entrenado y funcionando correctamente.		
- El perfil del usuario debe estar previamente registrado y asociado con imágenes en la		

base de datos.

Flujo Principal:

- El usuario se presenta ante la cámara del sistema.
 - El sistema captura una imagen en tiempo real del rostro del usuario.
 - El sistema de reconocimiento facial procesa la imagen capturada.
 - El sistema compara la imagen capturada con las imágenes almacenadas en la base de datos.
 - El sistema determina si hay una coincidencia entre la imagen capturada y algún perfil registrado.
 - Si se encuentra una coincidencia, el sistema autentica al usuario y concede acceso.
 - El sistema muestra un mensaje de éxito y permite al usuario acceder.
 - Si no se encuentra una coincidencia, el sistema muestra un mensaje de error indicando que el reconocimiento ha fallado.
-

Postcondiciones:

- Si el reconocimiento es exitoso, el usuario es autenticado y puede acceder.
 - Si el reconocimiento falla, el usuario no es autenticado y no puede acceder.
-

Fuente: elaboración propia.

4.2.9 Código de la Inteligencia Artificial (IA)

El entrenamiento del modelo implica ajustar los parámetros del algoritmo utilizando el conjunto de datos de entrenamiento. Durante el entrenamiento, el modelo aprende a realizar predicciones precisas a partir de los datos. Este proceso puede requerir ajustes iterativos y la utilización de técnicas como la validación cruzada para optimizar el rendimiento del modelo. Una vez que el modelo ha sido entrenado, es crucial evaluar su rendimiento utilizando el conjunto de datos de validación y prueba. Las métricas de evaluación, como la precisión, la exactitud, la sensibilidad y la especificidad, proporcionan información sobre el desempeño del modelo. Además, es importante realizar pruebas de robustez para asegurar que el modelo funcione bien en diferentes condiciones y con datos nuevos.

Importación de Librerías: El primer paso en el código es importar las librerías necesarias para el reconocimiento facial.

Figura 14. Librerías Python

```
import cv2
import numpy as np
from sklearn.neighbors import KNeighborsClassifier
from sklearn.svm import SVC
import os
import joblib
from django.http import HttpResponse
from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.metrics import accuracy_score
```

Fuente: elaboración propia.

Carga de Imágenes y Entrenamiento: El sistema carga las imágenes de los usuarios y las procesa para el entrenamiento.

Figura 15. Cargar imágenes

```
def cargar_imagenes_y_etiquetas(ruta_dataset):
    imagenes = []
    etiquetas = []

    for nombre_persona in os.listdir(ruta_dataset):
        carpeta_persona = os.path.join(ruta_dataset, nombre_persona)

        for imagen_nombre in os.listdir(carpeta_persona):
            imagen_ruta = os.path.join(carpeta_persona, imagen_nombre)
            imagen = cv2.imread(imagen_ruta, cv2.IMREAD_GRAYSCALE)
            imagen = cv2.resize(imagen, (100, 100))
            imagenes.append(imagen)
            etiquetas.append(nombre_persona)

        # Aumento de datos
        imagen_flip = cv2.flip(imagen, 1)
        imagenes.append(imagen_flip)
        etiquetas.append(nombre_persona)

        rows, cols = imagen.shape
        for angulo in range(-20, 21, 5):
            M = cv2.getRotationMatrix2D((cols / 2, rows / 2), angulo, 1)
            imagen_rotada = cv2.warpAffine(imagen, M, (cols, rows))
            imagenes.append(imagen_rotada)
            etiquetas.append(nombre_persona)

        imagen_desenfocada = cv2.GaussianBlur(imagen, (5, 5), 0)
        imagenes.append(imagen_desenfocada)
        etiquetas.append(nombre_persona)

        mean = 0
        var = 0.1 * 255
        sigma = var ** 0.5
        gaussian = np.random.normal(mean, sigma, imagen.shape)
        imagen_con_ruido = imagen + gaussian
        imagenes.append(imagen_con_ruido)
        etiquetas.append(nombre_persona)
```

Fuente: elaboración propia.

Figura 16. Entrenamiento de la IA

```
def entrenamiento(request):
    ruta_dataset = 'dataset/'
    imagenes, etiquetas = cargar_imagenes_y_etiquetas(ruta_dataset)

    imagenes = np.array(imagenes)
    etiquetas = np.array(etiquetas)

    imagenes = imagenes.astype('float32') / 255.0

    X_train, X_test, y_train, y_test = train_test_split(imagenes.reshape(len(imagenes), -1), etiquetas, test_size=0.2, random_state=42)

    # Entrenamiento y evaluación del clasificador KNN
    knn_clf = KNeighborsClassifier(n_neighbors=5, weights='distance', metric='euclidean')
    knn_clf.fit(X_train, y_train)
    knn_scores = cross_val_score(knn_clf, X_train, y_train, cv=5)
    knn_accuracy = accuracy_score(y_test, knn_clf.predict(X_test))

    # Guardar modelo KNN
    joblib.dump(knn_clf, 'modelo_knn_con_aumento_con_desconocido.pkl')

    # Entrenamiento y evaluación del clasificador SVM
    svm_clf = SVC(kernel='rbf', C=1.0, gamma='scale')
    svm_clf.fit(X_train, y_train)
    svm_scores = cross_val_score(svm_clf, X_train, y_train, cv=5)
    svm_accuracy = accuracy_score(y_test, svm_clf.predict(X_test))

    # Guardar modelo SVM
    joblib.dump(svm_clf, 'modelo_svm_con_aumento_con_desconocido.pkl')

    response = f"Entrenamiento completado.<br>KNN Accuracy: {knn_accuracy:.2f}<br>SVM Accuracy: {svm_accuracy:.2f}"

    return HttpResponse(response)
```

Fuente: elaboración propia.

Reconocimiento Facial: Utiliza las imágenes entrenadas para reconocer caras en tiempo real.

Figura 17. Código reconocimiento facial

```
@csrf_exempt
def reconocimiento_facial(request):
    if request.method == 'POST' and request.FILES.get('image'):
        # Obtener la imagen de la solicitud POST
        image = request.FILES['image']

        # Cargar los modelos entrenados (KNN y SVM)
        knn_clf = joblib.load('modelo_knn_con_aumento_con_desconocido.pkl')
        svm_clf = joblib.load('modelo_svm_con_aumento_con_desconocido.pkl')

        # Convertir la imagen a una matriz numpy
        nparr = np.frombuffer(image.read(), np.uint8)
        img = cv2.imdecode(nparr, cv2.IMREAD_COLOR)

        # Convertir la imagen a escala de grises
        gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

        # Mejorar el contraste de la imagen
        gray = cv2.equalizeHist(gray)

        # Inicializar el clasificador de rostros en cascada de Haar
        face_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + 'haarcascade_frontalface_default.xml')

        resultados = []

        # Detectar rostros en la imagen con parámetros ajustados
        faces = face_cascade.detectMultiScale(gray, scaleFactor=1.1, minNeighbors=7, minSize=(30, 30))

        for (x, y, w, h) in faces:
            roi_gray = gray[y:y+h, x:x+w]
            roi_gray_resized = cv2.resize(roi_gray, (100, 100))

            # Aplicar preprocesamiento
            roi_gray_resized = roi_gray_resized.astype('float32') / 255.0
            roi_gray_resized_flattened = roi_gray_resized.flatten().reshape(1, -1)

            # Hacer predicción con KNN
            knn_prob = knn_clf.predict_proba(roi_gray_resized_flattened)
            knn_label = knn_clf.classes_[np.argmax(knn_prob)]
            knn_confidence = np.max(knn_prob)
```

Fuente: elaboración propia.

4.2.9.1 Librerías usadas

Tabla 24. Librerías usadas para el diseño del software

Librería	Descripción	Uso en el proyecto	Captura
OpenCV	Biblioteca de visión artificial que permite el procesamiento de imágenes y videos.	Captura de video y dibujado de rectángulos alrededor de las caras detectadas.	<pre>import cv2</pre>
NumPy	Biblioteca fundamental para computación científica en Python.	Manipulación y procesamiento de datos de imágenes.	<pre>import numpy as np</pre>
Scikit-learn	Biblioteca de aprendizaje automático que proporciona herramientas simples y eficientes para análisis de datos y minería de datos.	Implementación de algoritmos de clasificación para el reconocimiento facial.	<pre>from sklearn.neighbors import KNeighborsClassifier from sklearn.svm import SVC from sklearn.model_selection import train_test_split, cross_val_score from sklearn.metrics import accuracy_score</pre>
OS	Biblioteca estándar de Python que proporciona una forma portátil de usar funcionalidades dependientes del sistema operativo.	Acceso a funcionalidades del sistema operativo como leer o escribir archivos.	<pre>import os</pre>
Joblib	Biblioteca para serialización y deserialización eficiente de objetos Python, especialmente para objetos de	Guardar y cargar modelos de reconocimiento facial entrenados.	<pre>import joblib</pre>

	gran tamaño como los modelos de aprendizaje automático.	
Django	Framework de alto nivel para el desarrollo de aplicaciones web en Python.	Responder a solicitudes HTTP de la aplicación web que utiliza el reconocimiento facial para procesar y devolver resultados.

```
from django.http import HttpResponse
```

Fuente: elaboración propia.

4.3 Construir e integrar a una interfaz de usuario al software de reconocimiento facial según el diseño establecido.

4.3.1 Planificación y estimación del diseño de la interfaz de usuario

Es crucial para definir el alcance del proyecto, los recursos necesarios y el cronograma de ejecución. En esta fase, se establecen los objetivos específicos del sistema, se identifican los requisitos técnicos y se elaboran proyecciones detalladas sobre el tiempo y ejecución de la investigación. La planificación precisa es fundamental para asegurar el éxito del proyecto, minimizando riesgos y optimizando el uso de recursos. La estimación del tiempo es una parte esencial, donde se elabora un cronograma detallado que divida el proyecto en fases más pequeñas y manejables, como el diseño, el desarrollo, las pruebas y la implementación. Cada historia de usuario tiene plazos específicos y entregables claramente definidos.

4.3.1.1 Crear y estimar historias de usuario

La creación y estimación de historias de usuario es una técnica fundamental en la gestión ágil de proyectos, que permite definir claramente los requisitos desde la perspectiva del usuario final. Estas historias de usuario ayudan a estructurar el desarrollo del sistema en funcionalidades específicas y comprensibles, facilitando la planificación y la priorización de tareas.

Tabla 25. Historias de usuario

Historias de usuario				
Épica	ID	Descripción	Estado	Prioridad
EP001	ID01	Como usuario, quiero poder registrar mi rostro en el sistema para que pueda ser reconocido.	Pendiente	Alta
EP002	ID02	Como usuario, quiero recibir una confirmación después de registrar mi rostro.	Pendiente	Media
EP003	ID03	Como administrador, quiero poder agregar nuevos perfiles de usuario para mantener la base de datos actualizada.	Pendiente	Alta
EP004	ID04	Como administrador, quiero poder actualizar los perfiles de usuario existentes.	Pendiente	Alta
EP005	ID05	Como administrador, quiero poder eliminar perfiles de usuario no deseados.	Pendiente	Media
EP006	ID06	Como usuario, quiero que mi acceso sea protegido mediante autenticación de dos factores (2FA).	Pendiente	Alta
EP007	ID07	Como administrador, quiero poder ver el historial de inicios de sesión de los usuarios.	Pendiente	Media

EP008	ID08	Como usuario, quiero recibir una notificación cuando mi rostro sea reconocido en una ubicación específica.	Pendiente	Baja
EP009	ID09	Como usuario, quiero recibir una alerta si se detecta un acceso no autorizado con mi perfil facial.	Pendiente	Alta

Fuente: elaboración propia.

4.3.1.2 Identificación y estimación de tareas

La identificación y estimación de tareas es una etapa crítica en el desarrollo de un software de reconocimiento facial. Esta fase permite desglosar el proyecto en componentes manejables, asignar recursos de manera eficiente y prever posibles obstáculos. A través de este proceso, se busca garantizar una planificación detallada y un control efectivo del proyecto, lo que contribuye a minimizar riesgos y optimizar la utilización de recursos. La identificación de tareas implica determinar y definir todas las actividades específicas que deben realizarse para completar un proyecto. En esta etapa, los equipos de proyecto analizan el alcance del proyecto y dividen el trabajo en unidades discretas, cada una con un objetivo claro y tangible.

Tabla 26. Tarea de Historias de Usuario Login

T01. Ingresar al sistema - login		
ID	Descripción	Horas
1	Crear la base de datos	10
2	Diseñar la vista de inicio del sistema	1
3	Codificar la vista diseñada	3
4	Conectar la vista con la base de datos	1

5	Implementar las validaciones para inicio de sesión	3
6	Testing (pruebas)	1
Tiempo estimado		19

Fuente: elaboración propia.

Tabla 27. Tarea de Historias de Usuario funcionalidades

T02. Visualizar funcionalidades		
ID	Descripción	Horas
1	Diseñar las vistas de funciones de administrador	5
2	Codificar las vistas diseñadas	14
3	Testing (pruebas)	3
Tiempo estimado		22

Fuente: elaboración propia.

Tabla 28. Tarea de Historias de Usuario editar datos

T03. Editar datos personales		
ID	Descripción	Horas
1	Diseñar vista de editar perfil administrador, estudiante, terceros	8
2	Codificar las vistas diseñadas	10
3	Conectar la base de datos que permita visualizar la información actual	3
4	Tomar los datos nuevos de la vista y hacer la actualización de la base de datos.	
5	Testing (pruebas)	3
Tiempo estimado		24

Fuente: elaboración propia.

Tabla 29. Tarea de Historias de Usuario panel administrador

T04. Panel administrador		
ID	Descripción	Horas
1	Diseñar la interfaz de entrada de datos en la aplicación web	10
2	Codificar las vistas diseñadas	2
3	Integrar la funcionalidad con la base de datos	3
4	Realizar las pruebas de ingreso de datos y validación en situaciones reales	2
Tiempo estimado		17

Fuente: elaboración propia.

Tabla 30. Tarea de Historias de Usuario editar datos

T05. Gestión de imágenes		
ID	Descripción	Horas
1	Diseñar la interfaz de usuario para la captura de imágenes faciales	8
2	Implementar la funcionalidad de subir imágenes	8
3	Integrar el almacenamiento seguro de imágenes en la base de datos	3
4	Tomar los datos nuevos de la vista y hacer la actualización de la base de datos.	4
5	Testing (pruebas)	3
Tiempo estimado		26

Fuente: elaboración propia.

Tabla 31. Tarea de Historias de Entrenamiento de la IA

T06. Entrenamiento de la Inteligencia artificial		
ID	Descripción	Horas
1	Entrenar el modelo de reconocimiento facial	8
2	Optimizar el rendimiento del modelo	10
3	Validar la precisión del modelo de reconocimiento facial con datos de prueba	3
4	Testing (pruebas)	3
Tiempo estimado		24

Fuente: elaboración propia.

Tabla 32. Tarea de Historias de generador de informes

T07. Generador de informes		
ID	Descripción	Horas
1	Diseñar una interfaz de generación de informes	8
2	Desarrollar la funcionalidad de consulta de históricos	10
3	Testing (pruebas)	3
Tiempo estimado		21

Fuente: elaboración propia.

Tabla 33. Tarea de Historias de Reunión Stakeholders

T08. Reunión Stakeholders		
ID	Descripción	Horas
1	Programar reuniones con los stakeholders para recopilar requisitos	8

2	Documentar los requisitos en un documento detallado	5
3	Realizar revisiones y aclaraciones con los stakeholders	3
Tiempo estimado		16

Fuente: elaboración propia.

Tabla 34. Tarea de Historias de Monitoreo y estadísticas

T09. Monitoreo de estadísticas y reportes		
ID	Descripción	Horas
1	Diseñar el módulo de estadísticas y reportes de uso y calidad	7
2	Desarrollar la funcionalidad de generación y presentación de reportes.	1
3	Diseñar el módulo de generación de reportes	3
4	Testing (pruebas).	1
Tiempo estimado		12

Fuente: elaboración propia.

4.3.1.3 Sprint Backlog del proyecto

Proporcionar una hoja de ruta clara y detallada para el equipo de desarrollo durante el sprint, permite que el equipo se concentre en un conjunto definido de tareas, evitando distracciones y facilitando una gestión eficiente del tiempo y los recursos. La creación del Sprint Backlog comienza con la reunión de planificación del sprint, donde el equipo de desarrollo, el Product Owner y el Scrum Master colaboran para seleccionar las historias de usuario y las tareas del Product Backlog que se abordarán en el próximo sprint. Una vez creado, el equipo de desarrollo se compromete a completar las tareas listadas durante el sprint. Cada día, el equipo

realiza reuniones diarias de Scrum, donde se documentan los daily stand-ups, que cada miembro del equipo informa sobre su progreso, las tareas que planea completar ese día y cualquier obstáculo que esté enfrentando. Estas reuniones permiten una comunicación constante y ayudan al equipo a mantenerse alineado y enfocado en los objetivos del sprint.

Sprint 1 – Login: proporcionar una funcionalidad completa y funcional de Login que permita a los usuarios autenticarse en el sistema utilizando un nombre de usuario y una contraseña. Para lograr esto, el equipo de desarrollo ha desglosado este objetivo en varias tareas específicas que cubrirán todos los aspectos necesarios del desarrollo, desde el diseño de la interfaz de usuario hasta la implementación de la lógica de autenticación y las medidas de seguridad. El Sprint 1 se enfoca en el diseño, desarrollo e implementación del sistema de Login, abarcando tanto el frontend como el backend, y asegurando que la funcionalidad sea segura y fácil de usar.

Tabla 35. Sprint 1 Login

<i>Sprint N° 1: Login</i>	<i>ID</i>	<i>DESCRIPCION</i>	<i>PRI</i>	<i>EST</i>	<i>ACT</i>	<i>H</i>
	HU1	Ingresar al sistema - login.	10	Alta	6	19
	HU2	Visualizar las funcionalidades del sistema	10	Alta	3	22
	HU3	Pantalla de inicio	8	Alta	5	7
	Actividades y tiempo del sprint				14	48

Fuente: elaboración propia.

Sprint 2 – gestión de imágenes faciales: El objetivo principal del Sprint 2 es proporcionar una funcionalidad completa y funcional de almacenamiento de imágenes faciales que permita al administrador de capturar y almacenar las imágenes de los usuarios (Estudiantes, docentes, administrativos y terceros) en el sistema de manera segura. Para lograr esto, el equipo de desarrollo desglosa este objetivo en varias tareas específicas que cubrirán todos los aspectos necesarios del

desarrollo, desde el diseño de la interfaz de usuario hasta la implementación de la lógica de almacenamiento y las medidas de seguridad.

Tabla 36. Sprint 2 – gestión de imágenes faciales

	<i>ID</i>	<i>DESCRIPCION</i>	<i>PRI</i>	<i>EST</i>	<i>ACT</i>	<i>H</i>
<i>Sprint N° 2: gestión de imágenes faciales</i>	HU4	<i>Diseño de la Interfaz de Usuario (UI) para la Captura de Imágenes Faciales</i>	10	Alta	6	19
	HU5	Implementación de la Funcionalidad de Captura de Imágenes Faciales	10	Alta	3	22
	HU6	Desarrollo del Backend para el Almacenamiento de Imágenes	8	Alta	5	7
	HU7	Optimización del Procesamiento de Imágenes	8	Alta	4	10
Actividades y tiempo del sprint					18	58

Fuente: elaboración propia.

Sprint 3 – Entrenamiento del algoritmo de reconocimiento facial (IA): El objetivo principal del Sprint 3 es desarrollar y entrenar un modelo de reconocimiento facial que pueda identificar con precisión las imágenes faciales capturadas y almacenadas. Para lograr este objetivo, el equipo de desarrollo ha desglosado las tareas en pasos específicos que abarcan desde la preparación del conjunto de datos hasta la evaluación del modelo entrenado. Estas tareas asegurarán que el algoritmo de reconocimiento facial sea robusto, preciso y eficiente.

Tabla 37. Sprint 3 – Entrenamiento del algoritmo de reconocimiento

<i>Spr int</i>	<i>ID</i>	<i>DESCRIPCION</i>	<i>PRI</i>	<i>EST</i>	<i>ACT</i>	<i>H</i>
--------------------	-----------	--------------------	------------	------------	------------	----------

HU8	Recolección y Preparación del Conjunto de Datos	10	Alta	6	15
HU9	Preprocesamiento de Datos	10	Alta	3	20
HU10	Entrenamiento del Modelo	8	Alta	3	12
Actividades y tiempo del sprint				12	47

Fuente: elaboración propia.

Sprint 4 – Gestionar alertas: El principal objetivo de este sprint es implementar un sistema de generación y gestión de alertas, con características diseñadas para identificar automáticamente emergencias o situaciones que excedan los límites del proceso de recubrimiento. Las "Alertas de administrador" de Sprint mejoran el control en la seguridad en el ingreso a la universidad al permitir que capturar usuarios que no se encuentren registrados y alertar oportunamente, mejorando así la capacidad de respuesta y la toma de decisiones en tiempo real.

Tabla 38. Sprint 4 - Gestionar alertas

	ID	DESCRIPCION	PRI	EST	ACT	T
Sprint N° 4: Alertas	HU20	Como administrador quiero enviar alertas al operador de campo y al ingeniero de calidad así mismo elegir el tipo de alerta a enviar.	6	Baja	5	19
	HU21	Como administrador quiero que las alertas sean de tipo importante, e informativa.	6	Baja	4	4
	Actividades y tiempo del sprint				9	23

Fuente: elaboración propia.

4.3.2 Planificación de los sprint

Tabla 39. Revisión de cumplimiento de sprint 1

Sprint 1 - Login

Fecha inicio	2-enero-2024	Revisión de avances
Fecha terminación	31-enero-2024	
Tareas para desarrollar	• Ingresar al sistema - login. • Visualizar las funcionalidades del sistema • Pantalla de inicio • Pruebas de funcionamiento	16-01-2024 23-01-2024 30-01-2024

Fuente: elaboración propia.

Tabla 40. Revisión de cumplimiento de sprint 2

<i>Sprint 2 - Gestión de imágenes faciales</i>		
Fecha inicio	1-abril-2024	Revisión de avances
Fecha terminación	30-abril-2024	
Tareas para desarrollar	• Diseño de la Interfaz de Usuario (UI) para la Captura de Imágenes Faciales • Implementación de la Funcionalidad de Captura de Imágenes Faciales • Desarrollo del Backend para el Almacenamiento de Imágenes • Optimización del Procesamiento de Imágenes	08-04-2024 16-04-2024 30-04-2024

Fuente: elaboración propia.

Tabla 41. Revisión de cumplimiento sprint 3

<i>Sprint 3 – Entrenamiento del algoritmo de reconocimiento</i>		
Fecha inicio	1-julio-2024	Revisión de avances
Fecha terminación	30-julio-2024	
Tareas para desarrollar	• Recolección y Preparación del Conjunto de Datos • Preprocesamiento de Datos • Entrenamiento del Modelo	16-07-2024 23-07-2024 30-07-2024

Fuente: elaboración propia.

4.3.3 Implementación de los sprint

El objetivo principal de esta fase es convertir los diseños y especificaciones en un sistema de reconocimiento facial completamente funcional. Esto incluye el desarrollo de todos los componentes del software, la integración de estos componentes en un sistema cohesivo, y la validación de su funcionamiento a través de la puesta en marcha, esto asegura que todas las funcionalidades planificadas se realicen de manera efectiva y que el sistema esté listo para su uso por el usuario final. La implementación no solo implica la codificación del software, sino también la integración de diversos componentes, pruebas exhaustivas y la preparación para el despliegue.

4.3.3.1 Crear entregables

Los componentes clave del proyecto requieren una ejecución cuidadosa para obtener resultados, lo que demuestra que el sistema se implemente, y la interfaz de usuario diseñada y la funcionalidad desarrollada se combinen en una versión completamente funcional de la aplicación. Del mismo modo, se proporciona una documentación completa que describa la funcionalidad, la arquitectura y las instrucciones de uso de la aplicación para garantizar un funcionamiento y cumplimiento sin problemas, y estos productos deben someterse a pruebas de software.

Tabla 42. Tareas iniciales

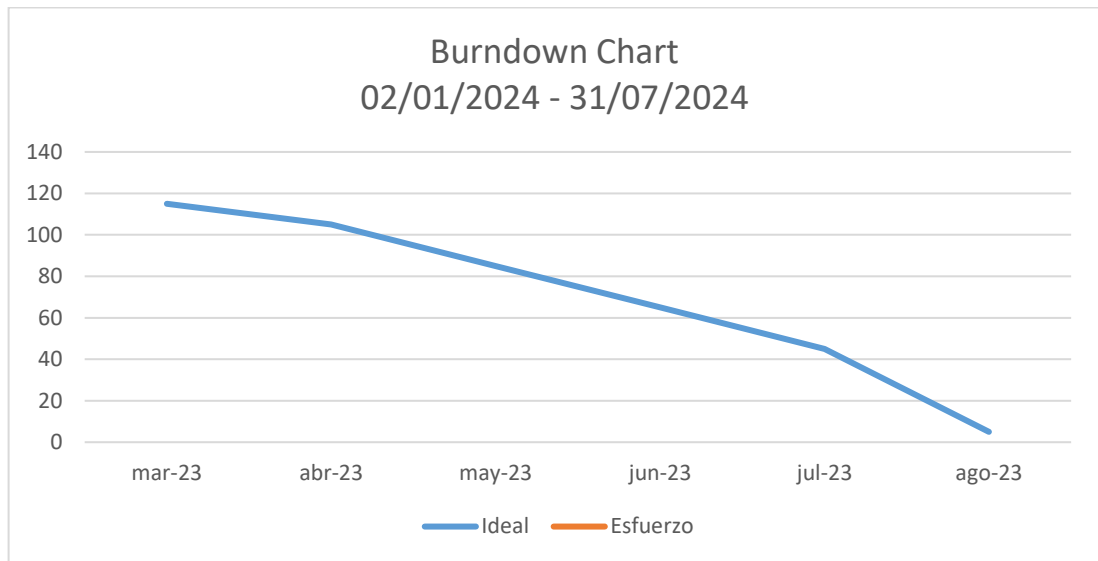
Tablero de tareas			
Inicio: 02/01/2024 Fin: 31/07/2024			
Historias de usuario	Por Hacer	Ejecutando	Hecho
Sprint 1	Ingresar al sistema - login.	X	
	Visualizar las funcionalidades del sistema		
	Pantalla de inicio		
	Pruebas de		

	funcionamiento	
Sprint 2	Diseño de la Interfaz de Usuario (UI) para la Captura de Imágenes Faciales	X
	Implementación de la Funcionalidad de Captura de Imágenes Faciales	
	Desarrollo del Backend para el Almacenamiento de Imágenes	
	Optimización del Procesamiento de Imágenes	
Sprint 3	Recolección y Preparación del Conjunto de Datos	X
	Preprocesamiento de Datos	
	Entrenamiento del Modelo	

Fuente: elaboración propia.

Después de completar el panel de tareas inicial, el equipo planteó un gráfico de avance con la cantidad de tareas establecidas en función del tiempo estimado de finalización de cada tarea, donde el tiempo en el eje X y las tareas en el eje Y. El sistema requiere 10 tareas, que se llevarán a cabo durante un período de seis meses a partir del 2 de enero de 2024. A partir de estos datos, se puede observar una línea diagonal en el gráfico que representa el comportamiento ideal para lograr el objetivo propuesto.

Figura 18. Burndown inicial



Fuente: elaboración propia.

Sprint 1

La primera etapa del proyecto de desarrollo del software tendrá una duración de tres semanas y se centrará en implementar la funcionalidad de inicio de sesión. Este componente es crucial porque proporciona el punto de entrada al sistema, asegurando que solo los usuarios autorizados puedan acceder y utilizar la funcionalidad del software. Durante este sprint, se desarrollarán todos los aspectos necesarios del sistema de inicio de sesión, incluido el diseño de la interfaz de usuario, la lógica de autenticación en el backend, la integración de medidas de seguridad y las pruebas necesarias para garantizar que funcione correctamente.

Semana 1

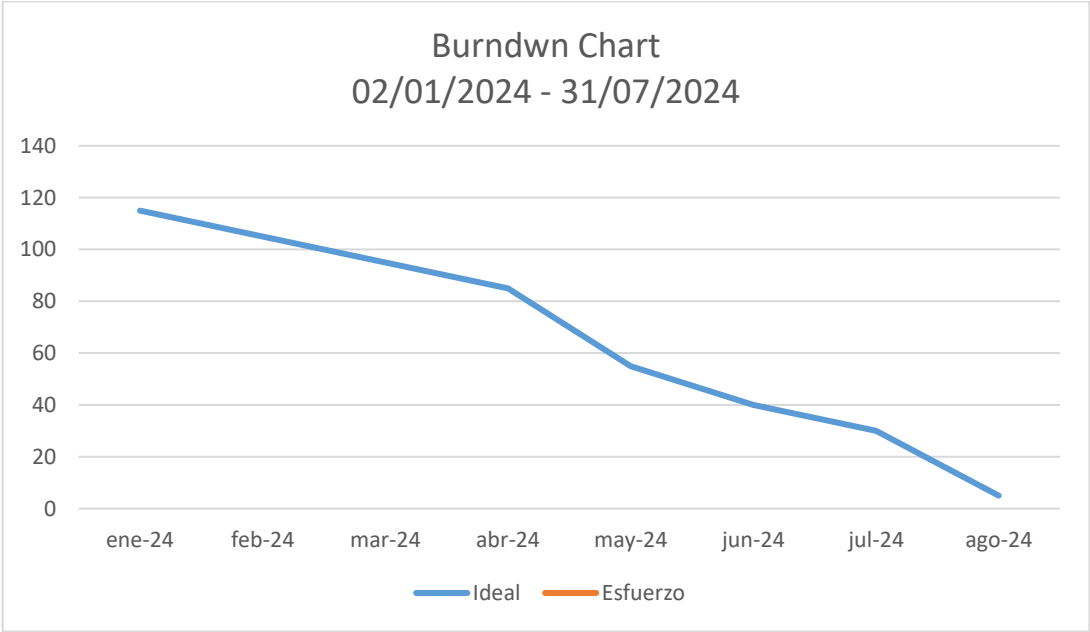
La primera semana del sprint estará dedicada al diseño y planificación. El diseño de la interfaz de usuario (UI) para la pantalla de Login será uno de los primeros pasos a abordar. La interfaz debe ser intuitiva y accesible, con campos claros para el nombre de usuario y la contraseña, así como botones para iniciar sesión y recuperar la contraseña.

Tabla 43. tareas semana 1

Tareas			
Inicio: 02/01/2024 Fin: 31/07/2024			
Historias de usuario	Por Hacer	Ejecutando	Hecho
Sprint 1	Ingresar al sistema - login.	X	
	Visualizar las funcionalidades del sistema	X	
	Pantalla de inicio	X	
	Pruebas de funcionamiento	X	

Fuente: elaboración propia.

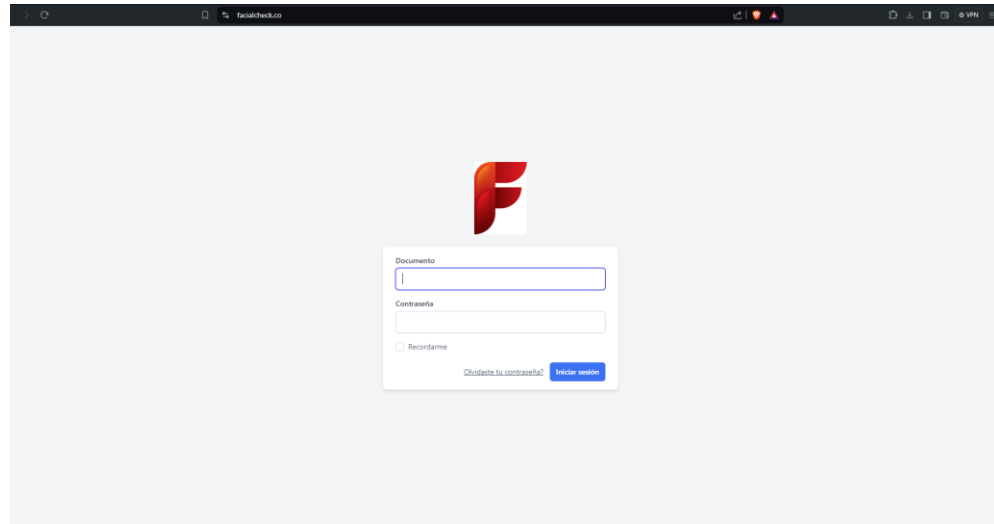
Figura 19. Burndown semana 1



Fuente: elaboración propia.

Historia ejecutada: El formulario de autenticación incluye campos de entrada claros y etiquetados, típicamente para el nombre de usuario (o identificación) y la contraseña. Estos campos deben estar alineados verticalmente para facilitar la lectura y la interacción.

Figura 20. Diseño de vista del Login



Fuente: elaboración propia.

Un botón de inicio de sesión prominente debe estar presente debajo de los campos de entrada. Este botón debe ser fácilmente identificable, utilizando un color contrastante con el fondo para destacar su importancia. Una casilla de verificación para "Recordarme" puede ser incluida para ofrecer comodidad a los usuarios frecuentes.

Semana 2

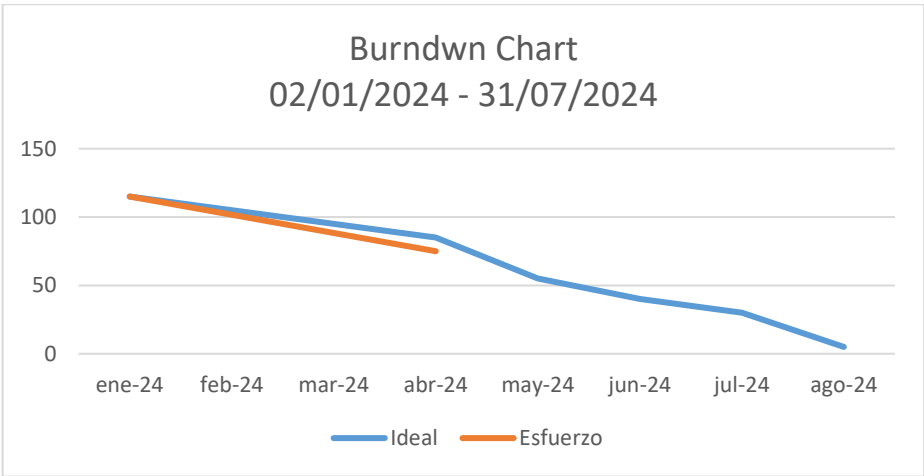
En la segunda semana, el enfoque se desplazará hacia el desarrollo del frontend y backend para la funcionalidad de Login. El diseño de navegación para el administrador incluye un menú principal con acceso a todas las secciones importantes del sistema, permitiendo la gestión completa de perfiles, imágenes, entrenamiento de la inteligencia artificial (IA), usuarios, y roles.

Tabla 44. tareas semana 2

Tareas			
Inicio: 02/01/2024 Fin: 31/07/2024			
Historias de usuario	Por Hacer	Ejecutando	Hecho
Sprint 1	Ingresar al sistema - login.		X
	Visualizar las funcionalidades del sistema	X	
	Pantalla de inicio	X	
	Pruebas de funcionamiento	X	

Fuente: elaboración propia.

Figura 21. Burndown semana 2



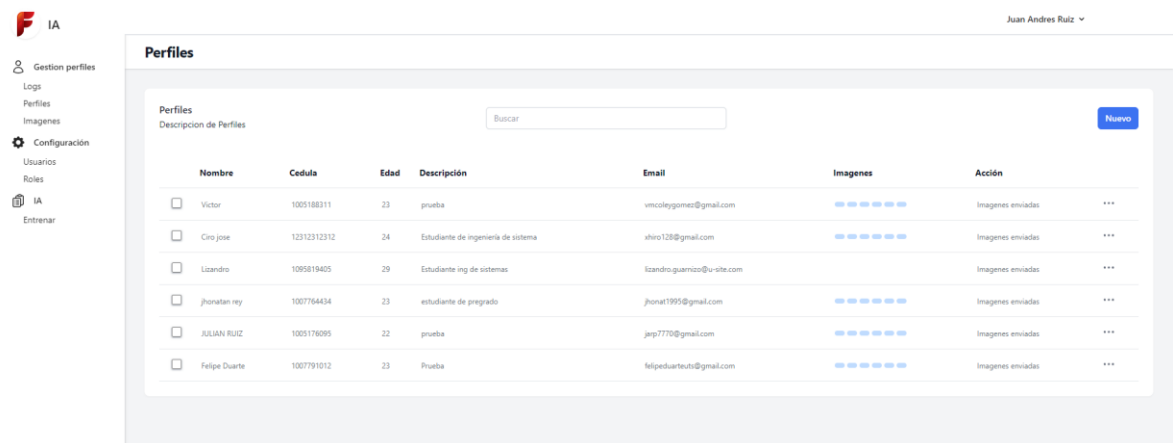
Fuente: elaboración propia.

Historia ejecutada: Este componente permite a los administradores gestionar usuarios, monitorear actividades y supervisar la operación general del sistema. La

implementación del panel de control no solo mejora la funcionalidad y la seguridad del software, sino que también proporciona herramientas esenciales para la administración y el mantenimiento del sistema.

El primer paso fue el diseño de la interfaz de usuario para el panel de control. El equipo de diseño creó maquetas detalladas que incluían elementos clave como paneles de navegación, tablas de datos y gráficos de análisis. La interfaz permite a los administradores acceder fácilmente a las diferentes funcionalidades. El objetivo era garantizar una experiencia de usuario fluida y coherente en todas las partes del panel de control.

Figura 22. Panel de control Administrador



Fuente: elaboración propia.

Funcionalidad:

- **Menú Principal:** Este menú permite al administrador acceder rápidamente a cada sección principal del sistema.
- **Perfiles:** Gestión completa de perfiles de usuarios, desde la creación hasta la eliminación.
- **Imágenes:** Carga y asociación de imágenes a los perfiles, así como la gestión de las mismas.

- **Entrenar:** Proceso de preparación y entrenamiento de la IA con las imágenes cargadas.
- **Usuarios:** Gestión de usuarios del sistema, incluyendo la asignación de roles.
- **Roles:** Definición y asignación de roles y permisos para los usuarios del sistema.

Este diseño de navegación asegura que el administrador pueda realizar todas las tareas necesarias de manera eficiente y organizada.

Semana 3

En la tercera semana del sprint, el equipo de desarrollo se enfocará en la funcionalidad de edición de datos personales dentro del sistema de reconocimiento facial. Este componente es crucial para asegurar que los usuarios puedan actualizar su información personal de manera sencilla y segura. La capacidad de editar datos personales mejora la experiencia del usuario y mantiene la precisión de los registros del sistema.

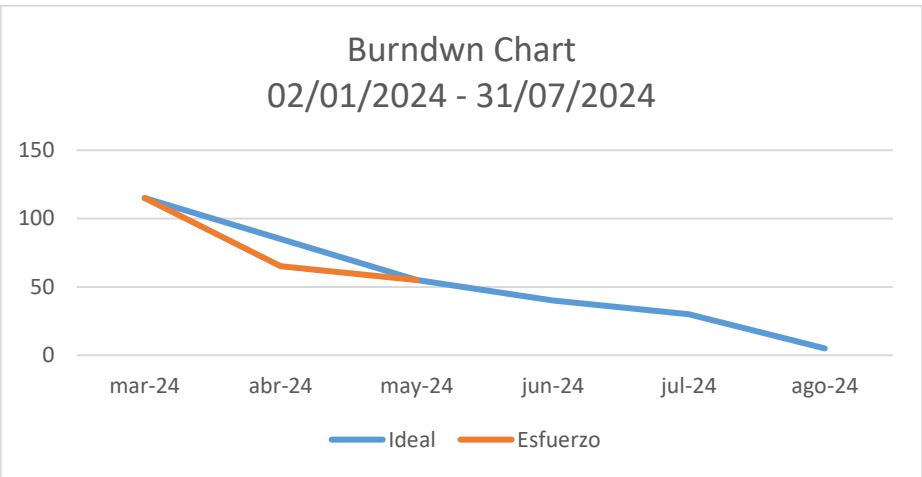
Tabla 45. tareas semana 3

Tareas			
Inicio: 02/01/2024 Fin: 31/07/2024			
Historias de usuario	Por Hacer	Ejecutando	Hecho
Sprint 1	Ingresar al sistema - login.		X
	Visualizar las funcionalidades del sistema		X
	Pantalla de inicio	X	

Pruebas de funcionamiento	X
---------------------------	---

Fuente: elaboración propia.

Figura 23. Burndown semana 3



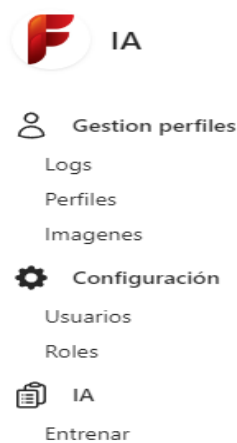
Fuente: elaboración propia.

Historia ejecutada – Diseño de vista pantalla de inicio

El diseño de la interfaz de usuario (UI) está pensado para ser intuitivo y eficiente, permitiendo al administrador realizar todas las tareas necesarias de manera fluida. Cada sección del sistema cuenta con su propia pantalla, organizada en formularios y listas para la gestión de perfiles, imágenes, entrenamiento de IA, usuarios y roles.

Funcionalidad: La pantalla de inicio presenta un panel de navegación lateral con acceso a todas las secciones principales del sistema.

Figura 24. Vista panel de inicio



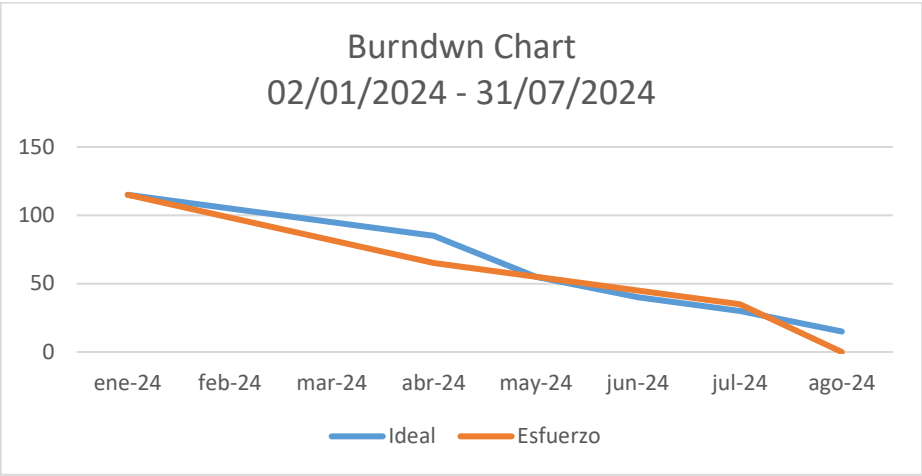
Fuente: elaboración propia.

Tabla 46. tareas semana 3

Tareas			
Inicio: 02/01/2024 Fin: 31/07/2024			
Historias de usuario	Por Hacer	Ejecutando	Hecho
Sprint 1	Ingresar al sistema - login.		X
	Visualizar las funcionalidades del sistema		X
	Pantalla de inicio		X
	Pruebas de funcionamiento	X	

Fuente: elaboración propia.

Figura 25. Burndown semana 3



Fuente: elaboración propia.

Diseño Pantalla de Crear Perfiles

Esta vista permite a los administradores y a los propios usuarios gestionar y actualizar sus datos personales, configuraciones de cuenta, y preferencias. Se utiliza un diseño limpio y minimalista para facilitar la navegación y la comprensión de las opciones disponibles. Diseño responsivo que se adapta a diferentes tamaños de pantalla y dispositivos, asegurando una experiencia óptima en móviles y tabletas.

Figura 26. Diseño vista Crear perfiles

Crear Perfiles

Nombre

nombre

0 / 100

Cedula

cedula

0 / 100

Edad

edad

0 / 100

Email

email

0 / 100

Descripcion

descripcion

0 / 100

Cerrar

Guardar

Fuente: elaboración propia.

Funcionalidad: Esta pantalla permite al administrador crear, editar, enviar las imágenes asociadas al usuario para lograr entrenar la IA. Contiene un formulario para ingresar los datos del perfil y una lista de todos los perfiles existentes.

Subsecciones:

- **Formulario de Creación/Edición:**
 - Campos: Nombre, Cédula, Edad, Descripción, Email.
- **Lista de Perfiles:**
 - Muestra todos los perfiles con opciones para editar, enviar las imágenes asociadas a la IA y eliminar.

Diseño vista Perfiles

El diseño de la vista de perfiles es una parte integral de la aplicación ya que permite al usuario visualizar y gestionar su información personal y profesional. Esta vista es visualmente atractiva y funcional, proporcionando una experiencia positiva y eficiente.

Figura 27. Diseño vista perfiles

Perfiles

Perfiles

Description de Perfiles

Buscar

Nuevo

Nombre	Cedula	Edad	Descripción	Email	Imágenes	Acción
☐ Victor	1005188311	23	prueba	vmcolegomez@gmail.com	    	Imágenes enviadas ...
☐ Ciro jose	12312312312	24	Estudiante de ingeniería de sistema	xhiro128@gmail.com	    	Imágenes enviadas ...
☐ Lizandro	1095819405	29	Estudiante ing de sistemas	lizandro.guarizto@u-site.com	    	Imágenes enviadas ...
☐ jhonatan rey	1007764434	23	estudiante de pregrado	jhonat1995@gmail.com	    	Imágenes enviadas ...
☐ JULIAN RUIZ	1005176095	22	prueba	jerp7770@gmail.com	    	Imágenes enviadas ...
☐ Felipe Duarte	1007791012	23	Prueba	felipeduarteuts@gmail.com	    	Imágenes enviadas ...

Fuente: elaboración propia.

Semana 4

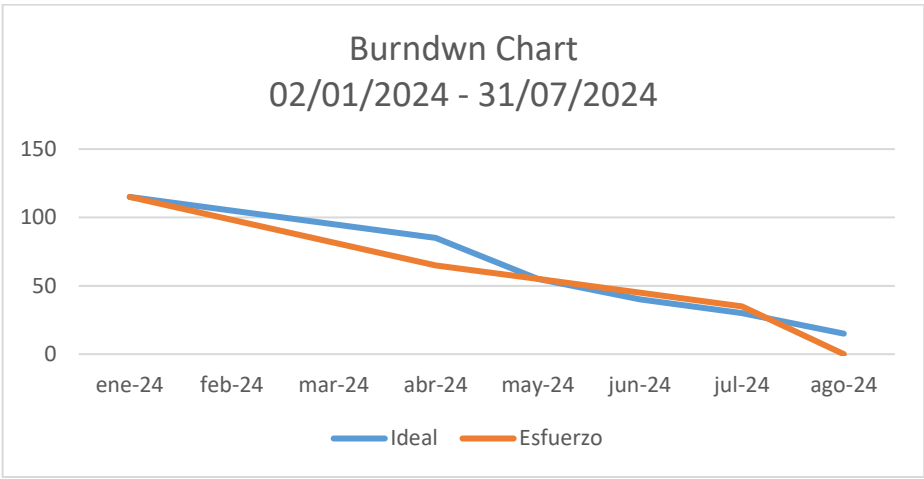
Esta tarea es fundamental ya que los usuarios deben poder registrarse dentro de la aplicación, razón por la cual la tabla indica que la tarea está actualmente en progreso.

Tabla 47. tareas semana 4

Tareas			
Inicio: 02/01/2024 Fin: 31/07/2024			
Historias de usuario	Por Hacer	Ejecutando	Hecho
Sprint 1	Ingresar al sistema - login.		X
	Visualizar las funcionalidades del sistema		X
	Pantalla de inicio		X
	Pruebas de funcionamiento	X	

Fuente: elaboración propia.

Figura 28. Burndown semana 4



Fuente: elaboración propia.

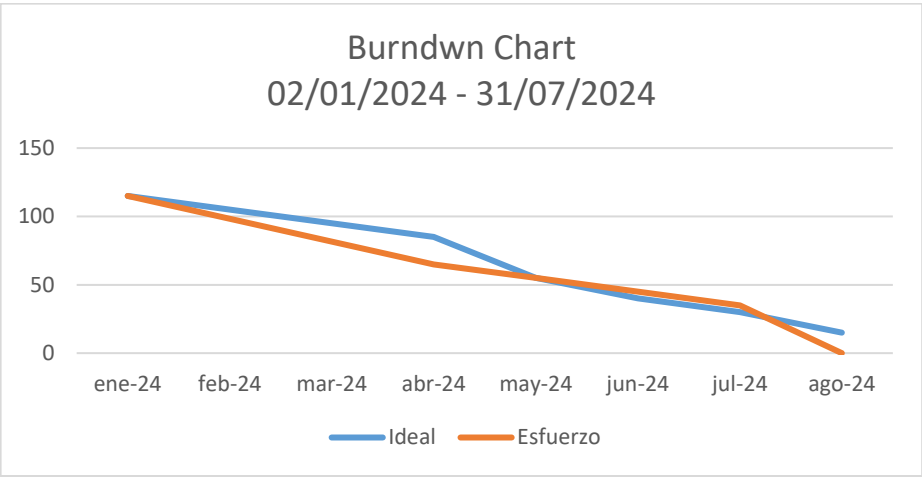
La capacidad de los usuarios de gestionar sus propios datos personales dentro del software proporciona una herramienta útil para mejorar la experiencia del usuario y maximizar el control de calidad. Después de completar este hito, el desarrollo del proyecto continúa a medida que se implementan funciones adicionales.

Tabla 48. tareas finalizadas semana 4

Tareas			
Inicio: 02/01/2024 Fin: 31/07/2024			
Historias de usuario	Por Hacer	Ejecutando	Hecho
Sprint 1	Ingresar al sistema - login.		X
	Visualizar las funcionalidades del sistema		X
	Pantalla de inicio		X
	Pruebas de funcionamiento		X

Fuente: elaboración propia.

Figura 29. Burndown final semana 4



Fuente: elaboración propia.

Historia ejecutada - Diseño vista Gestión de Usuarios

Permite la creación, edición y eliminación de usuarios del sistema, así como la asignación de roles.

Subsecciones:

- **Formulario de Creación/Edición:**
 - Campos: Documento, Email, Primer Nombre, Primer Apellido, contraseña, Rol.
- **Lista de Usuarios:**
 - Muestra todos los usuarios con opciones para editar y eliminar.

Figura 30. Diseño vista crear de usuarios

Crear Usuario

Documento

Documento

0 / 15

Email

Email

0 / 50

Primer Nombre

Primer nombre

0 / 40

Primer Apellido

Primer apellido

0 / 40

Contraseña

Contraseña

0 / 12

Rol

Seleccionar

Cerrar

Guardar

Fuente: elaboración propia.

Figura 31. Diseño vista gestión de usuarios

Usuarios

Administración de Usuarios

Buscar

Nuevo

☐	Documento	Nombre	Correo	Rol	Empresa	Estado	Ultima conexión	
☐	1092362816	Ciro Perrony	cperrony@gmail.com	Administrador	FacialCheck	☒	2024-05-28 07:47:15 PM	...
☐	1007764434	Jhonatan Rey	jhonat1995@gmail.com	Administrador	FacialCheck	☒	2024-05-17 07:59:24 PM	...
☐	1062916077	Aisquel Velasquez	aisquev128@gmail.com	Rol 2	FacialCheck	☒	2024-05-17 08:03:26 PM	...

Fuente: elaboración propia.

Historia ejecutada - Diseño vista Gestión de Roles

Se crean y gestionan los roles del sistema, especificando los permisos asociados a cada rol.

- **Formulario de Creación/Edición:**
 - Campos: Nombre del Rol, Descripción, selector de módulos a los que tendrá acceso ese rol.
- **Lista de Roles:**
 - Muestra todos los roles con opciones para editar y eliminar.

Figura 32. Diseño de vista crear Roles



Crear Rol

Nombre Descripción

Rol Descripción

0 / 40 0 / 100

SubModulos

Buscar Submodulos

Cerrar Guardar

Fuente: elaboración propia.

Figura 33. Diseño vista Gestión de Roles

Roles				
Roles		Descripción de Roles		Buscar
				Nuevo
Nombre	Descripción	SubModulos activos		Creado
☐ Administrador	Rol de administrador	Usuarios Roles Logs Perfiles Imagenes Entrenar		Lizandro ...
☐ Rol 2	Rol 2			...
☐ Rol 1	Rol 1	Usuarios Roles Logs Perfiles Imagenes		/Jonathan ...

Fuente: elaboración propia.

Sprint 2

El equipo crea un tablero de tareas y un gráfico de avance adecuados para comenzar la ejecución del sprint y también decide que el sprint durará cuatro semanas.

Semana 5

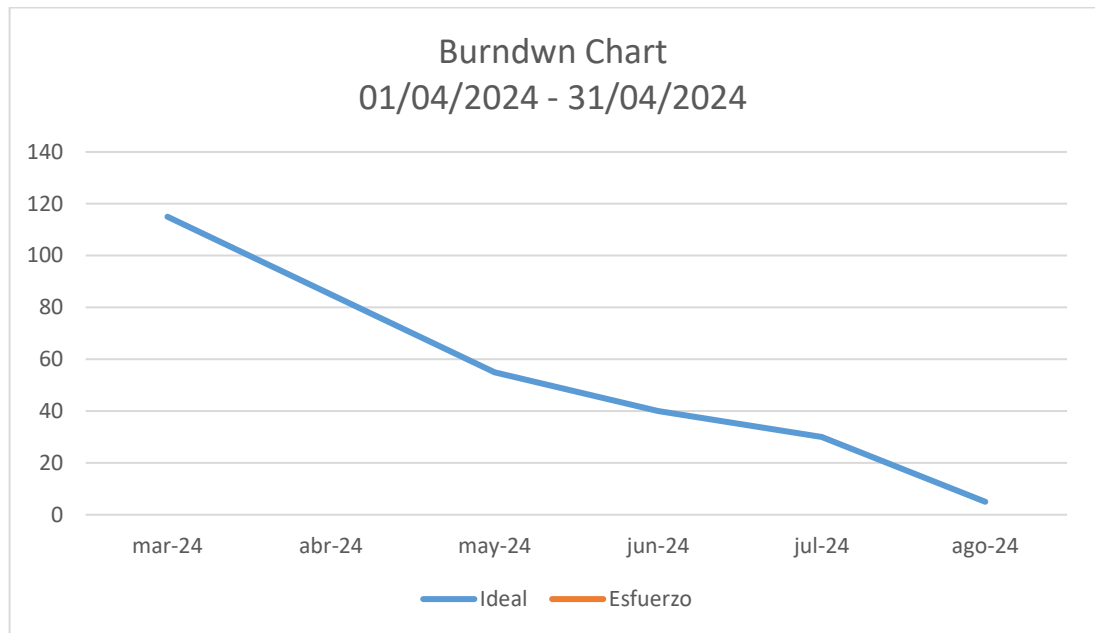
El objetivo principal de esta semana será crear un diseño intuitivo y funcional que facilite el proceso de captura de imágenes faciales, manteniendo la consistencia visual y la usabilidad del sistema.

Tabla 49. tareas semana 5

Tareas			
Inicio: 01/04/2024 Fin: 31/04/2024			
Historias de usuario	Por Hacer	Ejecutando	Hecho
Sprint 2	Diseño de la Interfaz de Usuario (UI) para la Captura de Imágenes Faciales	X	
	Implementación de la Funcionalidad de Captura de Imágenes Faciales	X	
	Desarrollo del Backend para el Almacenamiento de Imágenes	X	
	Optimización del Procesamiento de Imágenes	X	

Fuente: elaboración propia.

Figura 34. Burndown semana 5



Fuente: elaboración propia.

Historia ejecutada - Diseño vista Gestión de Imágenes

En esta pantalla se cargan las imágenes y se asocian a los perfiles creados. Incluye un formulario para la carga de imágenes y una lista de imágenes asociadas a cada perfil.

Subsecciones:

- **Formulario de Carga:**
 - Campo para seleccionar perfil y cargar imágenes desde el dispositivo.
- **Lista de Imágenes:**
 - Muestra todas las imágenes con opciones para ver y eliminar.

Figura 35. Diseño de vista subir imágenes

Crear Imagenes

Perfil

Seleccionar

Imagen 

Cerrar

Guardar

Fuente: elaboración propia.

Semana 6

El equipo actualizó el tablero de tareas al comienzo de la sexta semana y pudo visualizar las historias de usuarios que necesitaban mejorar esa semana.

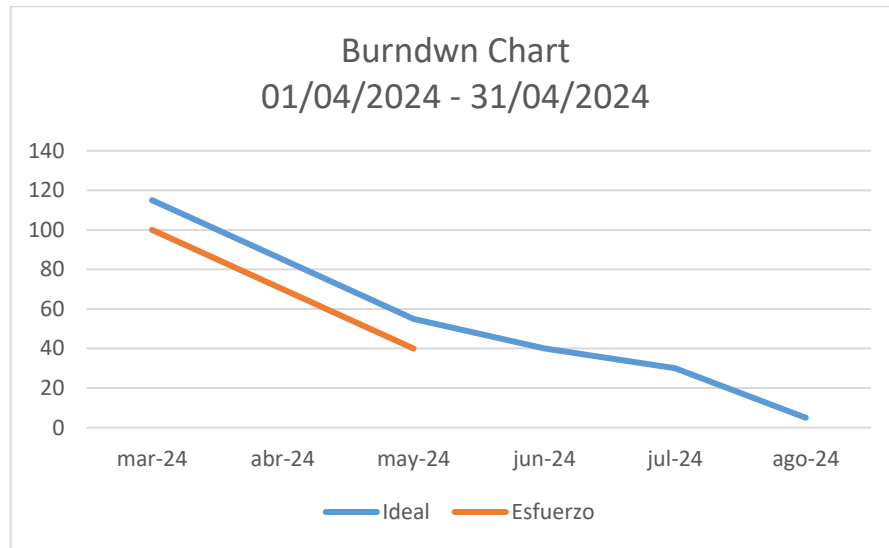
Tabla 50. tareas semana 6

Tareas			
Inicio: 01/04/2024 Fin: 31/04/2024			
Historias de usuario	Por Hacer	Ejecutando	Hecho
Sprint 2	Diseño de la Interfaz de Usuario (UI) para la Captura de Imágenes Faciales		X
	Implementación de la Funcionalidad de Captura de Imágenes Faciales	X	
	Desarrollo del Backend para el Almacenamiento de Imágenes	X	
	Optimización del Procesamiento de Imágenes	X	

Fuente: elaboración propia.

De esta manera, se actualizo el Burndwn Chart, donde se visualiza el trabajo realizado en las historias de usuario completadas la semana pasada, dando cumplimiento a sus objetivos según el tiempo y el esfuerzo asignados. Ver la línea debajo de la línea ideal es muy productivo para el equipo.

Figura 36. Burndown semana 6



Fuente: elaboración propia.

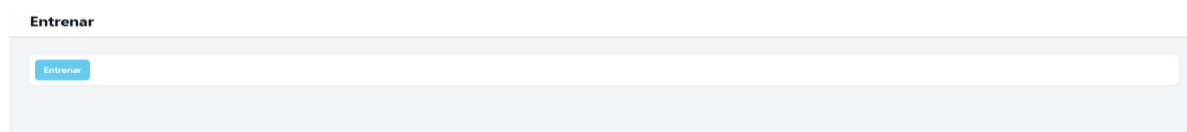
Historia ejecutada - Diseño vista Entrenamiento de la IA

Aquí se realiza el entrenamiento de la IA con las imágenes cargadas. La interfaz permite iniciar el proceso de entrenamiento.

Subsecciones:

- **Ejecución del Entrenamiento:**
 - Botón para iniciar el entrenamiento

Figura 37. Diseño vista Entrenar IA



Fuente: elaboración propia.

Semana 7

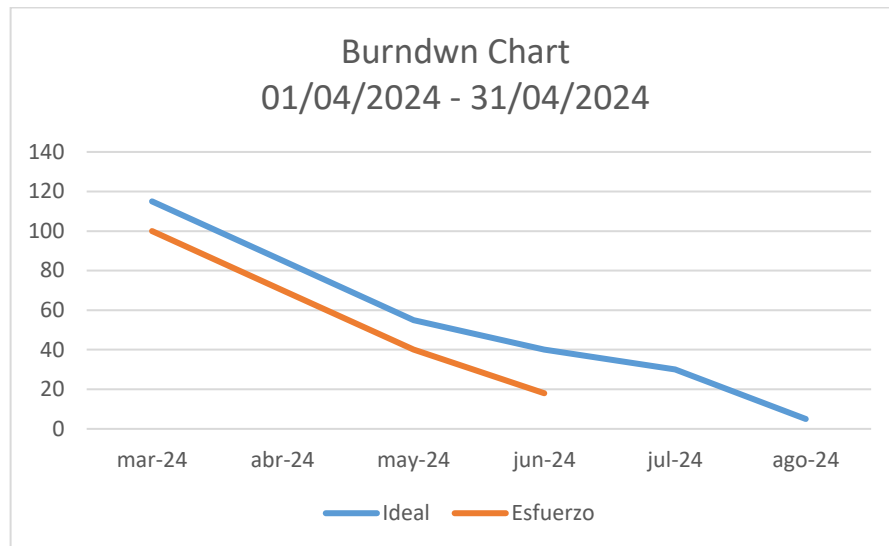
Tabla 51. tareas semana 7

Tareas			
Inicio: 01/04/2024 Fin: 31/04/2024			
Historias de usuario	Por Hacer	Ejecutando	Hecho
Sprint 2	Diseño de la Interfaz de Usuario (UI) para la Captura de Imágenes Faciales		X
	Implementación de la Funcionalidad de Captura de Imágenes Faciales		X
	Desarrollo del Backend para el Almacenamiento de Imágenes	X	
	Optimización del Procesamiento de Imágenes	X	

Fuente: elaboración propia.

Además, el equipo actualizó el borrador del gráfico y pudo ver un buen progreso hacia los objetivos de tiempo.

Figura 38. Burndown semana 7



Fuente: elaboración propia.

Historia ejecutada – Desarrollo del backend para el almacenamiento de imágenes

El primer paso en este proceso fue la configuración del servidor backend. Utilizando tecnologías como Node.js y Express, el equipo creó una API que manejara las solicitudes de almacenamiento de imágenes. Esta API fue diseñada para recibir imágenes faciales enviadas desde el frontend, procesarlas y almacenarlas de manera segura en una base de datos adecuada. Se optó por utilizar una base de datos NoSQL, como MongoDB, por su flexibilidad y capacidad para manejar grandes volúmenes de datos no estructurados.

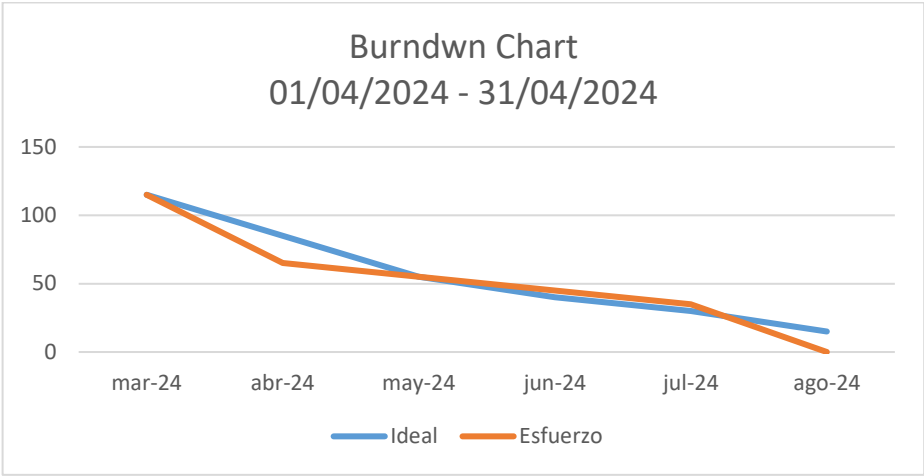
Semana 8

Tabla 52. tareas semana 8

Tareas			
Inicio: 01/04/2024 Fin: 31/04/2024			
Historias de usuario	Por Hacer	Ejecutando	Hecho
Sprint 2	Diseño de la Interfaz de Usuario (UI) para la Captura de Imágenes Faciales		X
	Implementación de la Funcionalidad de Captura de Imágenes Faciales		X
	Desarrollo del Backend para el Almacenamiento de Imágenes		X
	Optimización del Procesamiento de Imágenes	X	

Fuente: elaboración propia.

Figura 39. Burndown semana 8



Fuente: elaboración propia.

Historia ejecutada – Optimización del procesamiento de imágenes

Para asegurar que las imágenes se almacenaran de manera eficiente, el equipo implementó técnicas de compresión y optimización de imágenes. Estas técnicas permitieron reducir el tamaño de las imágenes sin comprometer su calidad, asegurando que el almacenamiento y la recuperación de imágenes fueran rápidos y eficientes.

Figura 40. Diseño vista optimización del procesamiento de imágenes

Imágenes		
Imágenes		
Description de Imágenes		Buscar
		Nuevo
Uri	Perfil	
☐ storage/MIEndb.png	JULIAN RUIZ	***
☐ storage/17ROZP.png	JULIAN RUIZ	***
☐ storage/BhdXU.png	JULIAN RUIZ	***
☐ storage/e1e5R.png	JULIAN RUIZ	***
☐ storage/FYJC3.png	JULIAN RUIZ	***
☐ storage/WCCAgT.png	JULIAN RUIZ	***
☐ storage/29Nk8.png	Felipe Duarte	***
☐ storage/05RETo.png	Felipe Duarte	***
☐ storage/hKkS.png	Felipe Duarte	***
☐ storage/nAQOu.png	Felipe Duarte	***

Fuente: elaboración propia.

Sprint 3

El equipo establece un período de tres semanas para desarrollar el sprint y actualiza los gráficos semanales y el plan del tablero de tareas para que sirvan como guía y visualización gráfica para desarrollar las historias de usuario que componen el sprint.

Semana 9

El equipo crea una tabla de panel de tareas y muestra el estado actual de las historias de usuario, lo que muestra que la primera historia de usuario se está ejecutando en la novena semana.

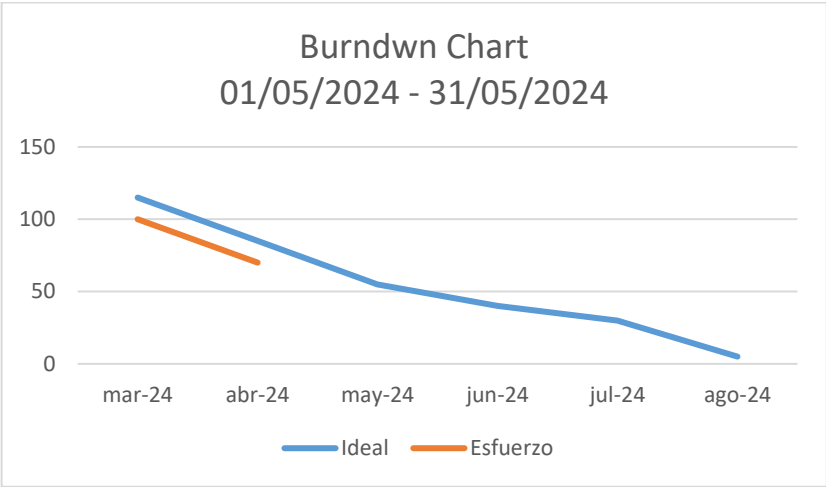
Tabla 53. Tareas semana 9

Tareas			
Inicio: 01/05/2024 Fin: 31/05/2024			
Historias de usuario	Por Hacer	Ejecutando	Hecho
Sprint 3	Recolección y preparación del conjunto de datos	X	
	Preprocesamiento de datos	X	
	Entrenamiento del modelo	X	

Fuente: elaboración propia.

Por otra parte, el equipo actualizó el Burndwn Chart y pudo ver un buen progreso hacia los objetivos de tiempo.

Figura 41. Burndown semana 9

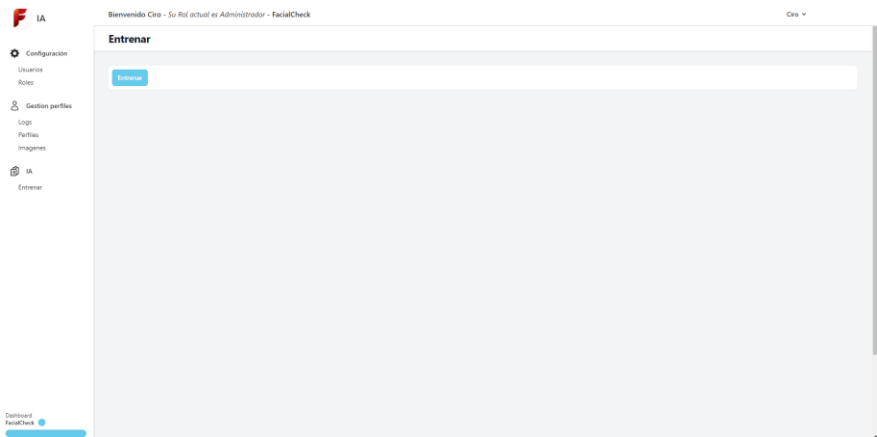


Fuente: elaboración propia.

Historia ejecutada - Recolección y preparación del conjunto de datos

El equipo también trabajó en la creación de endpoints adicionales que permitieran a los usuarios gestionar sus imágenes almacenadas. Esto incluyó la implementación de funcionalidades para eliminar imágenes no deseadas y actualizar imágenes existentes. Cada uno de estos endpoints fue diseñado con un enfoque en la usabilidad y la seguridad, asegurando que los usuarios pudieran gestionar sus imágenes de manera sencilla y protegida.

Figura 42. Recolección y preparación del conjunto de datos



Fuente: elaboración propia.

Semana 10

Para garantizar que las imágenes se almacenen de manera eficiente, el equipo implementó técnicas de optimización y compresión de imágenes. Estas técnicas reducen el tamaño de las imágenes sin comprometer la calidad, lo que garantiza que el almacenamiento y la recuperación de imágenes sean rápidos y eficientes. Este componente es fundamental para asegurar el almacenamiento y protección eficiente de las imágenes, facilitando así su posterior procesamiento y análisis por parte de sistemas de reconocimiento facial.

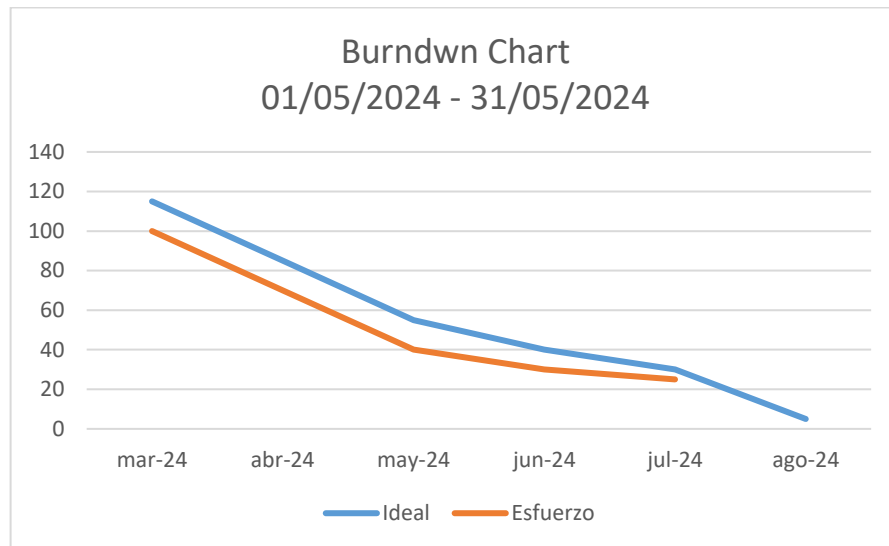
Tabla 54. Tareas semana 10

Tareas			
Inicio: 01/05/2024 Fin: 31/05/2024			
Historias de usuario	Por Hacer	Ejecutando	Hecho
Sprint 3	Recolección y preparación del conjunto de datos		X
	Preprocesamiento de datos	X	
	Entrenamiento del modelo	X	

Fuente: elaboración propia.

Por otra parte, el equipo actualizó el Burndwn Chart y pudo ver un buen progreso hacia los objetivos de tiempo.

Figura 43. Burndown semana 10



Fuente: elaboración propia.

Historia ejecutada – Preprocesamiento de datos

El preprocesamiento de datos implica una serie de técnicas y procedimientos destinados a preparar las imágenes para su uso efectivo por el algoritmo de reconocimiento facial, garantizando que sean consistentes y de alta calidad. El equipo de desarrollo comenzó esta etapa recolectando y organizando el conjunto de datos de imágenes faciales. Estas imágenes provenían de diversas fuentes y presentaban variaciones en términos de iluminación, ángulos y resoluciones. Para normalizar estos datos, se implementaron varias técnicas de preprocesamiento.

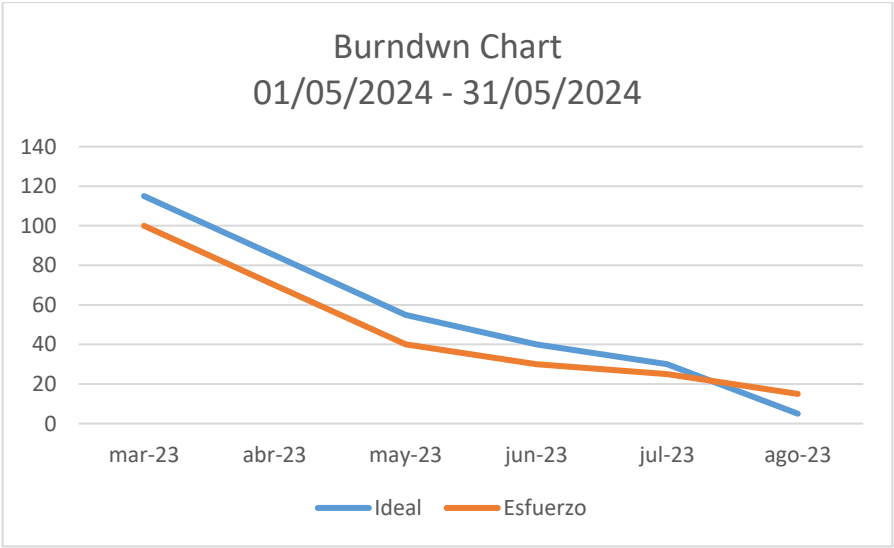
Semana 11

Tabla 55. Tareas semana 11

Tareas			
Inicio: 01/05/2024 Fin: 31/05/2024			
Historias de usuario	Por Hacer	Ejecutando	Hecho
Sprint 3	Recolección y preparación del conjunto de datos		X
	Preprocesamiento de datos		X
	Entrenamiento del modelo		X

Fuente: elaboración propia.

Figura 44. Burndown semana 11



Fuente: elaboración propia.

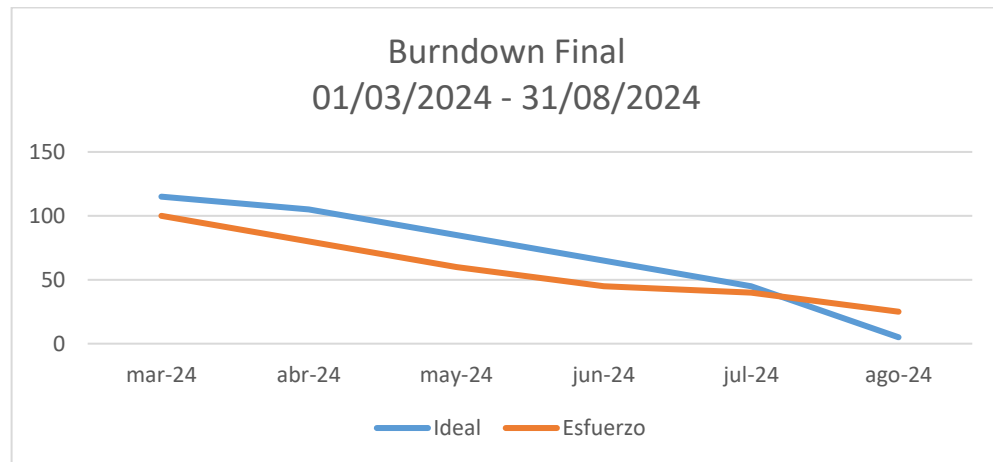
Al finalizar los sprint, el equipo entrega el sistema agregándolos a un diagrama de tablero de tareas común, donde cada historia de usuario para cada sprint se puede mostrar completa, como lo muestra un esquema que muestra la cantidad de trabajo completado.

Tabla 56. Taskboard final

Tablero de tareas			
Inicio: 02/01/2024 Fin: 31/07/2024			
Historias de usuario	Por Hacer	Ejecutando	Hecho
Sprint 1	Ingresar al sistema - login.		
	Visualizar las funcionalidades del sistema		X
	Pantalla de inicio		
	Pruebas de funcionamiento		
Sprint 2	Diseño de la Interfaz de Usuario (UI) para la Captura de Imágenes Faciales		
	Implementación de la Funcionalidad de Captura de Imágenes Faciales		X
	Desarrollo del Backend para el Almacenamiento de Imágenes		
	Optimización del Procesamiento de Imágenes		
Sprint 3	Recolección y Preparación del Conjunto de Datos		
	Preprocesamiento de Datos		X
	Entrenamiento del Modelo		

Fuente: elaboración propia.

Figura 45. Burndown final



Fuente: elaboración propia.

4.3.4 Fase de revisión y retrospectiva

Durante la revisión del sprint, el equipo se reúne para mostrar y demostrar la funcionalidad completa. Esta reunión incluye a todos los miembros del equipo de desarrollo, propietario del producto y otras partes interesadas relevantes. El equipo demostró características implementadas como el sistema de inicio de sesión, almacenamiento de imágenes faciales y preprocesamiento de datos, destacando cómo cada característica cumplió con los criterios de aceptación definidos al comienzo del sprint. Las partes interesadas tienen la oportunidad de probar nuevas funciones, brindar comentarios inmediatos y hacer preguntas. Este proceso ayuda a garantizar que el producto desarrollado cumpla con las expectativas y necesidades del usuario final.

La fase de revisión y retrospectiva es fundamental para el desarrollo del proyecto, ya que proporciona un ciclo continuo de retroalimentación y mejora. Al evaluar el trabajo completado, recibir comentarios de las partes interesadas y reflexionar sobre los procesos internos, los equipos pueden adaptar su enfoque y mejorar el desempeño. Esta etapa no sólo mejora la calidad del producto final, sino que

también mejora la cohesión y eficiencia del equipo, asegurando un desarrollo ágil y exitoso de futuros sprint.

4.3.5 Lanzamiento del software

La fase de lanzamiento es la preparación del entorno de producción. El equipo de desarrollo trabaja en la configuración del servidor y base de datos el cual es necesaria para respaldar el sistema de reconocimiento facial. Esto garantiza que la infraestructura sea escalable y pueda manejar el tráfico de usuarios esperado. Por lo tanto, se implementan medidas de seguridad para proteger los datos confidenciales, incluido el cifrado, la autenticación multifactor y el monitoreo continuo para detectar amenazas potenciales.

Con el entorno de producción configurado y validado, el equipo procedió al despliegue del sistema en fases. Inicialmente, se lanzó una versión beta a un grupo selecto de usuarios para recopilar retroalimentación y realizar ajustes finales basados en sus experiencias. Esta etapa permitió identificar y corregir cualquier problema antes del despliegue completo.

Simultáneamente, se llevaron a cabo sesiones de capacitación para administradores y usuarios finales, proporcionando manuales de usuario y tutoriales en video para facilitar el uso del sistema. Esta capacitación fue esencial para asegurar que los usuarios comprendieran cómo utilizar todas las funcionalidades del sistema de manera efectiva.

4.4 Realizar mediante una prueba piloto en la universidad popular del cesar seccional Aguachica la funcionalidad y aceptación del software de reconocimiento facial.

El presente apartado tiene como objetivo exponer los resultados obtenidos tras la implementación de una prueba piloto en la Universidad Popular del Cesar, seccional Aguachica, orientada a evaluar la funcionalidad y aceptación del software de reconocimiento facial desarrollado. Esta prueba se llevó a cabo con el propósito de

determinar el rendimiento técnico del software en un entorno real y medir la respuesta de los usuarios en cuanto a su facilidad de uso, confiabilidad y eficacia. A través de esta fase experimental, se recolectaron datos clave que permiten no solo validar el correcto funcionamiento del sistema, sino también comprender el nivel de aceptación por parte de la comunidad universitaria, lo cual es fundamental para su posible implementación a mayor escala.

Figura 46. Reporte de pruebas 1

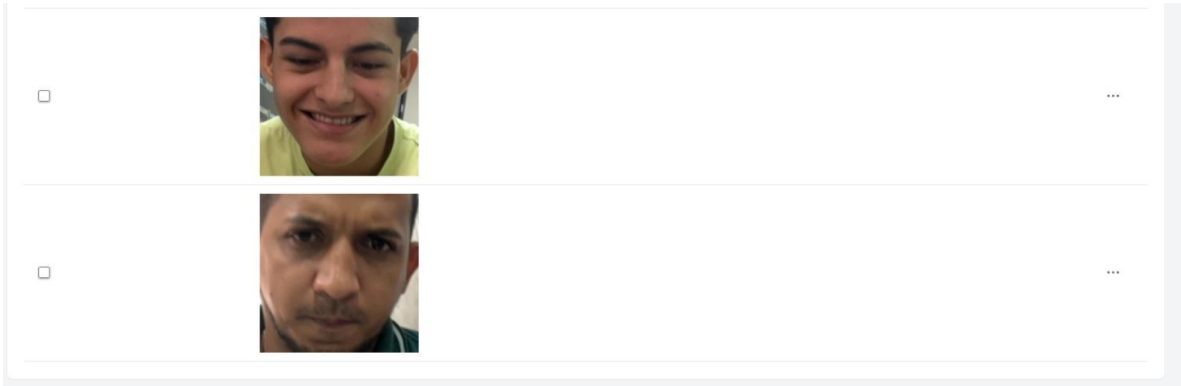
Detalle		Fecha	
☐	El usuario Lizandro - - lizandro.guerrero@u-site.app ha iniciado sesión	26/07/2024	...
☐	El usuario Lizandro - - lizandro.guerrero@u-site.app ha iniciado sesión	26/07/2024	...
☐	Valentina Olivar Gil, 1065867382, ha sido reconocido	18/09/2024	...
☐	Valentina Olivar Gil, 1065867382, ha sido reconocido	18/09/2024	...
☐	Ciro Perrony, 1092362816, ha sido reconocido	18/09/2024	...
☐	Valentina Olivar Gil, 1065867382, ha sido reconocido	18/09/2024	...
☐	Ciro Perrony, 1092362816, ha sido reconocido	02/08/2024	...
☐	Ciro Perrony, 1092362816, ha sido reconocido	02/08/2024	...
☐	Ciro Perrony, 1092362816, ha sido reconocido	02/08/2024	...
☐	Ciro Perrony, 1092362816, ha sido reconocido	02/08/2024	...
☐	Ciro Perrony, 1092362816, ha sido reconocido	02/08/2024	...
☐	Ciro Perrony, 1092362816, ha sido reconocido	02/08/2024	...
☐	Ciro Perrony, 1092362816, ha sido reconocido	02/08/2024	...
☐	Ciro Perrony, 1092362816, ha sido reconocido	02/08/2024	...

Fuente: elaboración propia.

El módulo de reportes del administrador es una funcionalidad clave dentro del sistema de reconocimiento facial, diseñada específicamente para proporcionar un registro detallado de los reconocimientos faciales realizados. Este módulo permite al administrador acceder de manera centralizada a una base de datos que almacena información precisa de cada reconocimiento facial, incluyendo la fecha, hora y ubicación de los eventos capturados por el software. De esta manera, facilita la generación de informes personalizados que permiten el análisis y monitoreo de patrones, frecuencia de reconocimientos exitosos, así como posibles fallos o inconsistencias en el sistema. De esta manera, el administrador tiene acceso a herramientas avanzadas de visualización de datos, lo que contribuye a la mejora

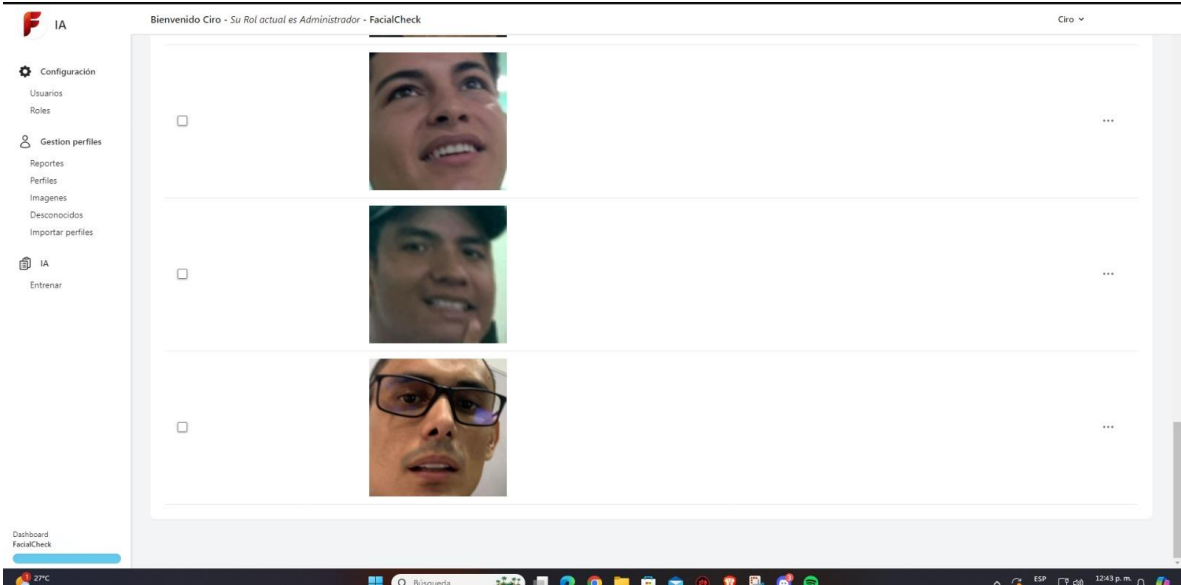
continua del sistema y a la toma de decisiones informadas respecto a su operación y seguridad.

Figura 47. Reporte de pruebas 2



Fuente: elaboración propia.

Figura 48. Reporte de pruebas 4



Fuente: elaboración propia.

El módulo de reportes, se encarga de gestionar las fotos de los reconocimientos faciales tomados, proporciona una interfaz robusta que permite visualizar y revisar las imágenes capturadas durante el proceso de identificación facial. A través de este módulo, el administrador puede acceder a un registro completo de las fotografías asociadas a cada evento de reconocimiento, facilitando la verificación visual de los resultados obtenidos por el sistema. Así logra almacenar y organizar las imágenes por parámetros como fecha, hora y usuario reconocido, el módulo ofrece opciones de búsqueda y filtrado avanzadas, lo que agiliza la localización de registros específicos. Esta funcionalidad no solo garantiza un control de los reconocimientos faciales realizados, sino que también refuerza la transparencia y trazabilidad de las operaciones del sistema.

Figura 49. Reporte de pruebas 3

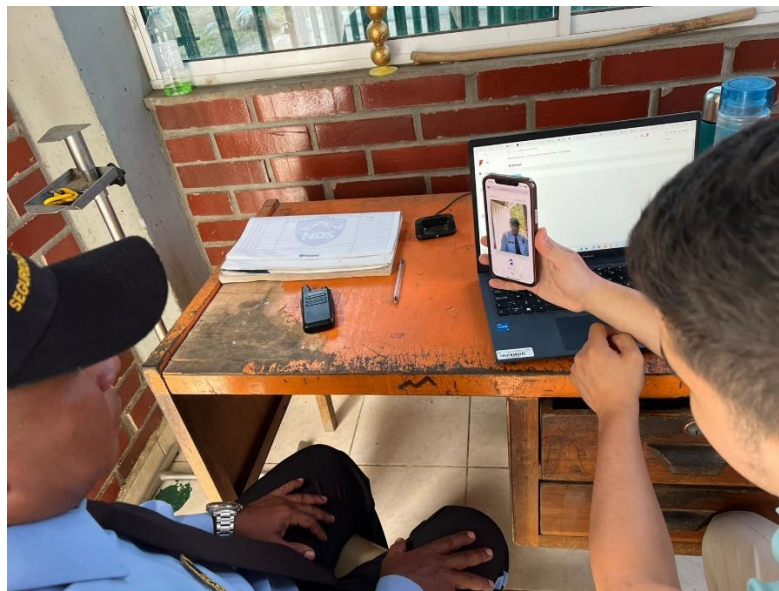
Reportes	
Logs	
Description de Logs	
Filtrar por fecha	
dd/mm/aaaa	
Exportar	
0 / 40	
Detalle	Fecha
☐ El usuario Ciro Perroy - cperroy@gmail.com ha iniciado sesión	18/06/2024
☐ El usuario Ciro Perroy - cperroy@gmail.com ha iniciado sesión	18/06/2024
☐ El usuario Ciro Perroy - cperroy@gmail.com ha iniciado sesión	03/06/2024
☐ El usuario Ciro Perroy - cperroy@gmail.com ha iniciado sesión	24/06/2024
☐ El usuario Lizandro - lizandro.guamizo@u-she.app ha iniciado sesión	14/06/2024
☐ El usuario Ciro Perroy - cperroy@gmail.com ha iniciado sesión	02/06/2024
☐ Eliminación de usuario Jhonatan Rey	01/06/2024
☐ El usuario Ciro Perroy - cperroy@gmail.com ha iniciado sesión	01/06/2024
☐ El usuario Ciro Perroy - cperroy@gmail.com ha iniciado sesión	31/07/2024
☐ El usuario Ciro Perroy - cperroy@gmail.com ha iniciado sesión	31/07/2024
☐ El usuario Ciro Perroy - cperroy@gmail.com ha iniciado sesión	31/07/2024

Fuente: elaboración propia.

El módulo de reportes de inicio de sesión está diseñado para proporcionar un registro exhaustivo de todas las veces que los usuarios han ingresado al sistema. A través de este módulo, el administrador puede acceder a un historial detallado que muestra las fechas y horas de cada sesión iniciada, así como la identidad del usuario correspondiente. Esta herramienta facilita el seguimiento y monitoreo de la

actividad dentro del sistema, permitiendo identificar patrones de uso, controlar el acceso, y detectar cualquier posible intento de ingreso no autorizado. Al ofrecer reportes claros y organizados, este módulo contribuye significativamente a la seguridad y transparencia operativa del sistema, garantizando que solo usuarios autorizados puedan acceder al mismo.

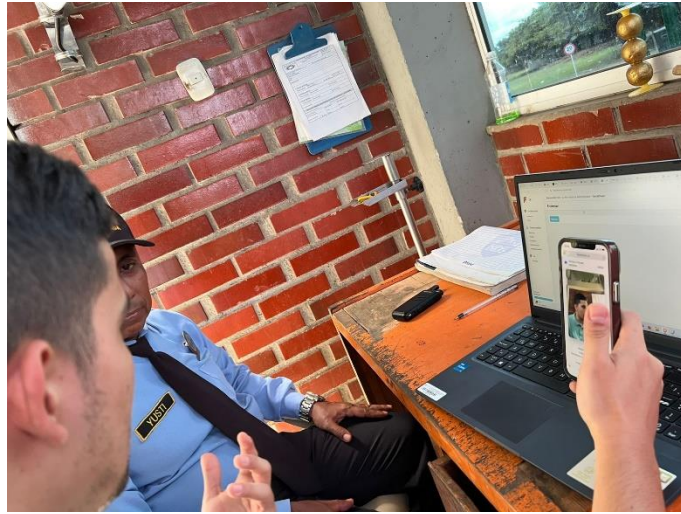
Figura 50. Socialización software parte 1



Fuente: elaboración propia.

El proceso de socialización del software de reconocimiento facial a los vigilantes de la Universidad Popular del Cesar, seccional Aguachica, representó una fase crucial en la implementación y eventual éxito del sistema. Esta etapa tuvo como objetivo principal capacitar al personal de vigilancia sobre el uso adecuado del software, su funcionamiento y los beneficios que ofrece en términos de seguridad y control de acceso. El enfoque estuvo centrado en garantizar que los vigilantes comprendieran a profundidad tanto el aspecto técnico como las implicaciones operativas del sistema, de manera que pudieran utilizarlo de forma eficiente en su labor cotidiana.

Figura 51. Socialización software parte 2



Fuente: elaboración propia.

En primer lugar, se socializo con los vigilantes donde pudieran familiarizarse con el software y entender su rol dentro del nuevo esquema de seguridad implementado por la universidad. Durante la socialización, se explicó en detalle cómo el sistema de reconocimiento facial captura imágenes en tiempo real, procesa los datos y verifica las identidades de las personas en base a una base de datos previamente registrada. Este proceso fue desglosado para que los vigilantes pudieran comprender las diferentes etapas, desde el momento en que una persona se encuentra frente a la cámara hasta la validación o rechazo de su identidad.

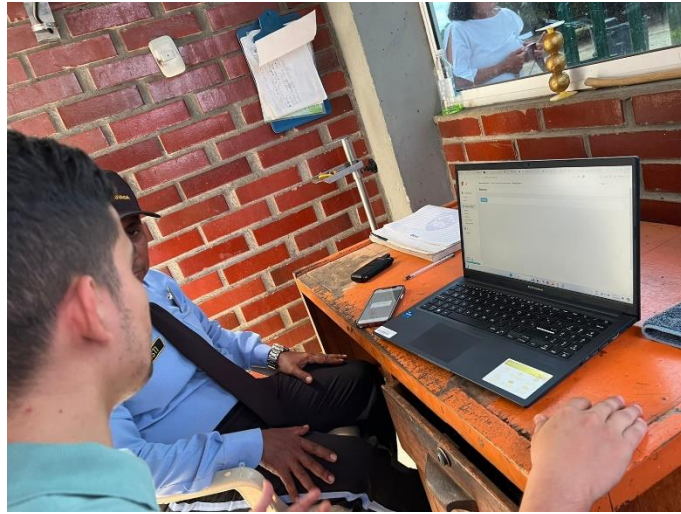
Figura 52. Socialización software parte 3



Fuente: elaboración propia.

Asimismo, se dedicó tiempo a explicar la interfaz de usuario del software, destacando la facilidad de su uso. Se realizó un recorrido detallado por los distintos módulos, particularmente el módulo de monitoreo en tiempo real y el módulo de alertas automáticas, que son de vital importancia para la labor de los vigilantes. Estos módulos permiten visualizar en pantalla las personas que ingresan a las instalaciones en tiempo real, alertando inmediatamente si se detecta un intento de ingreso por parte de una persona no autorizada o desconocida. Los vigilantes recibieron instrucciones detalladas sobre cómo actuar ante una alerta de este tipo, asegurando una respuesta rápida y adecuada en cada situación.

Figura 53. Socialización software parte 4



Fuente: elaboración propia.

El software, además de facilitar el control de acceso mediante reconocimiento facial, cuenta con funcionalidades avanzadas de gestión de usuarios, lo cual también fue socializado durante las capacitaciones. Se explicó cómo el sistema permite registrar y gestionar perfiles, actualizando las fotografías y datos de identificación de estudiantes, docentes, y personal administrativo de la universidad. Esto es particularmente útil para los vigilantes, quienes necesitan asegurarse de que solo individuos autorizados ingresen a áreas restringidas dentro del campus.

Figura 54. Socialización software parte 5



Fuente: elaboración propia.

Otro aspecto clave abordado durante la socialización fue la importancia de mantener la privacidad y confidencialidad de los datos recopilados por el software. Los vigilantes fueron informados sobre las normativas de protección de datos personales, enfatizando que la información capturada por el sistema, especialmente las imágenes faciales, está sujeta a estrictos protocolos de seguridad. Se proporcionó pautas claras sobre cómo manejar la información de manera responsable, asegurando que el uso del sistema se alinee con las leyes de protección de datos vigentes, como la Ley 1581 de 2012 en Colombia.

Figura 55. Socialización software parte 6



Fuente: elaboración propia.

A lo largo de este proceso de socialización, se promovió una actitud crítica y reflexiva por parte de los vigilantes, quienes fueron incentivados a expresar sus inquietudes, opiniones y sugerencias sobre el software. Esto permitió ajustar y mejorar algunos aspectos de la interfaz y del proceso operativo en función de las necesidades específicas del personal de seguridad. La socialización fue diseñada de manera participativa, lo que favoreció el intercambio de ideas y fortaleció la apropiación del sistema por parte de los vigilantes.

5. Conclusiones

A lo largo del proyecto de investigación, se definieron de manera precisa los requerimientos funcionales y no funcionales necesarios para el desarrollo del sistema, tomando en cuenta las necesidades específicas de la institución. Estos requerimientos permitieron establecer un marco técnico adecuado que asegurara tanto la operatividad del software como su compatibilidad con los sistemas ya existentes en la universidad. De esta manera, se logró estructurar una guía clara para el diseño y desarrollo del sistema, garantizando que el software pudiera cumplir con las expectativas funcionales y operativas necesarias para su implementación efectiva en el entorno universitario.

En cuanto al diseño del software, se implementaron algoritmos avanzados y librerías basadas en inteligencia artificial (IA) y técnicas de Machine Learning, lo cual fue fundamental para la construcción de un sistema de reconocimiento facial robusto y eficiente. Este diseño no solo permitió desarrollar un modelo preciso de identificación, sino que también garantizó la escalabilidad y adaptabilidad del sistema a diferentes condiciones, como variaciones en la iluminación o en las características faciales de los usuarios. Posteriormente, el software fue integrado exitosamente a una interfaz de usuario intuitiva, la cual fue diseñada para facilitar su uso tanto por el personal de vigilancia como por los administradores del sistema.

Finalmente, se realizó una prueba piloto en la Universidad Popular del Cesar, seccional Aguachica, para evaluar la funcionalidad y aceptación del software entre los usuarios. Los resultados de esta fase confirmaron el correcto funcionamiento del sistema y una alta aceptación por parte de la comunidad universitaria, lo que valida el cumplimiento del último objetivo específico del proyecto. En conclusión, el desarrollo de este software de control de acceso mediante reconocimiento facial representa un avance significativo en la seguridad del campus, demostrando la eficacia de las tecnologías de Machine Learning para resolver problemas complejos en entornos académicos.

6. Recomendaciones

- Se recomienda incorporar simulaciones y prácticas regulares que permitan al personal familiarizarse con posibles incidentes de seguridad o fallos técnicos, fortaleciendo su capacidad de respuesta ante estos escenarios.
- Es importante asegurar que haya un respaldo técnico constante, para solucionar posibles fallos de hardware o software. Una conexión estable y segura permitirá que el sistema opere de manera fluida y que los datos se almacenen de forma segura, previniendo problemas de latencia o pérdida de información crucial.
- Se recomienda la implementación de un plan de mantenimiento preventivo y la recolección de retroalimentación del personal y los estudiantes asegurarán la optimización continua del sistema.

7. Bibliografía

- Acevedo, E. S. (2017). Principios y características de las redes neuronales artificiales. *Desarrollo e innovación en ingeniería*, 173.
- Alcaldía de Aguachica, C. (2022). GOV. Obtenido de <http://www.aguachica-cesar.gov.co/municipio/nuestro-municipio>
- Alliance, A. (2023). *Agile Alliance*. Obtenido de Agile Alliance: [https://www.agilealliance.org/glossary/xp/#q=~\(infinite~false~filters~\(postType~\(~'post~'aa_book~'aa_event_session~'aa_experience_report~'aa_glossary~'aa_research_paper~'aa_video\)~tags~\(~'xp\)\)~searchTerm~'~sort~false~sortDirection~'asc~page~1\)](https://www.agilealliance.org/glossary/xp/#q=~(infinite~false~filters~(postType~(~'post~'aa_book~'aa_event_session~'aa_experience_report~'aa_glossary~'aa_research_paper~'aa_video)~tags~(~'xp))~searchTerm~'~sort~false~sortDirection~'asc~page~1))
- Aloson Sierra, J. D., & Castaño Saavedra, D. L. (2019). *Repository Ucatolica*. Obtenido de Repository Ucatolica: <https://repository.ucatolica.edu.co/server/api/core/bitstreams/c627cd7f-698d-4dc8-b552-736c318cae4c/content>
- Amazon. (2023). *Amazon Web Services*. Obtenido de Amazon Web Services: <https://aws.amazon.com/es/what-is/neural-network/>
- Becerra, B. X. (27 de 4 de 2022). Los países con los índices más altos de criminalidad organizada en todo el mundo. *La Republica*.
- Caballero Barriga, E. R. (2017). *Aplicación práctica de la visión artificial para el reconocimiento de rostros en una imagen, utilizando redes neuronales y algoritmos de reconocimiento de objetos de la biblioteca opencv*.
- CEJ . (2022). *Tasa de criminalidad en Colombia*.
- CEJ. (2022). *Hacinamiento carcelario en Colombia*.
- CEJ. (2022). *Libertades por vencimiento de términos en Colombia*.
- Cerezo , R., & Enríquez, C. (2024). Software de control de asistencia por reconocimiento facial, para dependencias del gobierno. *Revista Ecuatoriana*

- de *Derecho y Administración*, 1(1), 62-86. Obtenido de <https://revistas.itecsur.edu.ec/index.php/reda/article/view/126/81>
- Chaviano Arteaga, H. (2015). *Dianet*. Obtenido de <https://dialnet.unirioja.es/servlet/articulo?codigo=7192675>
- Code, V. S. (2022). *Visual Studio Code*. Obtenido de <https://code.visualstudio.com/docs>
- Congreso de Colombia. (2009). *Sic*. Obtenido de Sic: https://www.sic.gov.co/recursos_user/documentos/normatividad/Ley_1273_2009.pdf
- Congreso de la Republica. (2012). *Funcion Publica*. Obtenido de Funcion Publica: <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=49981>
- Congreso de la república de Colombia. (1982). *Alcaldia Bogota* . Obtenido de Alcaldia Bogota : <https://www.alcaldiabogota.gov.co/sisjur/normas/Norma1.jsp?i=3431&dt=S>
- Diaz Bravo, L., Turroco Garcia, U., Martinez Hernandez, M., & Varela Ruiz, M. (2013). *Scielo*. Obtenido de https://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S2007-50572013000300009#:~:text=Se%20argumenta%20que%20la%20entrevista,proceso%2C%20asegurando%20respuestas%20m%C3%A1s%20%C3%BAtiles.
- EcuRed. (2018). *Municipio Aguachica*. Obtenido de https://www.ecured.cu/Municipio_Aguachica
- EcuRed. (2022). *Ecured*. Obtenido de https://www.ecured.cu/Municipio_Aguachica
- Esan. (2017). *Esan* . Obtenido de Esan : <https://www.esan.edu.pe/conexion-esan/que-es-el-scrum-y-por-que-es-importante-desarrollarlo-en-un-proyecto#:~:text=Scrum%20es%20%C3%BAtil%20para%20desarrollar,la%20flexibilidad%20y%20la%20productividad>.

- García Falconí, R., & Barona Pazmiño, K. (2024). Inteligencia artificial y proceso penal. *Revista San Gregorio*, 1(58), 101-110. doi:<http://dx.doi.org/10.36097/rsan.v1i58.2808>
- Gardner, H. (1987). La teoría de las inteligencias múltiples. En H. Gardner, *Frames of Mind: The Theory of Multiple Intelligences* (págs. 287-305). Santiago de Chile: Instituto Construir. Obtenido de https://d1wqtxts1xzle7.cloudfront.net/51558533/La_Teoria_de_las_Inteligencias_Multiples_cortad-with-cover-page-v2.pdf?Expires=1669917342&Signature=lbyxmFwsTZifz2PIAa6ksqYGka6szOk6rxdUmuytDNTDeD4toW7aq0Z5dwGgz0nSJxJP7gtHZxnG3gjmK0OVnTu1aDEB6wQo0jmfOx6dE0qB
- Gestión, R. (2018). ¿ Qué es la inteligencia artificial y para qué sirve?. *Gestión*, 1. Obtenido de https://campus.mec.gob.ar/pluginfile.php/209250/mod_resource/content/1/Actividad%206%20-%20%C2%BFQu%C3%A9%20es%20la%20inteligencia%20artificial%20y%20para%20qu%C3%A9%20sirve_%20_%20Tecnolog%C3%ADa%20_%20Gesti%C3%B3n.pdf
- Gob, D. (Mayo de 2022). *Datos Gob*. Obtenido de <https://datos.gob.es/es/blog/11-librerias-para-crear-visualizaciones-de-datos#:~:text=Las%20librer%C3%ADas%20de%20programaci%C3%B3n%20son,resueltas%20previamente%20por%20otros%20programadores.>)
- Gómez Alzate, P. A. (2022). *Biblioteca Digital UDEA*. Obtenido de Biblioteca Digital UDEA: https://bibliotecadigital.udea.edu.co/bitstream/10495/29020/1/GomezPaula_2022_AplicacionMovilControl.pdf
- Gonzales Muñoz, A. (2018). *Aplicaciones de tecnicas de inteligencia artificial basadas en aprendizaje profundo (deep learning) al analisis y mejora de la*

eficiencia de procesos industriales. Obtenido de https://www.researchgate.net/profile/Ana-Gonzalez-Muniz/publication/334328255_Aplicaciones_de_tecnicas_de_inteligencia_artificial_basadas_en_aprendizaje_profundo_deep_learning_al_analisis_y_mejora_de_la_eficiencia_de_procesos_industriales/links/5d24c78ca6

González Esquivel, B., & González Contreras, B. (2024). Acceso automático por identificación de uso adecuado de cubrebocas. *Pistas Educativas*, 45(147), 560-578. Obtenido de <https://pistaseducativas.celaya.tecnm.mx/index.php/pistas/article/view/3423/2537>

Guallasamin Chasi, A., & Guallichico Guallichico, B. (2024). Desarrollo de un sistema de seguridad para controlar el encendido de un vehículo subcategoría L3, basado en un método de autenticación por reconocimiento facial. *Universidad Politécnica Salesiana*, 1-71. Obtenido de <https://dspace.ups.edu.ec/bitstream/123456789/27051/1/TTS1629.pdf>

Hostinger. (2022). *Hostinger*. Obtenido de <https://www.hostinger.co/tutoriales/que-es-cli>

IESC, A. M. (2018). *master-iesc-angers*. Obtenido de <https://master-iesc-angers.com/artificial-intelligence-machine-learning-and-deep-learning-same-context-different-concepts/>

informacion, T. (2018). *Tecnologias informacion*. Obtenido de <https://www.tecnologias-informacion.com/metodo-dinamico.html>

Jiménez García, J., Amézquita Sánchez, J., & Valtierra Rodríguez, M. (2203). Reconocimiento facial basado en aprendizaje profundo para aplicaciones de seguridad automotriz. *Jóvenes en la Ciencia*, 25, 1-15. Obtenido de <http://repositorio.ugto.mx/bitstream/20.500.12059/10515/1/Reconocimiento%20facial%20basado%20en%20aprendizaje%20profundo%20para%20aplicaciones%20de%20seguridad%20automotriz.pdf>

- Juanazo Arriaga, B., & Lamboggia Yar, M. (2024). Diseño e implementación de un sistema de reconocimiento facial mediante P5.JS y Teachable Machine para la apertura de una puerta. *Universidad Politécnica Salesiana Ecuador*, 1-62. Obtenido de <https://dspace.ups.edu.ec/bitstream/123456789/27738/1/UPS-GT005232.pdf>
- Kaspersky. (2022). *Kaspersky*. Obtenido de <https://latam.kaspersky.com/resource-center/definitions/biometrics>
- La Madrid Coz, G., & Ruiz Toledo, R. (2023). Implementación de un sistema de autenticación mediante validación biométrica para procesos bancarios. *Universidad Peruana de Ciencias Aplicadas*, 1-55. Obtenido de https://repositorioacademico.upc.edu.pe/bitstream/handle/10757/671962/La%20Madrid_CG.pdf?sequence=1&isAllowed=y
- Martins, J. (2022). *Asana*. Obtenido de Asana : <https://asana.com/es/resources/what-is-kanban>
- Masters, T. (2023). *Tesis Masters*. Obtenido de Tesis Masters: <https://tesisymasters.com.co/poblacion/>
- Medrano, C., & Igual, R. (s.f.). *Tutorial de OpenCV*. Obtenido de https://josbram.delifrut.cl/files/openCV/tutorial_opencv.pdf
- Mendieta, A. (2022). *Open Webinars*. Obtenido de Open Webinars: <https://openwebinars.net/blog/metodologia-lean-que-es-y-como-implementarla-en-empresas-it/>
- Millán, E. H. (2020). Desarrollo de Software a partir del ABP. Polinnova,. *Revista Piensapinter*, 4(1).
- Molina, C. (2020). *Universidad Popular del Cesar - Misión y Visión*. Obtenido de Unicesar.edu.co.: <https://www.unicesar.edu.co/index.php/es/mision-y-vision>

- Mousavi, e. a. (2018). *Aprendizaje por Refuerzo para robotica de enjambre*.
Obtenido de Aprendizaje por Refuerzo para robotica de enjambre:
<https://web.ua.es/va/phdinf/documentos/jdi-2022/alvarodiezdiaz.pdf>
- Muñoz Rios, L., & Ponce Almeida, J. (2024). Sistema Biométrico para el control de acceso a la UDIV de desarrollo computacional de la ESPAM-MFL empleando técnicas de Deep Learning. *Escuela Superior Politécnica Agropecuaria de Manabí*, 1-63. Obtenido de https://repositorio.espam.edu.ec/bitstream/42000/2349/1/TIC_C21D.pdf
- Navas-Triana, H. E.-P. (2021). *Desarrollo e implementación de un prototipo funcional de software basado en estándares de calidad, que permita la gestión documental para el levantamiento de requerimientos funcionales y no funcionales en la empresa TICS SAS*.
- OkDiario. (2019). *OkDiario*. Obtenido de <https://okdiario.com/curiosidades/conoce-metodo-observacion-directa-3628568>
- Oliva Rodriguez, A. (2018). *Desarrollo de una aplicacion de reconocimiento en imagenes utilizando deep leaning con OpenCv*. Obtenido de <https://riunet.upv.es/bitstream/handle/10251/107243/OLIVA%20-%20Desarrollo%20de%20una%20aplicaci%3%b3n%20de%20reconocimie%20en%20im%3%a1genes%20utilizando%20Deep%20Learning%20con%20O....pdf?sequence=1&isAllowed=y>
- Parra, A. (2023). *Quiestion Pro*. Obtenido de Quiestion Pro: <https://www.questionpro.com/blog/es/muestra-representativa-para-investigacion/>
- Pilay Ríos, D., & Alban Gómez, J. (2024). Diseño e implementación de un sistema de acceso peatonal mediante reconocimiento facial utilizando Teachable Machine. *Universidad Politécnica Salesiana Ecuador*, 1-90. Obtenido de <https://dspace.ups.edu.ec/bitstream/123456789/27727/1/UPS-GT005223.pdf>

Portilla Jimenez, J. J. (2018). *Análisis y Diseño de un Sistema de Reconocimiento Facial aplicando Machine Learning para detectar e identificar intrusos*. Guayaquil. Obtenido de repositorio.ug.edu.ec/bitstream/redug/32954/1/B-CINT-PTG-N.297%20Portilla%20Jimenez%20Jonathan%20Javier.pdf

Pypi. (2022). *Pypi*. Obtenido de <https://pypi.org/project/pip/>

Ramírez, D. H. (2018). *El Machine Learning a través de los tiempos, y los aportes a la humanidad*. Obtenido de <https://repository.unilibre.edu.co/handle/10901/17289>

Rouhiainen, L. (2018). *Inteligencia artificial*. Madrid: Alienta Editorial.

Santander Universidades. (Abril de 2021). *Santander*. Obtenido de <https://www.becas-santander.com/es/blog/python-que-es.html>

Scrum. (2023). *Scrum* . Obtenido de Scrum : <https://www.scrum.org/learning-series/what-is-scrum>

Serratosa, F. (s.f.). *La biometria para la identificacion de las personas*. Obtenido de https://sistemamid.com.ar/panel/uploads/biblioteca/2015-03-22_12-05-01117594.pdf

Significados. (2023). *Significados* . Obtenido de Significados : <https://www.significados.com/base-de-datos/>

SurveyMonkey. (2021). *SurveyMonkey*. Obtenido de <https://es.surveymonkey.com/mp/sample-size-calculator/>

Technical Report. (2019). *Technical Report*. Obtenido de https://www.researchgate.net/profile/Batta-Mahesh/publication/344717762_Machine_Learning_Algorithms_-_A_Review/links/5f8b2365299bf1b53e2d243a/Machine-Learning-Algorithms-A-Review.pdf?eid=5082902844932096

TecnoSeguro. (s.f.). *TecnoSeguro*. Obtenido de <https://www.tecnoseguro.com/faqs/control-de-acceso/que-es-un-control-de-acceso>

Universidad Popular del Cesar, S. (2022). *Aguachica Unicesar*. Obtenido de <http://aguachica.unicesar.edu.co/unidad-visionaria/>

Valencia Rodríguez, G. E., & Brito Poveda, V. P. (2015). *Escuela Superior Politécnica de Chimborazo*. Obtenido de Escuela Superior Politécnica de Chimborazo: <http://dspace.esPOCH.edu.ec/handle/123456789/5020>

Manual de Usuario: Software Para El Control De Acceso A Través De Reconocimiento Facial Mediante Machine Learning

Desarrollado con: Laravel, MongoDB, y librerías de Python

Institución: Universidad Popular del Cesar Seccional Aguachica

Índice

- 1. Introducción**
- 2. Requisitos del Sistema**
- 3. Instalación**
- 4. Configuración Inicial**
- 5. Uso del Software**
 - 5.1 Acceso al Sistema
 - 5.2 Registro de Usuarios
 - 5.3 Reconocimiento Facial
 - 5.4 Monitoreo de Accesos
- 6. Mantenimiento y Soporte**
- 7. FAQs**
- 8. Contacto**

1. Introducción

Este manual tiene como objetivo guiar al usuario en el uso del software de control de acceso mediante reconocimiento facial. Este sistema está diseñado para mejorar la seguridad en la Universidad Popular del Cesar, Seccional Aguachica, utilizando tecnología de Machine Learning para el reconocimiento de personas autorizadas.

2. Requisitos del Sistema

- **Hardware:**

- Procesador: Intel i5 o equivalente.
- RAM: 8 GB como mínimo.
- Almacenamiento: 100 GB libres.
- Cámara web de alta definición.

- **Software:**

- Sistema Operativo: Windows 10 o superior, Ubuntu 20.04 o superior.
- PHP 7.4 o superior.
- Composer.
- MongoDB 4.0 o superior.
- Python 3.7 o superior con las librerías necesarias (OpenCV, dlib).

3. Instalación

1. Clonar el repositorio:

bash

Copiar código

git clone <url_del_repositorio>

2. Instalar dependencias:

bash

Copiar código

```
cd <nombre_del_proyecto>
```

```
composer install
```

3. Configurar el entorno:

- Copiar el archivo .env.example a .env y configurar las variables de entorno.

4. Instalar MongoDB:

- Seguir las instrucciones de instalación específicas para su sistema operativo.

5. Ejecutar migraciones:

```
bash
```

Copiar código

```
php artisan migrate
```

4. Configuración Inicial

- **Conexión a MongoDB:** Asegúrate de que el servicio de MongoDB esté corriendo y que las credenciales en el archivo .env sean correctas.
- **Configuración de la cámara:** Verifica que la cámara esté funcionando correctamente y que el sistema tenga acceso a ella.

5. Uso del Software

5.1 Acceso al Sistema

1. Abrir un navegador web y navegar a la URL del sistema.
2. Iniciar sesión con las credenciales proporcionadas.

5.2 Registro de Usuarios

1. En el menú principal, seleccionar "Registrar Usuario".

2. Ingresar los datos solicitados (nombre, apellidos, correo electrónico, etc.).
3. Hacer clic en "Registrar". Se capturará la imagen del rostro del usuario.

5.3 Reconocimiento Facial

1. Al intentar acceder, el sistema capturará la imagen del rostro en tiempo real.
2. El software comparará la imagen con los rostros registrados en la base de datos.
3. Se mostrará un mensaje de acceso autorizado o denegado.

5.4 Monitoreo de Accesos

1. Seleccionar "Registro de Accesos" en el menú.
2. Podrás visualizar un historial de accesos, incluyendo fechas, horas y usuarios.

6. Mantenimiento y Soporte

- **Actualizaciones:** Mantén el sistema actualizado con las últimas versiones de Laravel, MongoDB y librerías de Python.
- **Backup de Datos:** Realiza copias de seguridad periódicas de la base de datos.

7. FAQs

- **¿Qué hacer si la cámara no funciona?**
 - Verifica que esté correctamente conectada y que los permisos estén habilitados en el sistema operativo.
- **¿Cómo restablecer la contraseña?**
 - Utiliza la opción "¿Olvidaste tu contraseña?" en la página de inicio de sesión.