UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI

FACULTAD DE INDUSTRIAS AGROPECUARIAS Y CIENCIAS AMBIENTALES

CARRERA DE COMPUTACIÓN

Tema: "Reconocimiento facial para la verificación de identidad de estudiantes en exámenes en línea"

Trabajo de Integración Curricular previo a la obtención del título de Ingeniero en Ciencias de la Computación

AUTOR: Ruales Yucás Galo David

TUTOR: Ing. Yandún Velasteguí Marco Antonio, MSc

Tulcán, 2026.

CERTIFICADO DEL TUTOR

Certifico que el estudiante Ruales Yucás Galo David con el número de cédula 0402057921. Respectivamente ha desarrollado el Trabajo de Integración Curricular: "Reconocimiento facial para la verificación de identidad de estudiantes en exámenes en línea".

Este trabajo se sujeta a las normas y metodología dispuesta en la Codificación del Reglamento de Régimen Académico y de Estudiantes de la UPEC, por lo tanto, autorizo la presentación de la sustentación para la calificación respectiva.

Firmado electrónicamente por:
MARCO ANTONIO
YANDUN VELASTEGUI
Validar únicamente con FirmaEC

_____

Ing. Yandún Velasteguí Marco Antonio, MSc

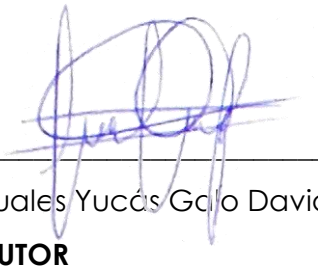
TUTOR

Tulcán, enero de 2026

AUTORÍA DE TRABAJO

El presente Trabajo de Integración Curricular constituye un requisito previo para la obtención del título de Ingeniero en la Carrera de Computación de la Facultad de Industrias Agropecuarias y Ciencias Ambientales

Yo, Ruales Yucás Galo David con cédula de identidad número 0402057921 respectivamente declaro que la investigación es absolutamente original, auténtica, personal y los resultados y conclusiones a los que he llegado son de mi absoluta responsabilidad.

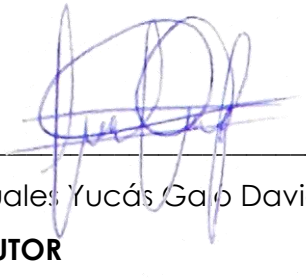


Ruales Yucás Galo David
AUTOR

Tulcán, enero de 2026

ACTA DE CESIÓN DE DERECHOS DEL TRABAJO DE INTEGRACIÓN CURRICULAR

Yo Ruales Yucás Galo David declaro ser autor de los criterios emitidos en el Trabajo de Integración Curricular: "Reconocimiento facial para la verificación de identidad de estudiantes en exámenes en línea" y eximo expresamente a la Universidad Politécnica Estatal del Carchi y a sus representantes de posibles reclamos o acciones legales.



Ruales Yucás Galo David
AUTOR

Tulcán, enero de 2026

AGRADECIMIENTO

En primer lugar, agradezco a la Universidad Politécnica Estatal del Carchi y a la Carrera de Computación por haberme brindado las herramientas y conocimientos necesarios para mi formación como profesional.

Un agradecimiento especial a mi tutor MSc. Marco Yandún, por su guía experta, paciencia y valiosos consejos durante todo el desarrollo de este trabajo de integración curricular; su apoyo fue fundamental para llevar este proyecto a un término exitoso.

DEDICATORIA

Dedico este trabajo de investigación, que representa el fruto de mi esfuerzo y aprendizaje, a las personas más importantes de mi vida:

A mis padres, David Ruales y Alba Yucás, quienes, con su amor incondicional, esfuerzo y apoyo constante han sido el motor de mi existencia y el pilar fundamental para alcanzar esta meta profesional. Gracias por ser mi ejemplo de integridad, por enseñarme el valor de la perseverancia y por creer siempre en mis sueños, incluso cuando el camino se tornaba difícil. Todo este logro es, en gran medida, gracias a ustedes.

ÍNDICE

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

ÍNDICE DE ANEXOS

RESUMEN

La suplantación de identidad en exámenes virtuales representa un desafío significativo para la integridad académica en instituciones de educación superior. Esta investigación se centró en el desarrollo de un sistema de verificación biométrica basado en reconocimiento facial mediante aprendizaje profundo, integrado nativamente con Moodle. La metodología empleó un enfoque cuantitativo con tres revisiones sistemáticas y encuesta a 213 estudiantes, revelando un Índice de Prevalencia del Fraude del 47.1% y un Índice de Confianza en la Tecnología del 69.8%. El sistema consta de dos componentes principales: un servidor Flask que implementa InsightFace mediante arquitectura ArcFace con RetinaFace, complementado por FaceNet como respaldo más MiniFASNet para detección anti-suplantación de identidad; junto con un plugin PHP para Moodle que realiza captura facial en tiempo real. Se concluye que el sistema constituye una alternativa viable y escalable para fortalecer la seguridad en evaluaciones remotas.

Palabras Claves: reconocimiento facial, aprendizaje profundo, integridad académica, evaluación en línea, Moodle, InsightFace.

ABSTRACT

Identity theft in online exams poses a significant challenge to academic integrity in higher education institutions. This research focused on developing a biometric verification system based on facial recognition using deep learning, natively integrated with Moodle. The methodology employed a quantitative approach with three systematic reviews and a survey of 213 students, revealing a Fraud Prevalence Index of 47.1% and a Technology Trust Index of 69.8%. The system consists of two main components: a Flask server implementing InsightFace using an ArcFace architecture with RetinaFace, complemented by FaceNet as a backup and MiniFASNet for anti-spoofing detection; along with a PHP plugin for Moodle that performs real-time facial capture. The study concludes that the system constitutes a viable and scalable alternative for strengthening security in remote assessments.

Keywords: facial recognition, deep learning, academic integrity, online assessment, Moodle, InsightFace.

INTRODUCCIÓN

El reconocimiento facial se ha convertido en una tecnología fundamental para garantizar la integridad académica en entornos educativos virtuales. Esta solución tecnológica permite verificar la identidad de los estudiantes que realizan evaluaciones a distancia, asegurando que quien presenta el examen es realmente quien dice ser (García-Peñalvo et al., 2021).

En el contexto actual, donde la educación en línea ha experimentado un crecimiento exponencial, las instituciones educativas buscan implementar mecanismos que mantengan la validez y confiabilidad de sus procesos de evaluación. Según Dawson (2021), "la verificación de identidad representa uno de los desafíos más significativos en la evaluación remota, ya que la distancia física imposibilita los métodos tradicionales de supervisión".

El reconocimiento facial funciona mediante algoritmos de inteligencia artificial que capturan y analizan las características faciales del estudiante, comparándolas con una imagen de referencia previamente registrada (Zhao et al., 2023). Durante el examen, el sistema puede realizar verificaciones continuas o aleatorias para confirmar que el mismo estudiante permanece frente a la pantalla durante toda la evaluación.

"Los sistemas modernos de reconocimiento facial pueden detectar con precisión superior al 99% la correspondencia entre el rostro del usuario y su identidad registrada, incluso en condiciones variables de iluminación" (Li & Jain, 2022). Esta precisión ha convertido esta tecnología en una alternativa viable para los métodos tradicionales de supervisión presencial.

I. EL PROBLEMA

1.1. PLANTEAMIENTO DEL PROBLEMA

A nivel mundial, la educación en línea ha experimentado un crecimiento extraordinario, particularmente impulsado por la pandemia de COVID-19, que ha transformado radicalmente el panorama educativo global. Singh et al. (2021) documentan cómo la pandemia forzó la digitalización acelerada de la educación superior, lo que resultó en un incremento de casi el doble en las inscripciones a programas virtuales comparado con periodos anteriores. Esta rápida transición ha generado importantes retos relacionados con la integridad académica. De acuerdo con Moubayed et al. (2020), uno de los principales desafíos técnicos y éticos que enfrentan las instituciones educativas globalmente es lograr verificar eficazmente la identidad de los estudiantes durante las evaluaciones en línea. Un informe reciente de la UNESCO (2023) revela que más de tres cuartas partes de las instituciones de educación superior a nivel mundial identifican la verificación de identidad como su principal inquietud en contextos de evaluación remota.

A nivel institucional, existe una creciente presión sobre las instituciones educativas para implementar sistemas de verificación fiables que salvaguarden la credibilidad de sus programas y titulaciones. Ullah et al. (2022) han documentado que las universidades sin métodos sólidos de autenticación experimentan una reducción significativa en la percepción del valor de sus titulaciones por parte del sector empleador. En este nivel intermedio, los organismos reguladores y de acreditación han elevado sus estándares considerablemente. Según documenta la Comisión de Acreditación de Educación Superior (2024), los programas virtuales deben ahora implementar mecanismos de verificación de identidad estudiantil tan rigurosos como los utilizados en entornos presenciales. La investigación de Guerrero-Roldán y Noguera (2021) indica que aproximadamente la mitad de los estudiantes reconoce que la ausencia de supervisión durante evaluaciones en línea constituye un incentivo para realizar prácticas académicas deshonestas.

A nivel individual, tanto estudiantes como profesores sufren directamente el impacto de carecer de sistemas adecuados de verificación. Karim et al. (2023) han encontrado que una proporción mayoritaria de estudiantes considera que las evaluaciones en línea sin sistemas robustos de verificación crean situaciones de desventaja para quienes actúan con integridad académica. Por su parte, según la investigación de Yaman y Koyuncu (2022), la gran mayoría del profesorado universitario vincula directamente la integridad de las evaluaciones virtuales con la implementación de sistemas efectivos de verificación de identidad. Respecto a la experiencia del usuario, Lai et al. (2021) han observado que los sistemas de reconocimiento facial necesitan encontrar un equilibrio entre seguridad y usabilidad, ya que un número significativo de estudiantes manifiesta incomodidad frente a procedimientos de verificación que consideran invasivos.

En este escenario complejo, la tecnología de reconocimiento facial se posiciona como una alternativa prometedora, aunque presenta sus propios desafíos. García-Peñalvo y Corell (2021) sugieren que los sistemas biométricos basados en reconocimiento facial proporcionan un balance ideal entre precisión y eficiencia de recursos para verificar identidades durante exámenes virtuales. Sin embargo, estos sistemas enfrentan importantes desafíos técnicos y sociales. Además, Castro et al. (2024) señalan que la brecha digital representa un obstáculo adicional, pues aproximadamente una cuarta parte de los estudiantes en países en desarrollo no cuenta con el equipamiento tecnológico necesario para implementar estos sistemas adecuadamente.

1.2. FORMULACIÓN DEL PROBLEMA

La ausencia de un software basado en reconocimiento facial limita la autenticación efectiva de los evaluados, lo que conlleva a una verificación de identidad poco fiable en los exámenes en línea, dando lugar a conductas académicas deshonestas.

1.3. JUSTIFICACIÓN

El desarrollo de un software de reconocimiento facial para el entorno virtual de aprendizaje es sumamente relevante debido a la necesidad de incorporar tecnologías avanzadas de autenticación en los procesos de evaluación en línea y mejorar la confiabilidad en la verificación de identidad de los estudiantes. Este nuevo módulo optimizaría los exámenes virtuales, minimizaría los riesgos de fraude

académico derivados de métodos manuales y ofrecería una solución más segura tanto para los docentes como para los alumnos.

El impacto de esta propuesta se reflejaría en un aumento de la calidad y validez de las evaluaciones en línea. Los resultados beneficiarían a las instituciones educativas y al cuerpo docente al facilitar la supervisión de exámenes y elevar los estándares de integridad académica. Al reducir las suplantaciones de identidad y automatizar el proceso de verificación, se agilizaría la administración de evaluaciones, mejorando la experiencia de todos los usuarios del entorno virtual de aprendizaje.

Además, esta solución contribuiría a resolver la falta de herramientas nativas de autenticación biométrica en el entorno virtual de aprendizaje, permitiendo una gestión más confiable y eficiente de los exámenes en línea. La propuesta del software reduciría la carga de supervisión manual y permitiría un mejor control de los procesos evaluativos, lo que beneficiaría tanto a los administradores del sistema como a los usuarios finales.

El desarrollo propuesto de este software proporcionaría un marco de referencia que abarcaría desde el diseño hasta la conceptualización de soluciones de inteligencia artificial aplicadas a la educación virtual. Los conocimientos que se obtendrían también servirían para explorar nuevas formas de brindar mayor seguridad en las evaluaciones en línea y aumentar la eficiencia académica mediante el uso de tecnologías de reconocimiento facial integradas en plataformas educativas.

Para esta propuesta, cuento con el apoyo de la Unidad de Tecnología Educativa, lo que garantizaría el acceso al ambiente tecnológico de desarrollo necesario para diseñar eficazmente esta solución.

1.4. OBJETIVOS Y PREGUNTAS DE INVESTIGACIÓN

1.4.1. Objetivo General

Desarrollar una herramienta de software basada en reconocimiento facial que permita verificar la identidad de los estudiantes en evaluaciones en línea.

1.4.2. Objetivos Específicos

- Identificar herramientas de visión artificial y reconocimiento facial utilizadas para prevenir la suplantación de identidad durante exámenes en línea.
- Analizar los desafíos y limitaciones de estudios previos sobre el desarrollo de sistemas de reconocimiento facial en exámenes virtuales

- Proponer un software funcional que se integre con el entorno virtual de aprendizaje, permitiendo la verificación de identidad mediante reconocimiento facial en los exámenes en línea.

1.4.3. Preguntas de Investigación

¿Qué herramientas de visión artificial y reconocimiento facial se han utilizado para prevenir la suplantación de identidad en exámenes en línea?

¿Cuáles son los principales desafíos y limitaciones identificados en la literatura científica sobre el desarrollo de sistemas de reconocimiento facial para la verificación de identidad en pruebas virtuales?

¿Cómo puede integrarse un sistema de reconocimiento facial en un entorno virtual de aprendizaje para verificar la identidad de los estudiantes durante exámenes en línea de manera efectiva y segura?

II. FUNDAMENTACIÓN TEÓRICA

2.1. ANTECEDENTES DE LA INVESTIGACIÓN

Villalobos Sánchez (2025), en su tesis de pregrado "Reconocimiento facial para prevenir suplantaciones en exámenes de admisión utilizando TensorFlow para la UNTRM", desarrollada en la Universidad Nacional Toribio Rodríguez de Mendoza de Amazonas (Perú), se planteó como objetivo determinar el grado de efectividad del sistema de reconocimiento facial en la prevención de suplantaciones en exámenes de admisión. El estudio utilizó un diseño cuantitativo cuasiexperimental evaluando como variable independiente el sistema de reconocimiento facial desarrollado mediante TensorFlow y metodología Scrum, y como variable dependiente la prevención de suplantaciones medida por precisión de detección, seguridad y rapidez operativa. La muestra consistió en 23 estudiantes de Ingeniería de Sistemas. Los resultados demostraron que el 100% de los participantes confirmó la detección correcta de intentos de suplantación, el 91.3% destacó la minimización de errores en autenticación, y el 87% consideró el proceso totalmente seguro. La investigación concluyó que el sistema basado en TensorFlow representa una solución altamente eficaz para prevenir fraudes por suplantación en procesos masivos de admisión universitaria sin requerir modificaciones en el hardware de los postulantes (p. 101).

Beraún Barrantes (2021), en su tesis de maestría "Sistema de reconocimiento facial en línea para prevenir la suplantación y el plagio en el examen de admisión virtual en la Universidad de Huánuco 2020", desarrollada en la Universidad de Huánuco (Perú), se propuso como objetivo implementar un sistema de reconocimiento facial en línea para prevenir la suplantación de identidad y el plagio durante exámenes de admisión virtual. La investigación utilizó un diseño de tipo aplicada con alcance descriptivo y enfoque cuantitativo no experimental transversal, enfocándose en dos variables primarias: como variable independiente, el sistema de reconocimiento facial en línea desarrollado en plataforma web con tecnologías .NET Core (backend) y Angular (frontend); y como variable dependiente, el control de casos de suplantación y plagio, medido por tasa de reconocimiento facial exitoso, precisión en identificación y reducción de necesidad de supervisión humana. El sistema "sin entrenamiento" fue

evaluado con 1,252 postulantes reales durante el examen de admisión virtual 2020, contexto crítico que requería garantizar identidad sin presencia física. Los resultados demostraron una tasa de reconocimiento exitoso del 90.42%, reconociendo correctamente a 1,132 postulantes de forma automatizada y requiriendo apoyo de controladores humanos solo para 120 postulantes (9.58%), sin registrarse ningún caso de falso positivo. La investigación concluyó que la implementación del sistema reduce significativamente la posibilidad de suplantación y plagio en exámenes masivos virtuales, permitiendo que los supervisores concentren su atención exclusivamente en casos problemáticos y demostrando escalabilidad efectiva sin necesidad de instalaciones previas de software en equipos de postulantes (p. 53).

Galindo Taype, Huaringa Gallardo & Samaniego Canales (2021) En su tesis de pregrado "Reconocimiento facial para la identificación de los alumnos en exámenes finales en la modalidad presencial de la Universidad Continental - Huancayo, 2021", desarrollada en la Universidad Continental (Perú), se plantearon como objetivo desarrollar un sistema de escritorio mediante la metodología Kanban para el reconocimiento facial y la identificación de alumnos en exámenes finales presenciales, abordando el problema de suplantación de identidad que se presentaba frecuentemente en asignaturas generales. El estudio utilizó un diseño de investigación tecnológica con enfoque de desarrollo tecnológico, evaluando como variable independiente el sistema de reconocimiento facial desarrollado con las librerías OpenCV y Face-Recognition en Python, y como variable dependiente la reducción de suplantación de identidad medida por precisión de detección y porcentaje de acierto. La muestra consistió en 5 estudiantes con 50 fotografías cada uno (250 imágenes totales), capturadas en diferentes escenarios: sin mascarilla-sin lentes, con mascarilla-sin lentes, y sin mascarilla-con lentes. Los resultados hallados mediante la aplicación de la matriz de confusión demostraron una precisión del 93% en el reconocimiento facial. La investigación concluyó que el sistema de reconocimiento facial es favorable para reducir la suplantación de identidad en exámenes presenciales (p. 143).

Coronel Teanga (2021), en su tesis de pregrado "Sistema Inteligente de identificación facial para registro de asistencia estudiantil en la Universidad Ecotec", desarrollada en la Universidad Tecnológica ECOTEC (Ecuador), propuso un sistema web que utiliza reconocimiento facial para automatizar la asistencia de estudiantes tanto en clases presenciales como virtuales, contexto surgido por la crisis sanitaria que evidenció

deficiencias en el control de asistencia online de la institución. El sistema fue desarrollado con Python y OpenCV, implementando los algoritmos Eigenfaces, basado en análisis de componentes principales (PCA) para representar rostros como combinaciones lineales de vectores característicos, y Haar Cascade para detección facial en tiempo real mediante características rectangulares simples. La metodología empleó un enfoque cualitativo con entrevistas para el levantamiento de información, determinando inconvenientes significativos en el registro manual de asistencia. El prototipo fue diseñado como servicio web y validado mediante trazabilidad de casos de prueba y requisitos funcionales, demostrando que registraba correctamente la identidad del estudiante de forma automática, eliminando errores como suplantaciones, registros incorrectos o incompletos del proceso manual. Se concluyó que la solución mejora significativamente el control de identidad y asistencia académica, proporcionando un mecanismo confiable y no intrusivo adaptado tanto a modalidad presencial como virtual.

Marrugo Cogollo y Castro Flórez (2022), en su tesis de grado "Sistema de reconocimiento facial para la gestión y el seguimiento de estudiantes ausentes (SEFAD)", desarrollada en la Universidad Piloto de Colombia, implementaron un sistema que mediante reconocimiento facial identifica automáticamente a estudiantes ausentes y facilita el seguimiento institucional para prevenir el absentismo y la deserción estudiantil. SEFAD fue diseñado como herramienta integral que no solo registra asistencia, sino que genera alertas y reportes sobre patrones de ausencia, permitiendo intervenciones proactivas de las autoridades educativas. El sistema fue evaluado con distintos tipos de cámaras bajo condiciones variables de iluminación, ángulos de captura y distancias para validar su robustez operacional en entornos reales, logrando detectar con éxito la identidad de los estudiantes y demostrando capacidad de adaptación sin comprometer su efectividad. Aunque no se reportaron métricas cuantitativas específicas como tasas de precisión o tiempos de respuesta, las pruebas cualitativas permitieron concluir que SEFAD es viable para mejorar el control de asistencia y prevenir el absentismo mediante identificación temprana de patrones de ausencia y generación de alertas automáticas. Los autores destacaron que la implementación representa un avance en la automatización de procesos administrativos educativos, recomendando considerar aspectos éticos y de protección de datos biométricos en futuras implementaciones.

2.2. MARCO TEÓRICO

2.2.1. Revisión sistemática de la literatura basado en método PRISMA enfocada en herramientas tecnológicas de reconocimiento facial

Se llevó a cabo una revisión sistemática siguiendo las directrices PRISMA, enfocada en estudios entre 2020 y 2025 sobre el uso de reconocimiento facial en entornos educativos universitarios.

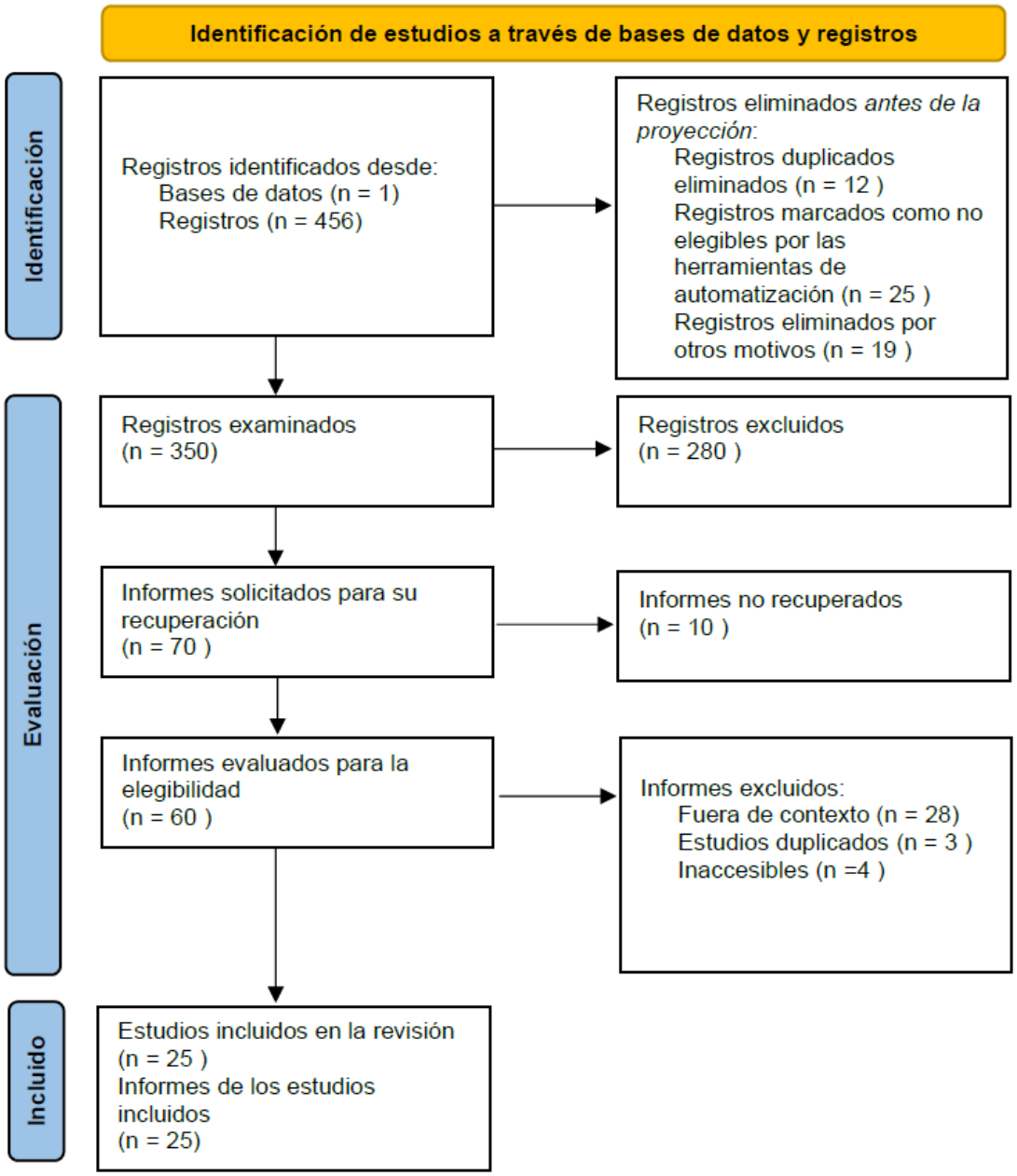


Figura 1. Diagrama PRISMA – Revisión de herramientas tecnológicas de reconocimiento facial

2.2.2. Herramientas tecnológicas utilizadas en reconocimiento facial

2.2.2.1. Algoritmos de reconocimiento facial

- FaceNet (Red neuronal profunda para reconocimiento facial): FaceNet es una arquitectura de red neuronal convolucional desarrollada por Google que genera embeddings de 128 dimensiones para cada rostro. Su fortaleza radica en el aprendizaje métrico, donde aprende a mapear imágenes faciales a un espacio euclidiano donde las distancias corresponden directamente a la similitud facial. Utiliza una función de pérdida triplet que optimiza simultáneamente la separación entre diferentes identidades y la agrupación de imágenes de la misma persona. Utilizado por GandhiSatra et al. (2021), mostró alta precisión en identificación. Al entrenarse con datos propios de estudiantes, logró mejorar su rendimiento en comparación con versiones genéricas. Fue clave para evitar falsos positivos, sobre todo en ambientes con baja iluminación.
- VGG-Face (Visual Geometry Group - Face Recognition): VGG-Face es una red neuronal convolucional basada en la arquitectura VGG-16, específicamente entrenada para reconocimiento facial con más de 2.6 millones de imágenes. Su arquitectura profunda de 16 capas permite extraer características faciales jerárquicas, desde bordes simples hasta patrones faciales complejos. Es particularmente efectivo para crear representaciones robustas que mantienen consistencia ante variaciones de iluminación y pose. Legarda y Loaiza (2022) lo emplearon para registrar asistencia automáticamente. Su precisión fue cercana al 100% en aulas controladas. Es un modelo eficaz, especialmente útil cuando se entrena con imágenes reales de estudiantes.
- OpenCV (LBPH, Haar) (Open Source Computer Vision Library - Local Binary Pattern Histograms, Haar Cascades): OpenCV es una biblioteca de visión computacional que incluye algoritmos clásicos de reconocimiento facial. LBPH (Local Binary Pattern Histograms) analiza la textura local de cada píxel comparándolo con sus vecinos, creando histogramas que caracterizan regiones faciales. Los clasificadores Haar utilizan características rectangulares simples para detectar objetos mediante cascadas de clasificadores débiles, siendo computacionalmente eficientes, aunque menos precisos que métodos modernos. Aparece en múltiples implementaciones latinoamericanas por ser

de código abierto y de fácil integración. Galindo et al. (2021) lo utilizaron para verificar identidad en exámenes presenciales en Perú. Su principal ventaja fue la accesibilidad y la posibilidad de adaptarlo sin licencias comerciales.

- MTCNN y RetinaFace (Multi-task Convolutional Neural Network y RetinaFace): MTCNN es una arquitectura de tres etapas (P-Net, R-Net, O-Net) que realiza detección facial, alineación de puntos clave y reconocimiento simultáneamente. RetinaFace mejora esta aproximación utilizando Feature Pyramid Networks y context modules para detectar rostros en múltiples escalas con alta precisión. Ambos son fundamentales en el pipeline de reconocimiento facial, ya que la calidad de la detección inicial determina significativamente el rendimiento del reconocimiento posterior. Fueron claves en la detección precisa del rostro. Utilizados como preprocesadores antes del reconocimiento, mejoraron los tiempos de respuesta y la calidad de la detección.
- YOLOv5 (You Only Look Once version 5 - Solo miras una vez versión 5): YOLOv5 es una arquitectura de detección de objetos en tiempo real que procesa toda la imagen en una sola pasada de red neuronal, dividiendo la imagen en una grilla y prediciendo simultáneamente cajas delimitadoras y probabilidades de clase para cada celda. Su eficiencia computacional y capacidad de detección multiclase lo hacen ideal para monitoreo de exámenes, donde debe identificar múltiples objetos prohibidos (dispositivos electrónicos, personas adicionales, materiales no autorizados) en tiempo real con alta precisión y baja latencia. En el trabajo de Singh et al. (2024), se aplicó para detectar objetos (como teléfonos o rostros adicionales) durante exámenes, ayudando a identificar trampas visuales.

2.2.2.2. Plataformas y frameworks

- Luxand: Este sistema comercial es utilizado en universidades de Asia Central. Luxand.cloud (Luxand.Cloud, 2024) proporciona soluciones de reconocimiento facial en varios sectores. Sakhipov et al. lo eligieron por su compatibilidad con políticas de privacidad como el GDPR, un aspecto crucial al manejar datos biométricos de estudiantes.
- Raspberry Pi + Python: La combinación de Raspberry Pi y Python ofrece una solución rentable y personalizable para tareas de reconocimiento facial

(Jaskolka et al., 2019). Un sistema basado en Raspberry Pi puede realizar verificación de identidad comparando la transmisión en vivo de la cámara de un estudiante con un perfil facial previamente registrado (Virata & Festijo, 2020). Python, con bibliotecas como OpenCV, proporciona las herramientas para implementar algoritmos de detección facial, extracción de características y reconocimiento (Ujang, 2019).

- APIs en la Nube: Las APIs de reconocimiento facial basadas en la nube, como las ofrecidas por Google Cloud Vision (Murtuzova, 2024) y Microsoft Azure Face API (AI-Powered Image & Face Recognition APIs in 2025, 2025), proporcionan facilidad de integración y escalabilidad. Estas APIs ofrecen funcionalidades como detección de rostros, reconocimiento, detección de emociones e identificación de atributos faciales.

2.2.3. Aplicaciones del reconocimiento facial en entornos universitarios

2.2.3.1. Verificación de identidad en exámenes

Durante la pandemia, muchas universidades adaptaron sus procesos de evaluación remota usando tecnologías biométricas. En este contexto, el reconocimiento facial jugó un papel clave.

- Galindo et al. (2021) diseñaron un sistema local que comparaba el rostro del estudiante con una base de datos institucional. Lograron una precisión del 93%, incluso con mascarillas.
- Sakhipov et al. (2025) desarrollaron un sistema más complejo: además de la verificación inicial, mantenía una vigilancia continua del rostro y detectaba si el estudiante se alejaba, era sustituido o había otra persona en pantalla.
- Singh et al. (2024) introdujeron el concepto de "vigilancia activa" con detección en tiempo real de elementos sospechosos (personas, móviles, cambios de voz), lo que ayudó a reducir drásticamente los intentos de suplantación.

En general, los estudios indican que estos sistemas son eficaces para prevenir suplantaciones, especialmente cuando combinan la verificación facial con mecanismos adicionales como análisis de movimiento, voz y patrones de comportamiento.

2.2.3.2. Control automatizado de asistencia

Los registros de asistencia fueron tradicionalmente manuales, propensos a errores o fraude. El reconocimiento facial ha permitido automatizar este proceso.

- Legarda & Loaiza (2022) desarrollaron un sistema capaz de registrar automáticamente la presencia de estudiantes en menos de 5 segundos por rostro. El sistema estaba basado en aprendizaje profundo y permitía identificar hasta 60 estudiantes con gran precisión.
- Heredia & Donoso (2024) aplicaron una solución similar en Ecuador, utilizando hardware económico. El sistema redujo los errores de registro y facilitó el reporte automatizado a las autoridades académicas.

2.2.3.3. Monitoreo de comportamiento y detección de trampas

Más allá de verificar quién rinde el examen, los sistemas comenzaron a analizar *cómo* lo hace. Esto incluye detectar si el estudiante está viendo a la cámara, si alguien más está presente o si se están usando dispositivos no permitidos.

- Ozdamli et al. (2022) diseñaron un sistema que combinaba detección facial, emociones y dirección de la mirada. Este sistema logró detectar comportamientos sospechosos con más del 95% de certeza.
- Resha et al. (2023) incorporaron reconocimiento de voz para saber si el estudiante hablaba con otra persona durante la evaluación. Las alertas automáticas ayudaron al docente a revisar situaciones específicas.

Este tipo de vigilancia ha generado debates éticos, pero los estudios coinciden en que mejora la transparencia y disuade el fraude.

2.2.3.4. Control de acceso a instalaciones

En universidades donde el control de acceso es necesario (por seguridad, pandemia o restricciones), el reconocimiento facial ha reemplazado el uso de tarjetas magnéticas.

- Wang et al. (2022) reportaron que los estudiantes se sentían más seguros y conectados con la institución cuando usaban su rostro como "llave" de acceso, siempre que se informara adecuadamente y se protegieran los datos personales.

La siguiente tabla presenta una síntesis comparativa de las principales tecnologías.

Tabla 1. Resultados PRISMA 1

Tecnología framework	Aplicación principal	Hallazgos clave	Autores y estudios representativos
OpenCV (Haar, LBPH, Dlib)	Verificación de identidad, asistencia, vigilancia	Fácil de implementar; alta precisión en entornos controlados; útil con hardware económico	Galindo et al. (2021), Heredia & Donoso (2024), Resha et al. (2023), Shrivastava et al. (2020)
FaceNet	Asistencia automatizada, reconocimiento personalizado	Mejora de precisión al personalizar el modelo con fotos de estudiantes; útil en aulas con muchos alumnos	GandhiSatra et al. (2021), Legarda & Loaiza (2022)
VGG-Face	Control de asistencia automatizado	Alta precisión (~99%); bien soportado en frameworks como Keras	Legarda & Loaiza (2022)
YOLOv5 / YOLOv3	Detección de trampas (móviles, personas, objetos)	Permite monitoreo del entorno; útil para reforzar exámenes seguros en línea	Singh et al. (2024), Jia & He (2022)
MTCNN / RetinaFace	Detección de rostros (preprocesamiento)	Precisión alta en detección rápida; mejora el reconocimiento posterior	Legarda & Loaiza (2022), Shrivastava et al. (2020)
Luxand Face SDK / API	Verificación de identidad, vigilancia continua	API comercial con alta precisión y cumplimiento de privacidad (GDPR)	Sakhipov et al. (2025)
ResNet (CNN personalizada)	Monitoreo facial, análisis de emociones	Reconocimiento emocional + atención visual; +95% de acierto en pruebas	Ozdamli et al. (2022), Jia & He (2022)
Speech-to-text / Recon. Voz	Verificación por voz, detección de conversación	Alta eficacia en combinación con video; detecta interacciones sospechosas durante exámenes	Resha et al. (2023), Masud et al. (2021)
Raspberry Pi + Python	Registro presencial de asistencia	Solución económica para instituciones con pocos recursos; confiable con buen entrenamiento facial	Heredia & Donoso (2024)
Google Cloud / AWS Rekognition	Verificación facial en plataformas online	Fácil integración, pero con riesgos de privacidad y dependencia comercial	Jia & He (2022) &
Aprendizaje Profundo – Keras / PyTorch / TensorFlow	Asistencia, reconocimiento continuo	Uso combinado con modelos como FaceNet o VGG-Face en entornos productivos	Legarda & Loaiza (2022), GandhiSatra et al. (2021), Jia & He (2022)
Sistemas personalizados / híbridos	Proctoring con IA, monitoreo remoto	Combinan IA, visión, micrófono y análisis estadístico para una vigilancia más robusta	Sahu & Kumar (2025), Güney et al. (2021), Baser et al. (2022)

2.2.4. Revisión sistemática de la literatura basado en método PRISMA enfocada en desafíos y limitaciones en el desarrollo del sistema de reconocimiento facial

Se llevó a cabo una revisión sistemática siguiendo las directrices PRISMA, enfocada en estudios entre 2020 y 2025 sobre las limitaciones y desafíos en el desarrollo de un sistema de reconocimiento facial en el entorno educativo.

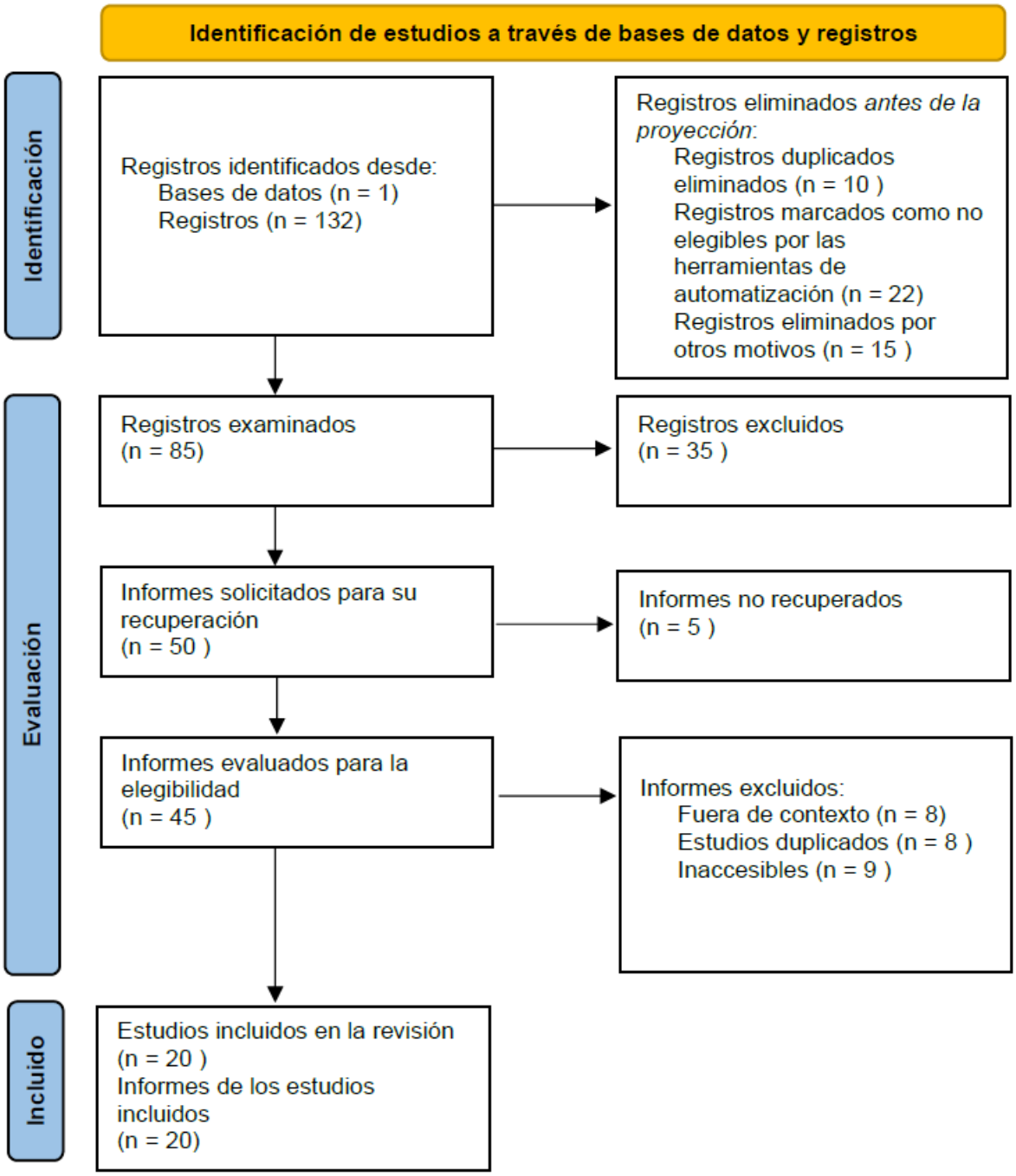


Figura 2. Diagrama PRISMA – Desafíos y limitaciones del reconocimiento facial

2.2.5. Desafíos técnicos

2.2.5.1. Imprecisión bajo condiciones reales

Los algoritmos de reconocimiento facial muestran un rendimiento significativamente inferior cuando se aplican en entornos domésticos con recursos tecnológicos limitados. En evaluaciones reales, factores como la baja resolución de las cámaras, la mala iluminación ambiental o la inestabilidad de la conexión a Internet provocan fallos de verificación y errores de autenticación (Aznarte et al., 2022; Guerrero, 2021). De hecho, un estudio realizado por Yamada et al. (2023) en una universidad pública de Brasil mostró que un 27% de los estudiantes no logró validar su identidad en el primer intento debido a problemas de imagen y conectividad.

Estos errores no solo afectan la experiencia del usuario, sino que también comprometen la equidad del proceso, ya que penalizan a quienes no disponen de equipos de alta gama o condiciones ambientales óptimas. Esto plantea dudas sobre la idoneidad de estos sistemas para contextos de educación superior masiva y diversa, como los de América Latina.

2.2.5.2. Sesgos algorítmicos

Una de las preocupaciones más documentadas en la literatura es la existencia de sesgos algorítmicos. Los sistemas de reconocimiento facial entrenados con bases de datos no representativas tienden a cometer más errores con mujeres, personas racializadas y usuarios con expresiones de género no normativas (Cashon et al., 2022; Manas, 2021). Buolamwini y Gebru (2018) ya habían advertido que estos sesgos pueden alcanzar tasas de error del 35% en rostros femeninos de piel oscura, frente a menos del 1% en varones blancos.

En contextos educativos, estos errores generan exclusiones injustas o falsas alarmas que pueden derivar en penalizaciones indebidas. Tal fue el caso reportado en la Universidad Estatal de Michigan (EE.UU.), donde estudiantes afroamericanos denunciaron repetidas fallas del sistema Examity para reconocer sus rostros, lo que retrasó sus exámenes y afectó su rendimiento académico (Rodríguez & Miller, 2022).

2.2.5.3. Alcance específico del reconocimiento facial

El reconocimiento facial está diseñado para verificar la identidad del estudiante,

no para detectar todas las formas de conducta deshonesta. Como señalan Bergmans et al. (2021), sistemas como Proctorio no detectaron casos de trampa que

involucraban dispositivos no visibles. Sin embargo, esto no representa una falla del reconocimiento facial en sí, sino una confirmación de que cada tecnología tiene un propósito específico dentro de un sistema de supervisión más amplio (Rivera & Soto, 2023). La verificación de identidad constituye una capa fundamental de seguridad que debe complementarse con otras estrategias de evaluación.

2.2.5.4. Problemas de accesibilidad

Numerosos informes señalan que los sistemas de reconocimiento facial no están diseñados para ser inclusivos con personas con discapacidad. Estudiantes con parálisis facial, movimientos involuntarios, visión reducida o neurodivergencias pueden tener dificultades para mantener el rostro dentro del encuadre, seguir instrucciones visuales o responder en tiempo limitado. Aznarte et al. (2022) alertan sobre la exclusión de estudiantes con trastornos del espectro autista o síndrome de Tourette, quienes pueden ser erróneamente clasificados como "sospechosos" por su comportamiento.

Organizaciones como EDUCAUSE (2022) han exigido a los desarrolladores implementar medidas de accesibilidad universal en sus plataformas de verificación biométrica.

2.2.6. Desafíos éticos

2.2.6.1. Privacidad de datos biométricos

El uso del rostro como dato biométrico implica una recolección masiva de información sensible. Guerrero (2021) y la Agencia Española de Protección de Datos (AEPD, 2025) señalan que el tratamiento de datos faciales con fines de vigilancia académica debe cumplir con los principios de necesidad, proporcionalidad y minimización. Sin embargo, múltiples investigaciones denuncian que estas tecnologías se implementan sin auditorías independientes, sin transparencia sobre el almacenamiento de datos y, en algunos casos, sin posibilidad de revocar el consentimiento.

Moor et al. (2023) advierten que el uso de estos sistemas en plataformas como Respondus o Honorlock ha derivado en fugas de datos y reventa de información biométrica a terceros con fines comerciales, lo que compromete gravemente la privacidad de los usuarios.

2.2.6.2. Consentimiento forzado

La validez del consentimiento otorgado por los estudiantes para el uso de tecnologías de reconocimiento facial ha sido cuestionada en múltiples foros. Según Guerrero (2021), cuando la participación en exámenes está condicionada a la aceptación de este tipo de vigilancia, el consentimiento deja de ser libre, ya que los estudiantes carecen de alternativas viables. Esta situación fue expuesta en la Universidad de Chile (2021), donde estudiantes de ingeniería protestaron públicamente porque no podían rendir sus pruebas si no activaban la cámara y el reconocimiento facial.

2.2.6.3. Vigilancia constante y efectos psicológicos

La literatura reporta que la sensación de ser monitoreado constantemente produce altos niveles de estrés, incomodidad y desconfianza. Manas (2021) afirma que el proctoring automatizado contribuye a un "clima de sospecha institucional" que rompe la relación pedagógica. En un estudio realizado en la Universidad de Columbia Británica, Canadá, el 71% de los encuestados manifestó sentirse más nervioso durante exámenes con vigilancia por IA que en pruebas presenciales tradicionales (Han & Hong, 2022).

2.2.7. Desafíos sociales

2.2.7.1. Brecha tecnológica y desigualdad de acceso

Las condiciones socioeconómicas influyen directamente en la capacidad del estudiante para cumplir con los requerimientos técnicos del reconocimiento facial. En países como México, Perú o Bolivia, donde el acceso a dispositivos modernos y conexión estable aún es limitado, esta tecnología ha generado una barrera adicional de acceso (Guerrero, 2021; Fernández et al., 2023).

En universidades públicas argentinas, docentes reportaron que hasta un 30% de sus alumnos no podía acceder a los sistemas de proctoring debido a limitaciones tecnológicas, lo que obligó a migrar a métodos alternativos como la evaluación asincrónica (Rivarola & Ortega, 2022).

2.2.7.2. Rechazo y percepción de injusticia

En 2020, más de 60.000 estudiantes firmaron peticiones en contra del uso de herramientas como ProctorU, Examity y Honorlock en universidades estadounidenses como UCLA, Harvard y MIT, alegando que violaban sus derechos fundamentales. En América Latina, la Universidad de Antioquia (Colombia) y la Universidad Nacional

Autónoma de México también enfrentaron protestas similares, con campañas como "*Mi rostro no es tu control*" (Patiño, 2021).

Estos movimientos denuncian que los sistemas castigan movimientos naturales, expresiones faciales o pausas prolongadas, interpretándolos como signos de trampa sin evidencia sólida.

2.2.7.3. Estigmatización de minorías

Estudiantes trans, no binarios o con apariencias no normativas han reportado errores de reconocimiento y bloqueos injustificados al intentar rendir sus exámenes. En un caso documentado en la Universidad de Toronto, una estudiante trans fue rechazada por el sistema Proctortrack al no coincidir su imagen facial actual con la registrada al inicio del curso (Lee & Shah, 2022). Esto evidencia que los algoritmos actuales están diseñados con supuestos binarios y no contemplan la diversidad de identidades.

2.2.8. Desafíos legales

2.2.8.1. Ausencia de legislación específica

Si bien muchos países cuentan con leyes de protección de datos, pocas han desarrollado normativas específicas sobre reconocimiento facial en educación. Esto ha creado un vacío legal que ha sido explotado por empresas proveedoras de software. La AEPD (2025) advierte que el tratamiento de datos biométricos con fines académicos debe cumplir con estándares más exigentes que los actualmente observados.

En América Latina, solo Brasil ha emitido lineamientos explícitos a través de la Ley General de Protección de Datos Personales (LGPD), que considera los datos faciales como categoría sensible y exige justificación legal robusta para su uso (Silva & Cardoso, 2023).

2.2.8.2. Proporcionalidad y alternativas

La jurisprudencia europea y algunas universidades estadounidenses han señalado que existen medios menos invasivos para garantizar la integridad académica, como las pruebas abiertas, orales, proyectos grupales o rúbricas de desempeño. Según Aznarte et al. (2022), aplicar reconocimiento facial en una evaluación domiciliaria es desproporcionado si se compara con la relevancia del objetivo (verificar identidad) frente al riesgo para los derechos del estudiante.

En la siguiente tabla se presentan los principales desafíos y limitaciones identificados en la literatura científica revisada sobre el uso de sistemas de reconocimiento facial para la verificación de identidad durante exámenes virtuales.

Tabla 2. Principales desafíos y limitaciones del uso de reconocimiento facial

Desafío / Limitación	Clasificación	Autor(es)
Imprecisión técnica (errores con cámaras, luz, hardware limitado)	Técnico	Aznarte et al. (2022), Guerrero (2021), Cashon et al. (2022)
Sesgo algorítmico (por raza, género, discapacidad)	Técnico / Ético	Cashon et al. (2022), Manas (2021), AEPD (2025)
No evita otras formas de trampa (libros, ayuda externa)	Técnico / Funcional	Bergmans et al. (2021), Aznarte et al. (2022)
Vulneración a la privacidad por datos biométricos	Ético / Legal	Guerrero (2021), AEPD (2025), Aznarte et al. (2022)
Falta de consentimiento libre e informado	Ético / Legal	Guerrero (2021), Aznarte et al. (2022)
Vigilancia constante genera ansiedad y malestar	Ético / Social	Manas (2021), Aznarte et al. (2022), Cashon et al. (2022)
Desigualdad tecnológica y brecha digital	Social / Técnico	Guerrero (2021), Aznarte et al. (2022)
Rechazo social y percepción de vigilancia excesiva	Social	Manas (2021), Guerrero (2021)
Falta de marco legal específico	Legal	AEPD (2025), Aznarte et al. (2022)
Dificultades con accesibilidad para estudiantes con discapacidad	Ético / Técnico	Aznarte et al. (2022)

2.2.9. Revisión sistemática de la literatura basada en el método PRISMA enfocada en la integración del reconocimiento facial en entornos virtuales de aprendizaje

Se realizó una revisión sistemática de la literatura siguiendo el protocolo PRISMA, centrada en estudios publicados entre 2020 y 2025 que abordan la integración de sistemas de reconocimiento facial en entornos virtuales de aprendizaje.

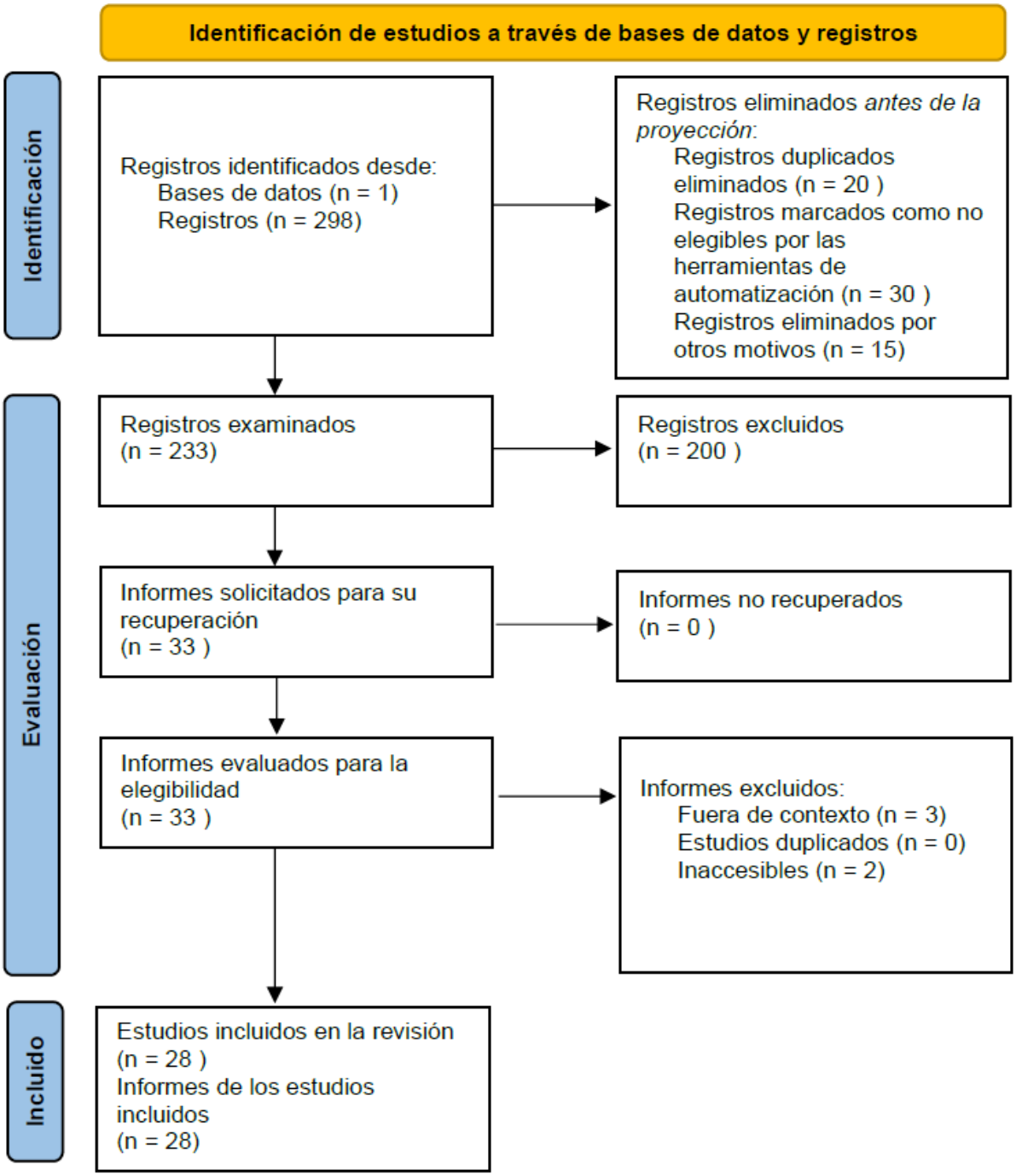


Figura 3. Diagrama PRISMA – Integración del reconocimiento facial en entornos virtuales de aprendizaje

2.2.9.1. Formas de integración tecnológica en LMS

La literatura reciente muestra que la integración del reconocimiento facial en EVA (típicamente LMS, Learning Management Systems) se logra principalmente de dos formas, mediante complementos (plugins) nativos en la propia plataforma, y a través de servicios externos que se acoplan vía API o mediante el estándar LTI.

En el primer caso, existen plugins diseñados específicamente para LMS como Moodle. Un ejemplo es Moodle Proctoring, que extiende la funcionalidad del cuestionario para capturar imágenes del estudiante durante el examen e identificarlas automáticamente mediante tecnologías biométricas (Shkodzinskyi et al., 2023). Este tipo de integración puede incluso conectarse a APIs externas de reconocimiento facial como Amazon Rekognition, permitiendo comparar en tiempo real la imagen capturada con la fotografía de registro del alumno y bloquear el acceso si no hay coincidencia.

Otra ventaja de este enfoque es que se mantiene dentro del ecosistema del LMS, aprovechando sus herramientas nativas sin necesidad de acceder a aplicaciones externas. Por ejemplo, Admira y Arnesia (2021) integraron el módulo Fullface Biometric en Moodle como regla de acceso a los cuestionarios: el estudiante debía tomarse una foto al momento del examen, que luego se comparaba con su imagen de perfil registrada. Si no coincidía, el sistema bloqueaba el acceso hasta lograr una verificación positiva.

De forma similar, Torres et al. (2024) desarrollaron un plugin institucional para la Universidad Continental (Perú), también sobre Moodle. Este sistema realizaba la autenticación en dos fases: primero antes de acceder al cuestionario, y luego monitoreaba continuamente al estudiante durante toda la prueba a través de la cámara web. El plugin utilizaba modelos de visión por computadora basados en face-api.js y Tiny YOLOv2, ejecutados localmente en el navegador, lo que reducía la carga en los servidores institucionales y evitaba enviar datos sensibles a terceros.

2.2.9.2 .Plataformas y herramientas utilizadas

Paralelamente al enfoque basado en plugins, varios estudios documentan el uso de servicios externos de proctoring con reconocimiento facial integrados a los EVA. Estas herramientas, como Proctortrack, Proctorio, Examity o SMOWL, actúan como plataformas paralelas: el estudiante accede al examen desde el LMS (Canvas, Moodle, Blackboard, Sakai, entre otros), pero la sesión de supervisión se gestiona desde el servicio de terceros, integrado a través de LTI o API (Shkodzinskyi et al., 2023).

Estas soluciones incluyen funcionalidades más amplias: además del reconocimiento facial inicial, ofrecen grabación de pantalla, monitoreo del comportamiento, detección de múltiples rostros en pantalla, bloqueo del navegador y alertas automáticas por movimientos considerados sospechosos.

Shkodzinskyi et al. (2023) señalan que estas plataformas comerciales logran una integración fluida con la mayoría de LMS institucionales. Sin embargo, al tratarse de soluciones de código cerrado, las instituciones educativas carecen de control sobre los algoritmos utilizados y sobre la gestión interna de los datos recolectados. Además, estas herramientas imponen condiciones técnicas exigentes (como buena iluminación o cámara de alta resolución), lo que puede generar errores en la autenticación e incluso bloqueos arbitrarios durante la sesión.

2.2.9.3. Resultados de eficacia en verificación

Los estudios revisados indican que la integración del reconocimiento facial puede mejorar significativamente la autenticación de estudiantes, especialmente frente a intentos de suplantación de identidad. En el caso del sistema desarrollado por Torres et al. (2024), se logró detectar automáticamente casos simulados en los que personas no autorizadas intentaron acceder a un cuestionario, gracias a la comparación biométrica con la imagen de registro.

De forma complementaria, Admira y Arnesia (2021) reportaron que, tras implementar su sistema de autenticación facial, ningún estudiante no verificado pudo acceder al examen, eliminando efectivamente los casos de suplantación.

Sin embargo, también se identificaron limitaciones. En el estudio de Torres et al. (2024), el sistema presentó un 39% de error promedio en la métrica de similitud facial bajo condiciones reales, lo cual sugiere que factores como cambios de apariencia, iluminación deficiente o ángulos inadecuados pueden afectar la precisión del sistema.

Ante estas limitaciones, algunos desarrollos recientes han optado por enfoques más robustos. Potluri et al. (2023) propusieron el sistema Attentive, que combina verificación facial con detección de múltiples personas, análisis de rostro vivo (liveness detection) y seguimiento de movimientos de cabeza. Este sistema alcanzó un 87% de precisión en la identificación automática de comportamientos deshonestos durante pruebas experimentales.

2.2.9.4. Consideraciones de seguridad y privacidad

Una integración segura del reconocimiento facial no solo depende de la precisión técnica, sino de la protección efectiva de los datos biométricos y de los derechos de los estudiantes. Diversos autores (Guerrero, 2021; AEPD, 2025) han señalado que el uso

del rostro como dato biométrico debe cumplir con principios como proporcionalidad, necesidad y minimización.

Durante la pandemia, muchas universidades implementaron sistemas de reconocimiento facial sin cumplir con garantías mínimas, lo que derivó en denuncias por consentimiento forzado, falta de transparencia sobre el almacenamiento de datos y uso indebido de la información capturada. En la Universidad de Chile, por ejemplo, se documentaron casos en los que los estudiantes fueron obligados a aceptar la verificación facial sin posibilidad de rendir el examen por otros medios (Guerrero, 2021).

Asimismo, Moor et al. (2023) advirtieron sobre riesgos de fugas de datos biométricos en plataformas como Respondus y Honorlock, donde la información facial fue almacenada sin control institucional y, en algunos casos, reutilizada con fines comerciales.

A continuación, se presenta una síntesis de los estudios más relevantes identificados durante la revisión sistemática, los cuales abordan distintos enfoques de integración del reconocimiento facial en entornos virtuales de aprendizaje. La tabla incluye información sobre la plataforma utilizada, el tipo de herramienta o sistema implementado, la forma de integración tecnológica y los principales resultados obtenidos en relación con la verificación de identidad de los estudiantes durante exámenes en línea. Esta sistematización permite observar tendencias, fortalezas y limitaciones que contribuyen a responder la pregunta de investigación sobre cómo implementar esta tecnología de forma efectiva y segura.

Tabla 3. Estudios seleccionados sobre la integración del reconocimiento facial en entornos virtuales de aprendizaje (2020–2025)

Autor (Año)	Plataforma / LMS	Herramienta / Tipo de integración	Resultados principales
Torres et al. (2024)	Moodle (Univ. Continental)	Plugin institucional. Face-api.js + Tiny YOLOv2. Autenticación previa y monitoreo durante el examen.	Detectó intentos de suplantación. Error del 39% en condiciones reales.
Admira & Arnesia (2021)	Moodle (STMIK Jakarta)	Plugin Fullface Biometric. Comparación con foto de perfil.	Impidió ingreso a usuarios no verificados. Eliminó casos de suplantación.
Shkodzinskyi et al. (2023)	Moodle, Canvas, Blackboard	Análisis comparativo. Integración de Proctorio, Proctortrack, SMOWL vía LTI.	Herramientas comerciales bien integradas, pero menos auditables.
Potluri et al. (2023)	Plataforma propia (Attentive)	Sistema con múltiples capas: reconocimiento facial, *liveness*, tracking.	Precisión del 87% en detección de trampas simuladas.

III. METODOLOGÍA

3.1. ENFOQUE METODOLÓGICO

3.1.1. Enfoque

En el desarrollo de este proyecto se empleará un enfoque cuantitativo que permitirá recolectar y analizar datos numéricos de manera sistemática. Este enfoque facilitará la medición de variables como el nivel de conocimiento sobre reconocimiento facial, la percepción de efectividad del sistema, la frecuencia de casos de fraude académico, las preocupaciones principales de los usuarios, los contextos de aplicación preferidos y las soluciones ante posibles fallos técnicos del sistema.

3.1.2. Tipo de Investigación

3.1.2.1. Investigación documental

La investigación documental permitirá recopilar información especializada de fuentes académicas confiables tales como: artículos científicos indexados, libros especializados en biometría y seguridad informática, tesis doctorales y de maestría, revistas científicas del área de computer vision y aprendizaje profundo. Esta recopilación tiene como propósito fundamentar sólidamente el marco teórico, estableciendo las bases conceptuales sobre algoritmos de reconocimiento facial.

3.1.2.2. Investigación de campo

Se realizará investigación de campo para recolectar datos primarios directamente de estudiantes de la carrera de Computación de la Universidad Politécnica Estatal del Carchi. Esta aproximación permitirá obtener información sobre el nivel de conocimiento y familiaridad con sistemas de reconocimiento facial en contextos educativos.

3.1.2.3. Investigación descriptiva

Esta investigación permitirá analizar y describir las características técnicas del sistema de reconocimiento facial basado en aprendizaje profundo, identificar los requerimientos de hardware y software necesarios para su implementación, documentar los estándares de seguridad y privacidad requeridos para el manejo de

datos biométricos, y caracterizar tanto la problemática del fraude académico en exámenes virtuales como las percepciones y necesidades de los usuarios finales

3.2. IDEA A DEFENDER

Un sistema de reconocimiento facial basado en aprendizaje profundo mejorará la integridad académica en exámenes en línea al prevenir la suplantación de identidad mediante autenticación segura.

3.3. DEFINICIÓN Y OPERACIONALIZACIÓN DE LAS VARIABLES

3.3.1. Definición de variables

Variable Independiente: Sistema de reconocimiento facial

Tecnología biométrica basada en algoritmos de aprendizaje profundo que permite identificar y verificar la identidad de una persona mediante el análisis automático de características faciales únicas, comparando el rostro capturado en tiempo real con una imagen de referencia previamente registrada en el sistema.

Variable Dependiente: Verificación de identidad en exámenes en línea

Proceso de autenticación que confirma que la persona que está realizando un examen virtual es realmente el estudiante matriculado en el curso, garantizando la integridad académica y previniendo la suplantación de identidad o fraude académico.

3.3.2. Operacionalización de las variables

Tabla 4. Operacionalización de variables

Variable	Dimensión	Indicadores	Técnicas	Instrumento
Variable Independiente:				
Sistema de reconocimiento facial	Conocimiento y familiaridad	- Nivel de conocimiento sobre reconocimiento facial - Grado de familiaridad con la tecnología - Comprensión del funcionamiento del sistema	Encuesta	Cuestionario estructurado
	Efectividad percibida	- Percepción de efectividad para prevenir suplantación - Comparación con otros métodos de autenticación - Confianza en la tecnología	Encuesta	Cuestionario estructurado
	Preocupaciones de implementación	- Temor a fallos de reconocimiento	Encuesta	Cuestionario estructurado

	Contextos de aplicación	- Percepción de invasión de la privacidad - Preocupación por pérdida de tiempo - Inquietudes sobre privacidad - Tipos de exámenes donde debe implementarse - Frecuencia de uso recomendada - Priorización de escenarios	Encuesta	Cuestionario estructurado
	Protocolos de contingencia	- Soluciones preferidas ante fallos técnicos - Opciones de verificación - Nivel de aceptación de respaldos	Encuesta	Cuestionario estructurado
Variable Dependiente:				
Verificación de identidad en exámenes en línea	Fraude académico	- Conocimiento de casos de suplantación de identidad - Frecuencia de casos detectados - Prevalencia del problema	Encuesta	Cuestionario estructurado
	Métodos de autenticación	- Preferencia por métodos de verificación - Comparación entre sistemas tradicionales y biométricos - Aceptación de métodos combinados	Encuesta	Cuestionario estructurado
	Seguridad e integridad académica	- Percepción de seguridad del sistema - Confianza en la prevención de fraude - Efectividad para garantizar integridad	Encuesta	Cuestionario estructurado

3.4. MÉTODOS UTILIZADOS

3.4.1. Método Analítico-Sintético

Se aplicará el método analítico-sintético para descomponer y examinar de manera detallada cada componente del sistema de reconocimiento facial basado en aprendizaje profundo, y posteriormente integrarlos en una solución coherente y funcional. El análisis permitirá examinar por separado los algoritmos de detección facial, los procesos de extracción de características mediante redes neuronales convolucionales, los mecanismos de comparación de vectores de embeddings, y los protocolos de seguridad para el manejo de datos biométricos. La síntesis facilitará la unificación de estos módulos en un sistema completo de autenticación biométrica, además de permitir sintetizar los hallazgos de la investigación documental con los resultados de la investigación de campo para establecer conclusiones generales

sobre la viabilidad, efectividad y aceptación del sistema propuesto (Rodríguez & Pérez, 2021).

3.4.2. Método Inductivo

Se utilizará el método inductivo para establecer conclusiones generales a partir de la observación de casos particulares. Mediante el análisis de los datos recopilados en la encuesta aplicada a estudiantes de la carrera de Computación, se extraerán patrones de comportamiento, tendencias de percepción y niveles de aceptación que permitirán generar conclusiones aplicables al contexto más amplio de la educación superior virtual. Este método facilitará la identificación de regularidades en las preocupaciones de los usuarios, las preferencias de implementación y la percepción de efectividad del sistema propuesto, partiendo de lo particular hacia lo general (Palmett, 2020).

3.4.3. Método Deductivo

El método deductivo se aplicará partiendo de los principios teóricos establecidos sobre sistemas biométricos, técnicas de aprendizaje profundo y seguridad informática para llegar a conclusiones específicas aplicables al caso particular del reconocimiento facial en exámenes en línea. Este método permitirá validar si las teorías generales sobre autenticación biométrica y prevención de fraude académico se cumplen en el contexto específico de la Universidad Politécnica Estatal del Carchi, deduciendo soluciones concretas basadas en fundamentos científicos previamente validados (Rodríguez & Pérez, 2021).

3.4.4. Método Descriptivo

Se empleará el método descriptivo para caracterizar de manera detallada tanto el sistema de reconocimiento facial propuesto como las percepciones y actitudes de los estudiantes respecto a su implementación. Este método permitirá describir las características técnicas del sistema basado en aprendizaje profundo, documentar los requerimientos tecnológicos, especificar los protocolos de seguridad, y detallar los resultados obtenidos en la encuesta mediante estadística descriptiva (frecuencias, porcentajes, promedios). La descripción sistemática de variables como nivel de conocimiento, percepción de efectividad, preocupaciones principales y contextos de aplicación proporcionará una comprensión completa del fenómeno estudiado (Guevara et al., 2020).

3.5. ANÁLISIS ESTADÍSTICO

3.5.1. Población

La población de estudio está conformada por los 356 estudiantes matriculados en la carrera de Ingeniería en Computación de la Universidad Politécnica Estatal del Carchi, ubicada en la ciudad de Tulcán, provincia del Carchi, Ecuador. Esta población fue seleccionada debido a que los estudiantes de esta carrera poseen conocimientos técnicos que les permiten comprender mejor las implicaciones tecnológicas del reconocimiento facial y pueden proporcionar perspectivas valiosas sobre la viabilidad de implementación de este tipo de sistemas en el contexto educativo.

3.5.2. Muestra

Para determinar el tamaño de la muestra se utilizó la fórmula de muestreo para poblaciones finitas, considerando un nivel de confianza del 95% y un margen de error del 5%:

$$n = \frac{N \cdot Z^2 \cdot p \cdot q}{(N-1) \cdot e^2 + Z^2 \cdot p \cdot q}$$

Donde:

- N = 356 (tamaño de la población)
- Z = 1.96 (nivel de confianza del 95%)
- p = 0.5 (proporción esperada)
- q = 0.5 (1 - p)
- e = 0.05 (margen de error del 5%)

$$n = \frac{356 \cdot 1.96^2 \cdot 0.5 \cdot 0.5}{(356-1) \cdot 0.05^2 + 1.96^2 \cdot 0.5 \cdot 0.5}$$

$$n = 186.29$$

Aplicando la fórmula, se obtuvo una muestra mínima requerida de 186 estudiantes. Sin embargo, la muestra final estuvo constituida por 213 estudiantes que completaron voluntariamente la encuesta sobre reconocimiento facial en exámenes en línea, lo que representa el 59.8% de la población total y supera ampliamente el tamaño mínimo requerido, garantizando así la representatividad y confiabilidad de los resultados obtenidos.

3.5.3. Instrumentos de investigación

3.5.3.1. Encuesta estructurada

El instrumento utilizado para la recolección de datos fue una encuesta estructurada diseñada específicamente para medir las percepciones, conocimientos y actitudes de los estudiantes respecto al uso de reconocimiento facial como método de verificación de identidad en exámenes en línea. La encuesta es uno de los instrumentos más utilizados en la investigación cuantitativa debido a que permite recopilar información de manera sistemática, estandarizada y eficiente sobre las opiniones, actitudes y comportamientos de grandes grupos de personas en un tiempo relativamente corto (Guevara et al., 2020).

IV. RESULTADOS Y DISCUSIÓN

4.1. RESULTADOS

4.1.1. Análisis e interpretación de resultados

Los datos recopilados a través de la encuesta estructurada fueron procesados y analizados mediante técnicas de estadística descriptiva, utilizando herramientas digitales para la tabulación y generación de gráficos. El análisis estadístico descriptivo permite organizar, resumir y presentar los datos de manera que faciliten su interpretación y comprensión, proporcionando una visión clara de las características de la muestra estudiada (Guevara et al., 2020).

1. ¿Conoce o ha escuchado sobre el uso del reconocimiento facial para verificar la identidad de los estudiantes durante exámenes en línea?

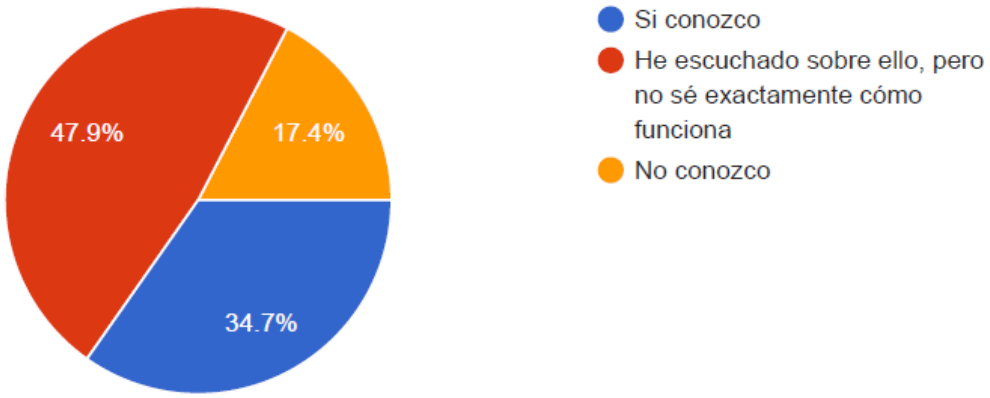


Figura 4. Conocimiento sobre reconocimiento facial en exámenes en línea

Tabla 5. Conocimiento sobre reconocimiento facial en exámenes

Opciones	Respuesta	Porcentaje
Sí conozco	102	47.9%
He escuchado sobre ello, pero no sé exactamente cómo funciona	74	34.7%
No conozco	37	17.4%
Total	213	100%

Análisis:

Para determinar el nivel general de conocimiento de los estudiantes sobre la tecnología de reconocimiento facial, se calculó el Índice de Conocimiento (IC)

mediante una escala ponderada donde: conocimiento pleno = 3 puntos, conocimiento parcial = 2 puntos, y sin conocimiento = 1 punto.

Fórmula del Índice de Conocimiento (IC):

$$IC = \frac{(P_1 \times V_1) + (P_2 \times V_2) + (P_3 \times V_3)}{V_{MAX}}$$

Donde:

P_1 = Porcentaje de "Si conozco" = 47.9%

P_2 = Porcentaje de "He escuchado, pero no sé" = 34.7%

P_3 = Porcentaje de "No conozco" = 17.4%

V_1 = Valor para conocimiento pleno = 3

V_2 = Valor para conocimiento parcial = 2

V_3 = Valor para sin conocimiento = 1

V_{MAX} = Valor máximo posible = 3

Calculo:

$$IC = \frac{(47.9 \times 3) + (34.7 \times 2) + (17.4 \times 1)}{3}$$

$$IC = \frac{143.7 + 69.4 + 17.4}{3} = \frac{230.5}{3} = 76.83\%$$

El Índice de Conocimiento de 76.83% refleja un nivel de familiaridad moderado-alto con la tecnología de reconocimiento facial. Este resultado indica que la mayoría de los estudiantes posee al menos un conocimiento básico del sistema, lo que facilita su adopción y sugiere que las preocupaciones estarán más enfocadas en aspectos prácticos de implementación que en el concepto mismo de la tecnología.

Pregunta 2: ¿Conoce casos donde un estudiante haya permitido que alguien más tome su examen virtual?

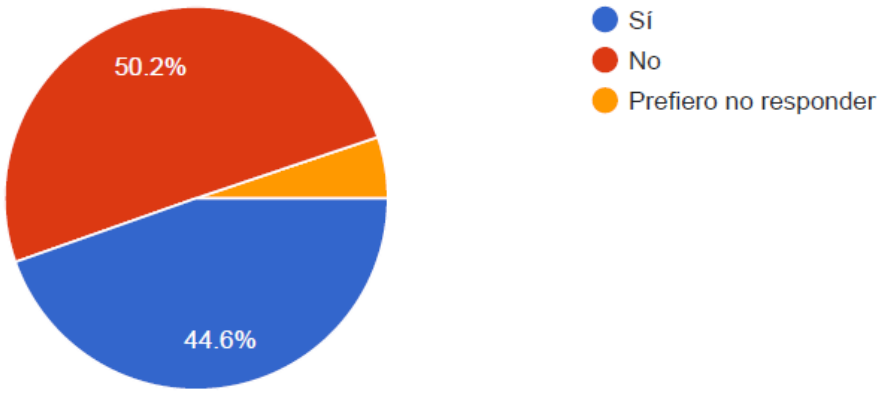


Figura 5. Conocimiento de casos de suplantación de identidad en exámenes virtuales

Tabla 6. Conocimiento de casos de suplantación de identidad

Opciones	Respuesta	Porcentaje
Sí	95	44.6%
No	107	50.2
Prefiero no responder	11	5.2%
Total	213	100%

Análisis:

Para evaluar la magnitud del problema de fraude académico por suplantación de identidad, se calculó el Índice de Prevalencia del Fraude (IPF), el cual determina qué proporción de estudiantes que pueden confirmar o negar casos ha sido testigo directo de esta problemática.

Fórmula del Índice de Prevalencia del Fraude (IPF):

$$IPF = \frac{P_{si}}{P_{si} + P_{no}}$$

Donde:

P_{si} = Porcentaje que respondió "Sí" = 44.6%

P_{no} = Porcentaje que respondió "No" = 50.2%

Calculo:

$$IPF = \frac{44.6}{44.6 + 50.2} = \frac{44.6}{94.8} = 0.471 = 47.1\%$$

El Índice de Prevalencia del Fraude de 47.1% indica que casi la mitad de los estudiantes que pueden confirmar o negar ha sido testigo de casos de suplantación de identidad. Esta cifra demuestra que el fraude académico no es un evento aislado sino un problema presente en la comunidad educativa. Adicionalmente, quienes prefirieron no responder podrían sugerir que el problema es mayor al reportado.

Pregunta 2.1: Si respondió afirmativamente en la pregunta anterior, ¿con qué frecuencia ha tenido conocimiento de estos casos?

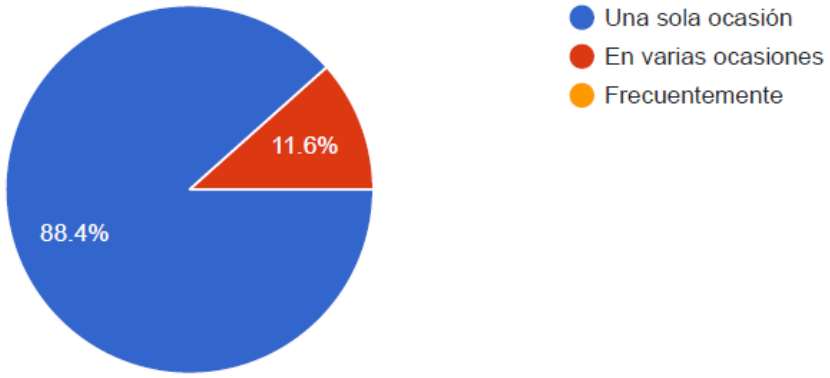


Figura 6. Frecuencia de casos conocidos de suplantación de identidad

Tabla 7. Frecuencia de casos conocidos de suplantación de identidad

Opciones	Respuestas	Porcentaje
Una sola ocasión	84	88.4%
En varias ocasiones	11	11.6%
Frecuentemente	0	0%
Total	95	100%

Análisis:

Para cuantificar la gravedad del problema del fraude académico, combinando prevalencia y recurrencia, se calcularon el Índice de Recurrencia del Fraude (IRF) y el Índice de Gravedad del Fraude (IGF). Se asignaron valores: una ocasión = 1 punto, varias ocasiones = 2 puntos, y frecuentemente = 3 puntos.

Fórmula del Índice de Recurrencia del Fraude (IRF):

$$IRF = \sum_{i=1}^{n} (P_i \times F_i)$$

Donde:

P_i = Porcentaje de cada categoría

F_i = Valor de frecuencia (1, 2 o 3)

Cálculo del IRF:

$$IRF = (0.884 \times 1) + (0.116 \times 2) + (0 \times 3)$$

$$IRF = 0.884 + 0.232 + 0 = 1.116$$

Fórmula del Índice de Gravedad del Fraude (IGF):

$$IGF = IPF \times IRF$$

Cálculo del IGF:

$$IGF = 0.471 \times 1.116 = 0.526$$

El Índice de Gravedad del Fraude de 0.526 en una escala de 0 a 3 posiciona el fraude académico como un problema moderado pero presente. Aunque la mayoría ha presenciado casos en una sola ocasión, la existencia de quienes han visto múltiples incidentes indica que el fenómeno tiene presencia en el entorno académico.

Pregunta 3: ¿Qué tan efectivo considera que sería el reconocimiento facial para prevenir la suplantación de identidad en exámenes en línea?

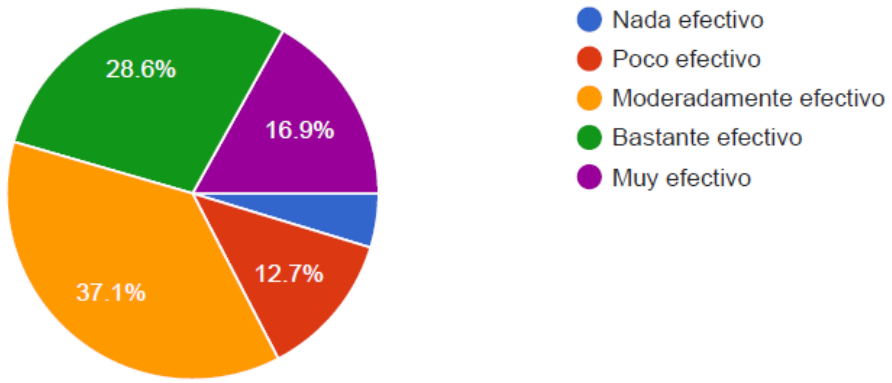


Figura 7. Percepción de efectividad del reconocimiento facial

Tabla 8. Percepción de efectividad del reconocimiento facial

Opciones	Respuestas	Porcentaje
Nada efectivo	9	2%
Poco efectivo	27	12.7%
Moderadamente efectivo	61	28.6%
Bastante efectivo	79	37.1%
Muy efectivo	36	16.9%
Total	213	100%

Análisis:

Para medir el nivel de confianza en la tecnología de reconocimiento facial, se calculó el Índice de Confianza en la Tecnología (ICT) mediante una escala de Likert

ponderada: nada efectivo = 1, poco efectivo = 2, moderadamente efectivo = 3, bastante efectivo = 4, y muy efectivo = 5.

Fórmula del Índice de Confianza en la Tecnología (ICT):

$$ICT = \frac{\sum_{i=1}^{n}(V_i \times P_i)}{100}$$

Donde:

V_i = Valor en escala (1 a 5)

P_i = Porcentaje de cada categoría

Cálculo:

$$ICT = \frac{(1 \times 4.7) + (2 \times 12.7) + (3 \times 28.6) + (4 \times 37.1) + (5 \times 16.9)}{100}$$

$$ICT = \frac{4.7 + 25.4 + 85.8 + 148.4 + 84.5}{100} = \frac{348.8}{100} = 3.49$$

El Índice de Confianza en la Tecnología de 3.49 sobre 5.0 (69.8%) indica una percepción mayoritariamente positiva hacia la efectividad del reconocimiento facial. La mayoría de las estudiantes tiene confianza en el sistema, mientras que solo una minoría muestra escepticismo. La opción "bastante efectivo" fue la más seleccionada, sugiriendo alta aceptación del sistema para cumplir su función de prevenir la suplantación de identidad en exámenes virtuales.

Pregunta 4: ¿Cuál de los siguientes métodos considera más efectivo para garantizar la seguridad y evitar el fraude académico en exámenes en línea?

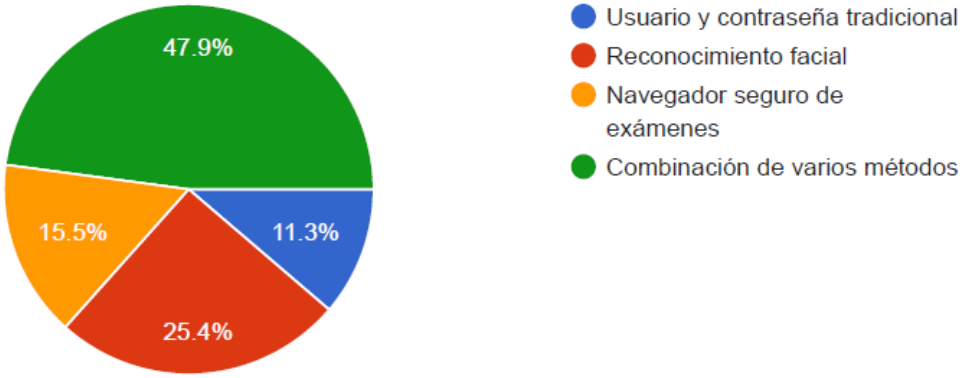


Figura 8. Métodos preferidos para garantizar seguridad en exámenes en línea

Tabla 9. Método preferido para garantizar seguridad en exámenes

Opciones	Respuestas	Porcentaje
Usuario y contraseña tradicional	24	11.3%
Reconocimiento facial	54	25.4%
Navegador seguro de exámenes	33	15.5%
Combinación de varios métodos	102	47.9%
Total	213	100%

Análisis:

Para analizar la coherencia entre la confianza en la efectividad del reconocimiento facial y su preferencia como método de seguridad, se calculó el Índice de Coherencia de Preferencias (ICP).

Fórmula del Índice de Coherencia de Preferencias (ICP):

$$ICP = \frac{P_{RF}}{P_{efectivo}}$$

Donde:

P_{RF} = Porcentaje que prefiere RF = 25.4%

$P_{efectivo}$ = Porcentaje que considera RF efectivo = 54.0%

Cálculo:

$$ICP = \frac{25.4}{54.0} = 0.470 = 47\%$$

Brecha de Coherencia:

$$BC = 100\% - 47\% = 53\%$$

El Índice de Coherencia de 47.0% revela una brecha significativa del 53.0% entre percepción de efectividad y preferencia real. Aunque muchos estudiantes

consideran el RF efectivo, solo una porción menor lo prefiere como método único. Esta discrepancia indica una visión pragmática que favorece sistemas de seguridad multicapa. La combinación de varios métodos obtiene la mayor preferencia, sugiriendo que un enfoque híbrido tendría mayor aceptación que sistemas únicos. La baja preferencia por métodos tradicionales evidencia que los estudiantes reconocen sus limitaciones y están preparados para adoptar soluciones tecnológicas más robustas.

Pregunta 5: ¿Qué te preocuparía más si tu universidad implementara reconocimiento facial en los exámenes?

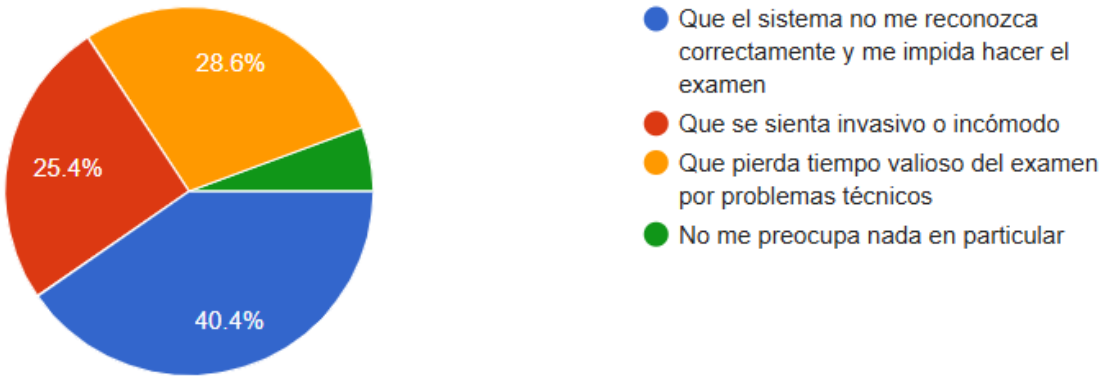


Figura 9. Principales preocupaciones sobre la implementación de reconocimiento facial

Tabla 10. Preocupaciones sobre la implementación de reconocimiento facial

Opciones	Respuestas	Porcentaje
Que el sistema no me reconozca correctamente y me impida hacer el examen	86	40.4%
Que se sienta invasivo o incómodo	54	25.4%
Que pierda tiempo valioso del examen por problemas técnicos	61	28.6%
No me preocupa nada en particular	12	5.6%
Total	213	100%

Análisis:

Para cuantificar el nivel de preocupación respecto a la implementación, se calculó el Índice de Riesgo Percibido (IRP). Se asignaron pesos según impacto: no reconozca = 5 (crítico), problemas técnicos = 4 (alto), invasivo = 3 (medio), sin preocupación = 1 (bajo).

Fórmula del Índice de Riesgo Percibido (IRP):

$$IRP = \frac{\sum_{i=1}^{n}(W_i \times P_i)}{100}$$

Donde:

W_i = Peso de impacto (1-5)

P_i = Porcentaje de cada categoría

Calculo:

$$IRP = \frac{(5 \times 40.4) + (4 \times 28.6) + (3 \times 25.4) + (1 \times 5.6)}{100}$$

$$IRP = \frac{202.0 + 114.4 + 76.2 + 5.6}{100} = \frac{398.2}{100} = 3.98$$

El Índice de Riesgo Percibido de 3.98 sobre 5.0 (79.6%) revela preocupaciones significativas que deben atenderse. La principal inquietud es de carácter técnico-operativo: el temor a que el sistema no reconozca correctamente y les impida realizar el examen. Las preocupaciones técnicas superan ampliamente a las preocupaciones personales de invasividad. Esto es positivo porque indica que la resistencia es práctica y solucionable mediante buena implementación técnica, no ideológica.

Pregunta 6: ¿En qué tipos de exámenes considera que sería más necesario usar reconocimiento facial?

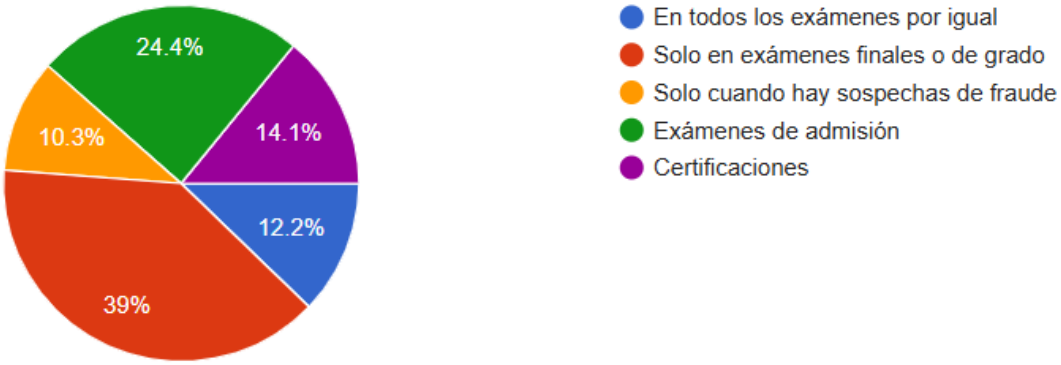


Figura 10. Percepción sobre el uso de reconocimiento facial en diferentes tipos de exámenes

Tabla 11. Percepción sobre el uso de reconocimiento facial en diferentes tipos de exámenes

Opciones	Respuesta	Porcentaje
En todos los exámenes por igual	26	12.2%
Solo en exámenes finales o de grado	83	39.0%
Solo cuando hay sospechas de fraude	22	10.3%
Exámenes de admisión	52	24.4%
Certificaciones	30	14.1%
Total	213	100%

Análisis:

Para establecer un orden de prioridad en la implementación según el tipo de examen, se calculó el Índice de Priorización (IP). Se asignaron pesos de criticidad: finales/grado = 5, admisión = 4, certificaciones = 3, con sospechas = 2, todos por igual = 2.

Fórmula del Índice de Priorización (IP):

$$IP_i = \frac{C_i \times P_i}{100}$$

Donde:

C_i = Criticidad del tipo de examen (2-5)

P_i = Porcentaje de preferencia

Cálculos:

$$IP_{finales} = \frac{5 \times 39.0}{100} = 1.95$$

$$IP_{admision} = \frac{4 \times 24.4}{100} = 0.976$$

$$IP_{certificaciones} = \frac{3 \times 14.1}{100} = 0.423$$

$$IP_{sospechas} = \frac{2 \times 10.3}{100} = 0.206$$

$$IP_{todos} = \frac{2 \times 12.2}{100} = 0.244$$

Índice de Aceptación de Implementación Gradual:

$$\text{IAIG} = 39.0 + 24.4 + 14.1 + 10.3 = 87.8\%$$

Los exámenes finales o de grado obtienen el IP más alto (1.95), identificándose como la mayor prioridad según los estudiantes. La mayoría de los estudiantes favorece una implementación selectiva del sistema, donde solo una minoría considera necesario su uso en todos los exámenes por igual o únicamente cuando hay sospechas de fraude. Este patrón de respuestas refleja que los estudiantes reconocen diferentes niveles de criticidad en los tipos de evaluación y prefieren aplicar el reconocimiento facial de manera diferenciada según la importancia del examen.

Pregunta 7: Si el reconocimiento facial fallara durante su examen (no te reconoce o se desconecta), ¿qué solución preferiría?

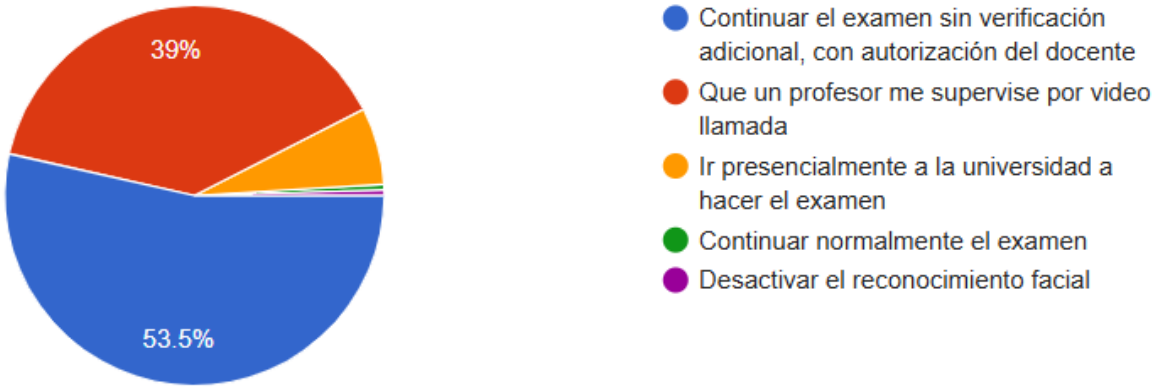


Figura 11. Preferencias de solución ante fallos del reconocimiento facial durante exámenes virtuales

Tabla 12. Preferencias de solución ante fallos del reconocimiento facial durante exámenes virtuales

Opciones	Respuesta	Porcentaje
Continuar el examen sin verificación adicional, con autorización del docente	114	53.5%
Que un profesor me supervise por video llamada	83	39.0%
Ir presencialmente a la universidad a hacer el examen	14	6.6%
Continuar normalmente el examen	1	0.5%
Desactivar el reconocimiento facial	1	0.5%
Total	213	100%

Análisis:

Para evaluar el equilibrio entre seguridad y continuidad ante fallos técnicos, se calcularon el Índice de Tolerancia al Fallo (ITF) y el Índice de Prioridad de Continuidad (IPC). Para el ITF se asignaron valores según nivel de control: supervisión profesor = 5, continuar con autorización = 4, ir presencial = 3, continuar normal = 2, desactivar RF = 1

Fórmula del Índice de Tolerancia al Fallo (ITF):

$$ITF = \frac{\sum_{i=1}^{n}(V_i \times P_i)}{100}$$

Cálculo del ITF:

$$ITF = \frac{(5 \times 39.0) + (4 \times 53.5) + (3 \times 6.6) + (2 \times 0.5) + (1 \times 0.5)}{100}$$

$$ITF = \frac{195 + 214 + 19.8 + 1.0 + 0.5}{100} = \frac{430.3}{100} = 4.30$$

Índice de Prioridad de Continuidad (IPC):

$$\text{IPC} = 53.5\% + 0.5\% = 54.0\%$$

El ITF de 4.30 indica preferencia por soluciones que mantengan control, pero permitan continuidad. La mayoría prefiere continuar el examen con autorización del docente, reflejando confianza en el criterio académico y necesidad de evitar interrupciones. Un grupo considerable opta por supervisión alternativa mediante videollamada, demostrando apertura hacia métodos complementarios

4.1.2. Propuesta

Se propone el desarrollo de un sistema de verificación biométrica compuesto por dos componentes integrados: un plugin para Moodle que intercepte el acceso a los exámenes y capture la imagen facial del estudiante mediante la cámara web, y un servidor Flask desarrollado en Python que procese las imágenes utilizando algoritmos de Aprendizaje Profundo.

4.1.3. Selección y justificación de tecnologías

Tabla 13. Selección y justificación de tecnologías

Tecnología	Descripción y justificación
Flask (Python)	Microframework web ligero utilizado para crear el servidor de inteligencia artificial. Permite desarrollar APIs REST de forma rápida y modular, facilitando la comunicación con Moodle.
PyTorch	Framework de aprendizaje profundo empleado para ejecutar los modelos de reconocimiento facial. Su flexibilidad y alto rendimiento lo convierten en una opción ideal para procesamiento en CPU o GPU.
ONNX Runtime	Motor de inferencia que permite ejecutar modelos exportados de PyTorch o TensorFlow en diferentes plataformas, mejorando la portabilidad y optimización del sistema.
InsightFace (ArcFace)	Modelo de verificación facial que genera vectores de características altamente precisos, mejorando la autenticación de identidad mediante comparación de embeddings.
RetinaFace	Detector facial que localiza y alinea rostros con alta precisión, asegurando una entrada de calidad para el modelo ArcFace.
MiniFASNet (Silent-Face)	Red neuronal ligera utilizada para detección de vida (anti-spoofing), evitando fraudes mediante fotografías o videos.
EasyOCR	Librería de reconocimiento óptico de caracteres que permite leer y validar números de cédula en documentos mostrados por el usuario.
OpenCV	Biblioteca fundamental de visión por computadora usada para capturar, preprocesar y analizar imágenes antes del reconocimiento facial.
PHP (Plugin Moodle)	Lenguaje empleado en el desarrollo del plugin tipo *Quiz Access Rule*, que conecta la plataforma Moodle con el servidor Flask para la verificación facial.

Flask-CORS	Extensión de Flask que habilita el intercambio seguro de datos entre dominios diferentes (Moodle y servidor de IA), evitando errores de seguridad.
WebRTC (getUserMedia API)	API del navegador que permite acceder a la cámara del usuario, capturar imágenes y transmitirlas al servidor en tiempo real.
MySQL / MariaDB	Sistema de gestión de bases de datos utilizado para almacenar la configuración y perfiles de usuarios verificados, totalmente compatible con Moodle.
Git	Herramienta de control de versiones que permite llevar registro del código y colaborar eficientemente durante el desarrollo del sistema.
Python venv	Entorno virtual de Python que aísla dependencias y garantiza la estabilidad del sistema durante las etapas de desarrollo y pruebas.

4.1.4. Requerimientos funcionales

Tabla 14. Requerimientos funcionales

ID	Nombre del Requerimiento	Descripción	Prioridad
RF-01	Verificación basada en foto de perfil Moodle	El sistema realizará la verificación facial tomando como referencia la imagen de perfil registrada del usuario en Moodle, la cual será utilizada para comparar con la imagen capturada en tiempo real.	Alta
RF-02	Inicio de verificación facial	El usuario debe iniciar manualmente el proceso de verificación facial antes de rendir el examen en línea.	Alta
RF-03	Solicitud de acceso a la cámara	El sistema debe solicitar permiso al usuario para acceder a la cámara del dispositivo y poder capturar su rostro.	Alta
RF-04	Captura de imagen facial en vivo	El sistema debe capturar una imagen del rostro del usuario en tiempo real mediante la cámara web.	Alta
RF-05	Envío de imagen al servidor Flask	El plugin enviará la imagen capturada, junto con los datos del usuario y del examen, al servidor Flask para su procesamiento.	Alta
RF-06	Detección de rostro real (anti-spoofing)	El servidor Flask debe verificar que la imagen capturada corresponda a un rostro real y no a una fotografía o pantalla.	Alta
RF-07	Comparación facial con foto de perfil	El sistema debe comparar el rostro capturado en vivo con la foto de perfil del usuario para determinar su identidad.	Alta
RF-08	Evaluación de umbral de similitud	El sistema debe determinar si la similitud entre ambos rostros supera el umbral definido para aprobar la verificación.	Alta
RF-09	Envío de resultado de verificación	El servidor Flask debe devolver al plugin Moodle un resultado en formato JSON indicando si	Alta

		la verificación fue exitosa o fallida.	
RF-10	Actualización de interfaz Moodle	El plugin debe mostrar al usuario el resultado de la verificación facial en pantalla dentro de Moodle.	Alta
RF-11	Habilitación del acceso al examen	Si la verificación facial es exitosa, el sistema permitirá el acceso al examen en línea. En caso contrario, se negará.	Alta
RF-12	Configuración de modo de verificación por el docente	El docente podrá activar o desactivar el uso del reconocimiento facial en sus cuestionarios, eligiendo entre los modos: "sin verificación facial", "con verificación facial sin validación" o "con verificación facial completa".	Media

4.1.5. Requerimientos no funcionales

Tabla 15. Requerimientos no funcionales

ID	Nombre del Requerimiento	Descripción	Prioridad
RNF-01	Rendimiento del sistema	El proceso completo de verificación facial (captura, análisis y respuesta) no debe superar los 2 segundos en condiciones normales de red.	Alta
RNF-02	Precisión del reconocimiento facial	El sistema debe mantener una precisión mínima del 95% en la identificación del usuario, minimizando falsos positivos y negativos.	Alta
RNF-03	Seguridad de la información	Las imágenes y datos personales del usuario deben transmitirse mediante canales seguros (HTTPS) y eliminarse automáticamente después del proceso.	Alta
RNF-04	Protección contra suplantación	El sistema debe implementar un modelo de detección anti-spoofing que evite intentos de fraude mediante fotografías o pantallas.	Alta
RNF-05	Usabilidad de la interfaz	La interfaz del plugin en Moodle debe ser intuitiva, mostrando mensajes claros y visuales sobre el estado de la verificación facial.	Media
RNF-06	Compatibilidad multiplataforma	El sistema debe ser compatible con navegadores modernos (Chrome, Firefox, Edge) y funcionar con cámaras web estándar.	Media

4.1.6. Diseño del sistema

El diseño del sistema representa la fase en la que se define la estructura, funcionamiento y comunicación entre los componentes que conforman el sistema de verificación facial para la autenticación de identidad en exámenes en línea. Este sistema se integra con la plataforma Moodle mediante un plugin desarrollado en PHP y JavaScript, el cual se comunica con un servidor Flask encargado del procesamiento biométrico de las imágenes faciales.

El propósito de este diseño es garantizar la correcta interacción entre los módulos del sistema, asegurando que la verificación facial se realice de manera eficiente, segura y transparente para el usuario. En este apartado se presentan los diagramas que describen la arquitectura del sistema, el flujo de datos, los componentes principales y la interacción con el usuario durante el proceso de autenticación.

4.1.6.1. Diagrama de contexto del sistema

El diagrama de contexto proporciona una vista de alto nivel del sistema de verificación facial, identificando los límites del sistema, los actores que interactúan con él y los sistemas externos necesarios para su funcionamiento.

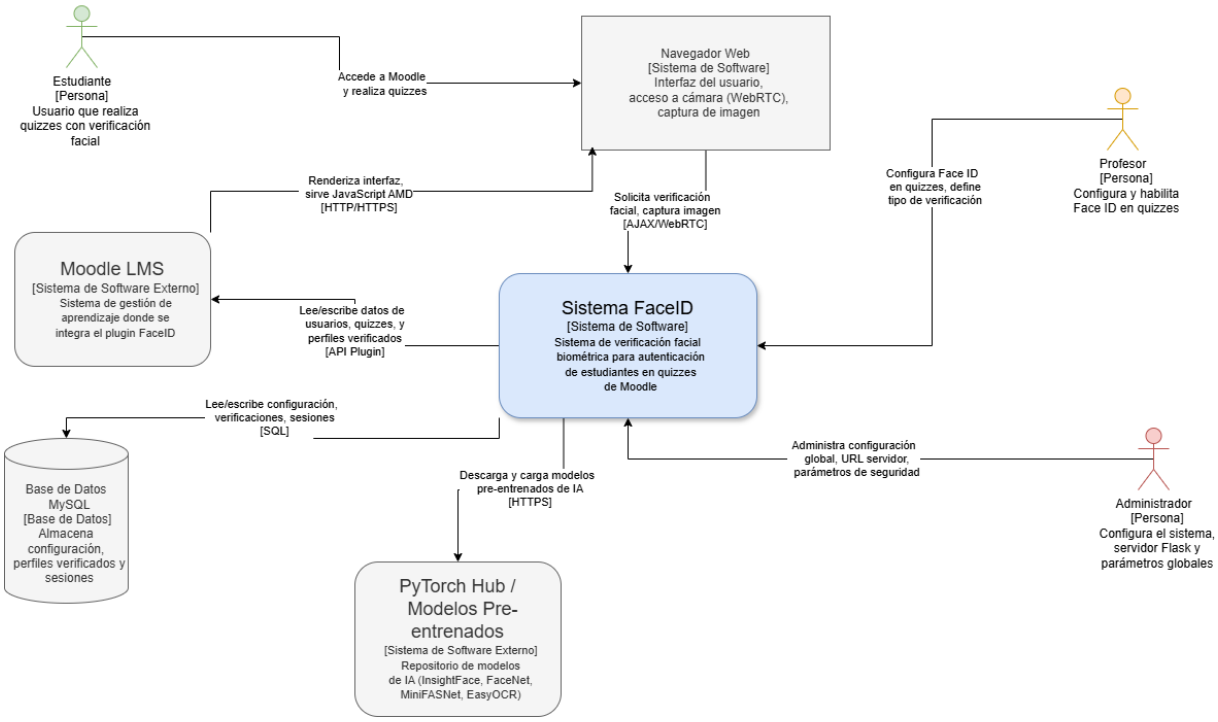


Figura 12. Diagrama de contexto

4.1.6.2. Diagrama de componentes

El diagrama de componentes representa la estructura lógica del sistema FaceID, mostrando los módulos principales que lo conforman, las tecnologías utilizadas y la forma en que se comunican entre sí. Este diagrama permite visualizar la arquitectura del sistema desde una perspectiva modular, identificando claramente la interacción entre el plugin de Moodle, el servidor Flask de inteligencia artificial y la base de datos MySQL.

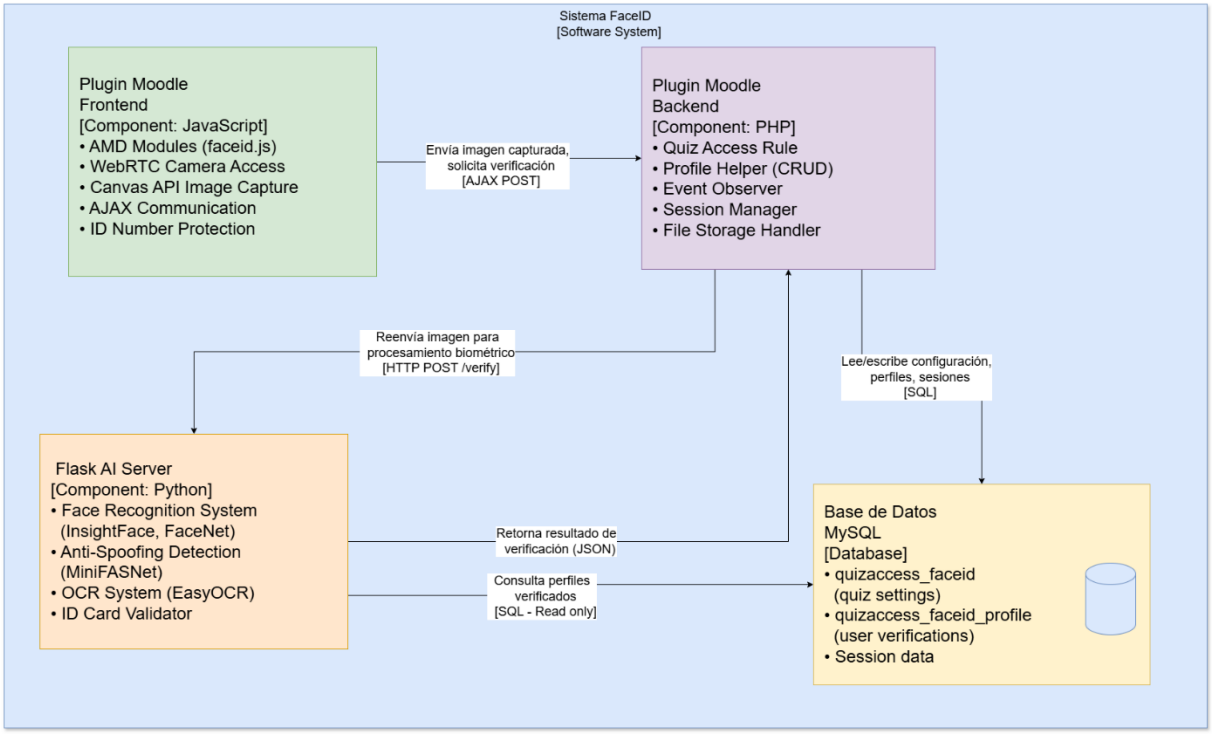


Figura 13. Diagrama de componentes

4.1.6.3. Diagramas de caso de uso

Los diagramas de casos de uso permiten representar de manera gráfica las principales interacciones entre los usuarios del sistema y las funcionalidades que este ofrece. Su objetivo es mostrar de forma general el comportamiento del sistema desde la perspectiva del usuario, identificando los actores, los casos de uso y las relaciones que existen entre ellos.

Figura 14. Caso de uso estudiante

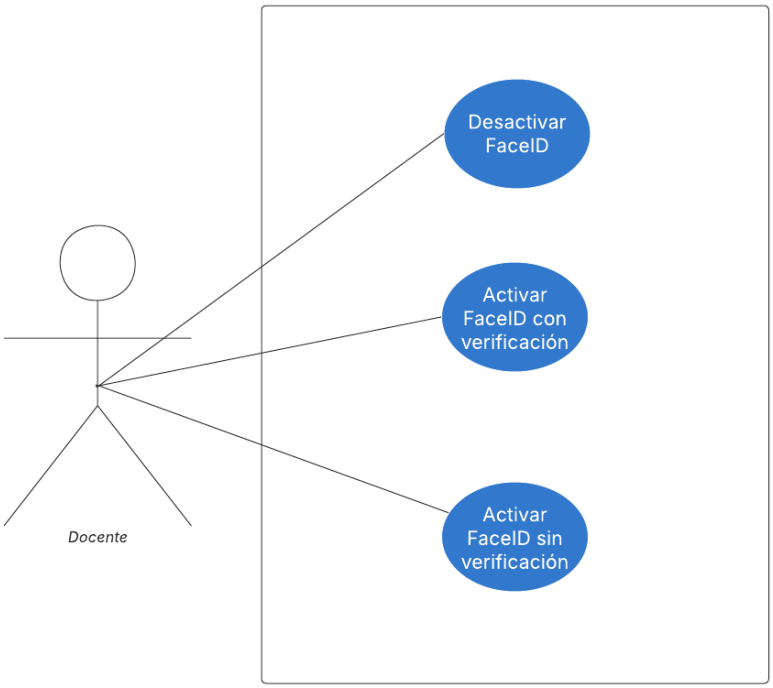


Figura 15. Caso de uso docente

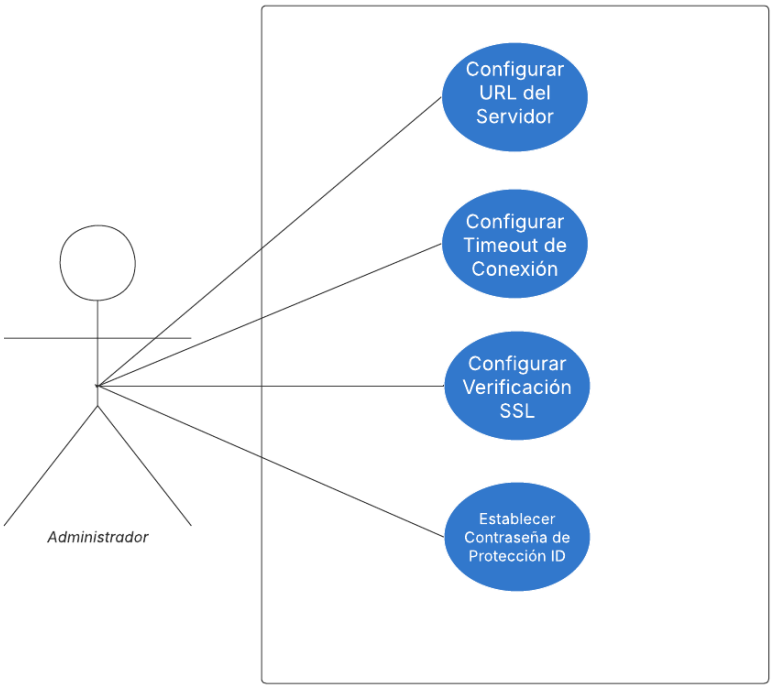


Figura 16. Caso de uso administrador

4.1.6.4. Diagrama de flujo

El diagrama de flujo presentado en esta sección describe, de manera secuencial y estructurada, el proceso completo de verificación de identidad mediante reconocimiento facial implementado en el sistema de exámenes en línea. Su objetivo es representar gráficamente la lógica que sigue la plataforma desde el momento en que el estudiante intenta acceder al quiz, hasta la validación final que permite o bloquea su ingreso. Este diagrama detalla los diferentes escenarios de uso, incluyendo la activación del módulo Face ID, la verificación básica por sesión, la verificación con perfil registrado, el uso de modelos de anti-spoofing para evitar suplantaciones y la comparación biométrica basada en embeddings generados por herramientas como InsightFace y FaceNet. Asimismo, se visualizan las interacciones entre Moodle, el servidor Flask de reconocimiento facial y el navegador del usuario, permitiendo comprender cómo se integran los componentes técnicos para garantizar un proceso de autentificación confiable.

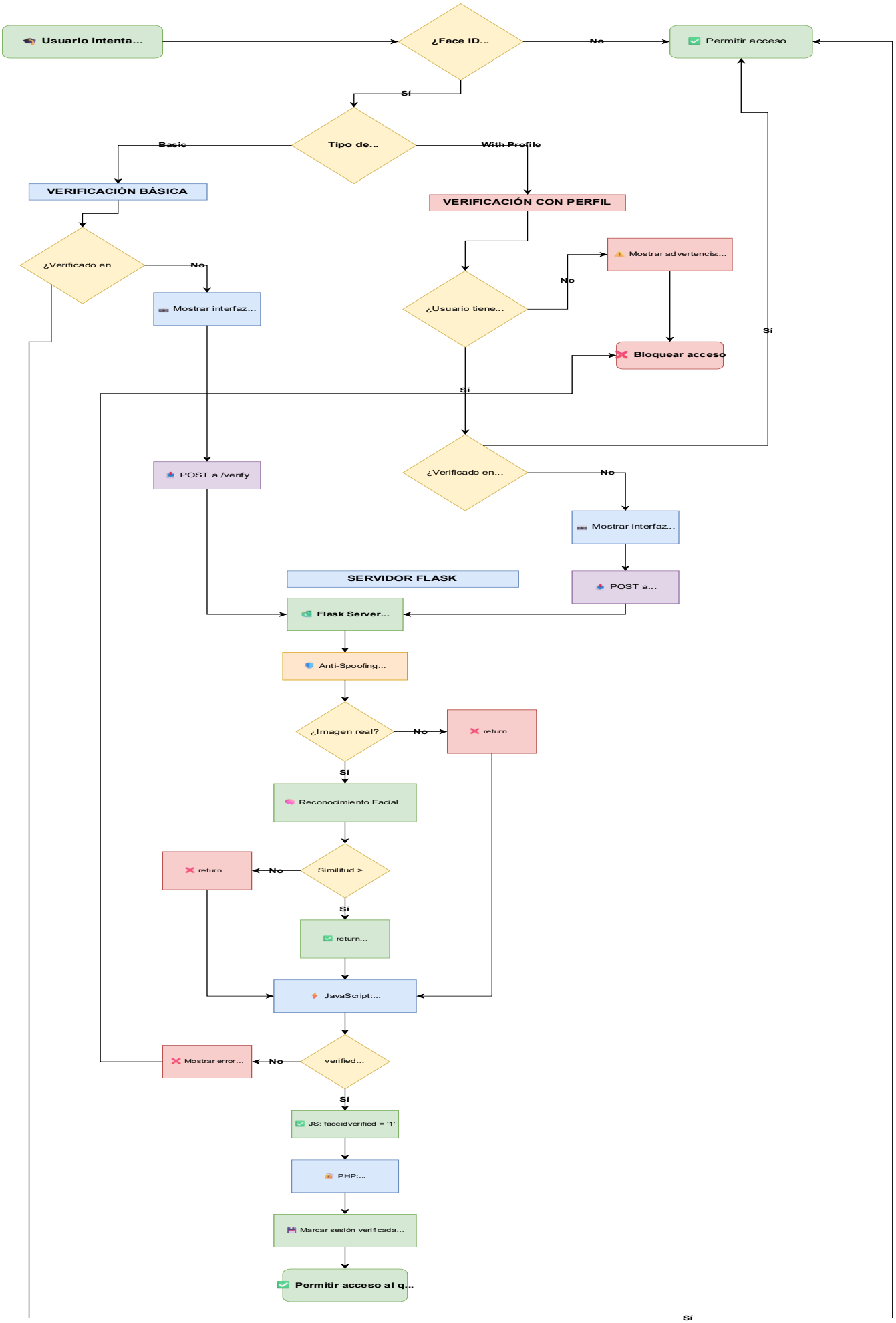


Figura 17. Diagrama de flujo

4.1.6.5. Diagrama de Entidad-Relación

El Diagrama de Entidad-Relación representa la estructura de persistencia de datos del sistema de verificación biométrica, documentando las entidades, sus atributos y las relaciones entre ellas. Este modelo constituye un elemento fundamental para comprender la arquitectura de almacenamiento del plugin Face ID. El sistema implementa dos tablas principales: quizaccess_faceid, que almacena la configuración de verificación por cuestionario, y quizaccess_faceid_profile, que mantiene el estado de verificación de cada usuario. Estas tablas se integran con el núcleo de Moodle mediante relaciones de clave foránea con las tablas user, quiz, quiz_attempts y files, aprovechando la infraestructura existente de la plataforma.

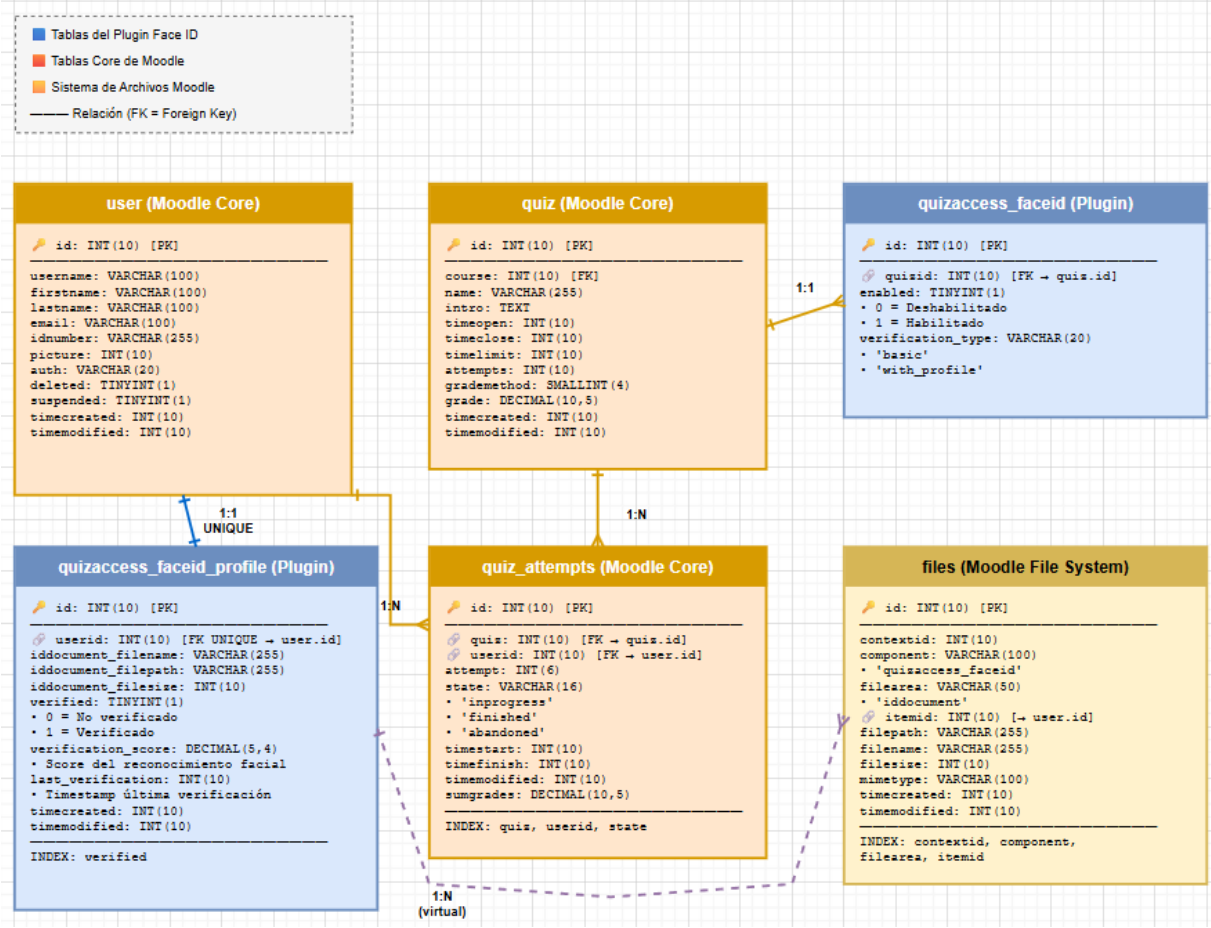


Figura 18. Diagrama de Entidad-Relación

4.1.7. Desarrollo

La fase de desarrollo comprende la implementación técnica de los componentes del sistema de verificación biométrica, abarcando tanto el servidor de procesamiento de imágenes basado en inteligencia artificial como el plugin de integración con Moodle. Este proceso se realizó siguiendo una metodología de desarrollo modular,

priorizando la modularidad, escalabilidad y seguridad del sistema. A continuación, se detallan los aspectos técnicos más relevantes de la implementación.

4.1.7.1. Configuración del Entorno de Desarrollo Python

El entorno Python se configuró con la versión 3.8, instalando las librerías especializadas para procesamiento de imágenes y modelos de inteligencia artificial mediante el gestor de paquetes pip.

RFSERVER > requirements.txt

```
torch>=2.0.0
torchvision>=0.15.0
numpy>=1.24.0
opencv-python>=4.8.0
Pillow>=10.0.0
insightface>=0.7.3
deepface>=0.0.79
retina-face>=0.0.13
face-recognition>=1.3.0
onnxruntime>=1.16.0
facenet-pytorch>=2.5.0
flask>=3.0.0
flask-cors>=4.0.0
requests>=2.31.0
scikit-image>=0.21.0
imutils>=0.5.4
albumentations>=1.3.1
tensorflow==1.15.0
protobuf>=3.11.3,<4.0.0
easyocr>=1.7.0
tqdm>=4.65.0
easydict>=1.10
```

Figura 19. Archivo de dependencias del servidor Flask

- Estructura de directorios del servidor

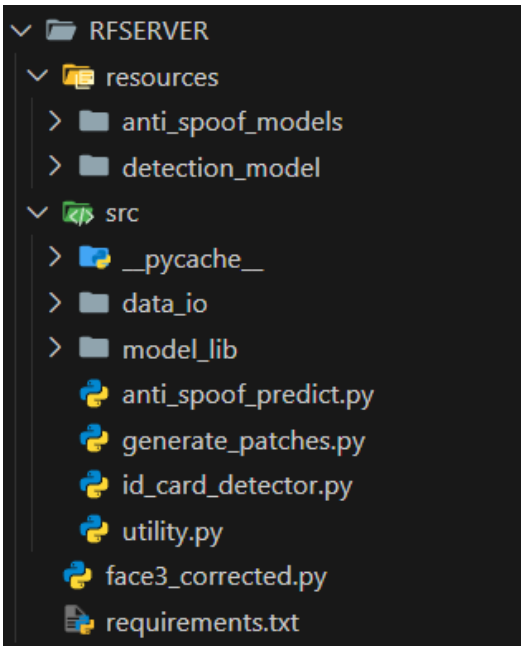


Figura 20. Estructura de directorios del servidor Flask

4.1.7.2. Implementación de modelos de Reconocimiento Facial

Como modelo principal se usa InsightFace el cual es un proyecto de análisis facial desarrollado y mantenido principalmente por Jia Guo y Jiankang Deng, quienes también son autores de los modelos relacionados ArcFace y RetinaFace, ampliamente reconocidos por su precisión en tareas biométricas (Deng et al., 2019; Deng et al., 2020).

```
if INSIGHTFACE_AVAILABLE:
    try:
        self.insightface = FaceAnalysis(providers=['CPUExecutionProvider'])
        self.insightface.prepare(ctx_id=0, det_size=(640, 640))
        logging.info("✅ InsightFace cargado correctamente")
    except Exception as e:
        logging.error(f"❌ Error cargando InsightFace: {e}")
        self.insightface = None
else:
    self.insightface = None
```

Figura 21. Código implementación InsightFace

Como modelo secundario de respaldo se implementa el conjunto MTCNN + FaceNet. MTCNN (Multi-task Cascaded Convolutional Networks) se utiliza para la detección precisa de rostros, realizando alineación y recorte antes del procesamiento; mientras que FaceNet emplea una arquitectura basada en Inception-ResNet-V1 entrenada con el conjunto VGGFace2 para generar embeddings faciales altamente discriminativos. Esta combinación permite disponer de un sistema alternativo de reconocimiento en caso de fallos del modelo principal, manteniendo robustez y consistencia biométrica (Schroff et al., 2015; Cao et al., 2018).

```
try:
    self.mtcnn = MTCNN(
        image_size=160, margin=0, min_face_size=20,
        thresholds=[0.6, 0.7, 0.7], factor=0.709,
        post_process=True, device=self.device
    )
    self.facenet = InceptionResnetV1(pretrained='vggface2').eval().to(self.device)
    logging.info("✅ FaceNet cargado correctamente")
except Exception as e:
    logging.error(f"❌ Error cargando FaceNet: {e}")
    self.mtcnn = None
    self.facenet = None
```

Figura 22. Código implementación FaceNet

4.1.7.3. Mecanismos de Seguridad en el Proceso de Reconocimiento Facial

El sistema aplica seguridad biométrica mediante detección de rostro vivo con el modelo Silent-Face-Anti-Spoofing, capaz de identificar intentos de suplantación con fotos o pantallas (Minivision AI, 2020). Además, se valida que solo exista un rostro en cada captura, reforzando la autenticidad del proceso de verificación.

```
model_test = AntiSpoofPredict(0)
image_cropper = CropImage()
logging.info("[SERVER] Silent-Face Anti-Spoofing cargado correctamente")
```

Figura 23. Código implementación Silent-Face-Anti-Spoofing

```
def detect_spoofing(image_data):
    """Detectar si una imagen es real o spoof usando Silent-Face-Anti-Spoofing"""
    try:
        # Convertir a numpy array
        if isinstance(image_data, io.BytesIO):
            image_data.seek(0)
            pil_image = Image.open(image_data)
            image_array = cv2.cvtColor(np.array(pil_image), cv2.COLOR_RGB2BGR)
        elif hasattr(image_data, 'read'):
            pil_image = Image.open(image_data)
            image_array = cv2.cvtColor(np.array(pil_image), cv2.COLOR_RGB2BGR)
        elif isinstance(image_data, np.ndarray):
            image_array = image_data
        elif isinstance(image_data, str):
            image_array = cv2.imread(image_data)
        else:
            pil_image = image_data
            image_array = cv2.cvtColor(np.array(pil_image), cv2.COLOR_RGB2BGR)

        if image_array is None:
            raise ValueError("No se pudo cargar la imagen")

        logging.info(f"[SPOOFING] Imagen cargada: {image_array.shape}")

        # Usar el sistema original de anti-spoofing
        image_bbox = [0, 0, image_array.shape[1], image_array.shape[0]]
        prediction = np.zeros((1, 3))

        for model_name in os.listdir("./resources/anti_spoof_models"):
            if model_name.endswith('.pth'):
                h_input, w_input, model_type, scale = parse_model_name(model_name)
                param = {
                    "org_img": image_array,
                    "bbox": image_bbox,
                    "scale": scale,
                    "out_w": w_input,
                    "out_h": h_input,
                    "crop": True,
                }

                if scale is None:
                    param["crop"] = False

                img = image_cropper.crop(**param)
                prediction += model_test.predict(img, os.path.join("./resources/anti_spoof_models", model_name))

        label = np.argmax(prediction)
        value = prediction[0][label] / len(os.listdir("./resources/anti_spoof_models"))

        if label == 1:
            result_label = "Real Face"
            is_real = True
        else:
            result_label = "Fake Face"
            is_real = False

        logging.info(f"[SPOOFING] Resultado: {result_label}, Score: {value:.3f}")

        return {
            'is_real': is_real,
            'score': float(value),
            'label': result_label
        }

    except Exception as e:
        logging.error(f"[SPOOFING] Error: {str(e)}")
        return {
            'is_real': False,
            'score': 0.0,
            'label': f'Error: {str(e)}'
        }
```

Figura 24. Código función detect_spoofing()

```
logging.error(f"[SECURITY] InsightFace detectó {len(faces)} rostros en la imagen")
return {
    'success': False,
    'error': f'Se detectaron {len(faces)} personas en la imagen. Por favor, asegúrese de estar solo en el encuadre.',
    'faces_count': len(faces)
}
```

Figura 25. Código validación de rostro único

4.1.7.4. OCR para verificar número de documento

Se utiliza un módulo OCR basado en EasyOCR para extraer automáticamente el número de documento desde la imagen de la cédula y compararlo con el registrado en el sistema, asegurando coherencia e identidad del usuario (JaidedAI, 2020).

```
class OCRSystem:

    def __init__(self):
        self.reader = None
        if EASYOCR_AVAILABLE:
            try:

                self.reader = easyocr.Reader(['es'], gpu=False, verbose=False)
                logging.info("[OCR] EasyOCR inicializado correctamente (es) - Modo optimizado")
            except Exception as e:
                logging.error(f"[OCR] Error inicializando EasyOCR: {e}")
                self.reader = None
        else:
            logging.warning("[OCR] EasyOCR no disponible")

    def extract_id_number(self, image: np.ndarray) -> dict:

        if self.reader is None:
            return {
                'found': False,
                'numbers': [],
                'confidence': 0.0,
                'raw_text': '',
                'error': 'OCR no disponible'
            }

        try:

            optimized_image = self._resize_for_ocr(image)

            logging.info("[OCR] Procesando imagen optimizada...")
            results_original = self.reader.readtext(
                optimized_image,
                paragraph=False,
                min_size=10,
                text_threshold=0.6,
                low_text=0.3,
                link_threshold=0.3,
                canvas_size=1280,
                mag_ratio=1.0
            )

            if not results_original:
                logging.info("[OCR] No se encontró texto en imagen original, probando con preprocesamiento...")
                processed_image = self._preprocess_for_ocr(image)
                results = self.reader.readtext(processed_image)
            else:
                results = results_original
```

Figura 26. Código clase OCRSystem

4.1.7.5. Validación Documental mediante ID Card Detector

Previo a la extracción OCR, el sistema implementa un módulo de validación documental que determina si la imagen cargada corresponde efectivamente a un documento de identidad oficial. Este detector analiza características geométricas y estructurales de la imagen mediante técnicas de visión por computadora con OpenCV, evaluando criterios como relación de aspecto, presencia de bordes definidos, y existencia de regiones de texto estructurado.

```python
class IDCardDetector:


    def __init__(self):

        self.logger = logging.getLogger(__name__)
        self.min_aspect_ratio = 1.3
        self.max_aspect_ratio = 1.9
        self.min_area_ratio = 0.05
        self.confidence_threshold = 0.4

        self.logger.info("IDCardDetector inicializado correctamente")

    def detect(self, image):

        # Cargar imagen si es una ruta
        if isinstance(image, str):
            img = cv2.imread(image)
            if img is None:
                return {
                    'is_valid': False,
                    'confidence': 0.0,
                    'details': {'error': 'No se pudo cargar la imagen'}
                }
        else:
            img = image.copy()

        # Validaciones básicas
        if img is None or img.size == 0:
            return {
                'is_valid': False,
                'confidence': 0.0,
                'details': {'error': 'Imagen vacía o inválida'}
            }

        height, width = img.shape[:2]
        total_area = height * width

        # Lista de caracteristicas detectadas
        features = {
            'rectangular_shape': 0.0,
            'correct_proportions': 0.0,
            'text_presence': 0.0,
            'edges_quality': 0.0,
            'color_consistency': 0.0
        }

        try:
            # 1. Detección de contornos rectangulares
            rect_score = self._detect_rectangular_shape(img)
            features['rectangular_shape'] = rect_score

            # 2. Verificar proporciones de documento
            prop_score = self._check_proportions(img)
            features['correct_proportions'] = prop_score

            # 3. Detectar presencia de texto (caracteristico de IDs)
            text_score = self._detect_text_regions(img)
            features['text_presence'] = text_score

            # 4. Calidad de bordes (documentos tienen bordes definidos)
            edge_score = self._analyze_edge_quality(img)
            features['edges_quality'] = edge_score
```

Figura 27. Código clase IDCardDetector

4.1.7.6. Definición de Endpoints de la API REST

El servidor Flask expone endpoints REST que permiten la comunicación entre el plugin de Moodle y el sistema de reconocimiento facial. Estos endpoints están diseñados para operaciones específicas: verificación facial en vivo, validación de documentos de identidad, pruebas de componentes individuales, y consultas de estado del sistema. Todas las respuestas utilizan formato JSON para facilitar el procesamiento en el cliente, y emplean el método HTTP POST para operaciones que transmiten datos sensibles (imágenes faciales) y GET para consultas de información.

- /verify – Verificación en vivo

  Compara la imagen capturada en tiempo real con la foto de perfil del usuario en Moodle e incluye detección anti-spoofing para evitar suplantaciones.

```python
@app.route('/verify', methods=['POST'])
def verify():
    """Ruta principal con sistema corregido - incluye info de número de ID"""
    try:
        user_id = request.form.get('userid')
        quizid = request.form.get('quizid')
        wwwroot = request.form.get('wwwroot', 'http://localhost')
        image_file = request.files.get('image')
        user_idnumber = request.form.get('idnumber', '')  # Número de ID del perfil

        if not user_id or not image_file:
            return jsonify(success=False, message='Faltan datos obligatorios'), 400

        logging.info(f"[SERVER] Verificando usuario {user_id} - SISTEMA CORREGIDO")
        if user_idnumber:
            logging.info(f"[SERVER] Usuario tiene número de ID registrado: {user_idnumber}")

        # Anti-spoofing
        spoofing_result = detect_spoofing(image_file)

        if not spoofing_result['is_real']:
            return jsonify(
                success=True,
                verified=False,
                blocked_by_antispoofing=True,
                message=f'Acceso bloqueado: {spoofing_result["label"]}',
                antispoofing_result=spoofing_result
            ), 200

        logging.info(f"[SERVER] ✓ Anti-spoofing aprobado")

        # Verificación facial
        image_file.seek(0)
        live_image = convert_image_data_to_array(image_file)
        if live_image is None:
            return jsonify(
                success=True,
                verified=False,
                message='No se pudo procesar la imagen',
                antispoofing_result=spoofing_result
            ), 200

        # Cargar imagen de perfil
        try:
            profile_url = f'{wwwroot}/user/pix.php/{user_id}/f3.jpg'
            response = requests.get(profile_url, timeout=10, verify=False)
            response.raise_for_status()
            profile_image = convert_image_data_to_array(io.BytesIO(response.content))

            if profile_image is None:
                raise ValueError("Imagen de perfil inválida")

        except Exception as e:
            return jsonify(
                success=True,
                verified=False,
                message=f'Error cargando imagen de perfil: {str(e)}',
                antispoofing_result=spoofing_result
            ), 200

        # Verificar con sistema corregido
        verification_result = corrected_system.verify_faces(
            profile_image, live_image, 'live_vs_profile'
        )
```

Figura 28. Código /verify

- /verify-profile – Verificación de documento

  Compara la foto de perfil del usuario con la imagen de su documento de identidad y utiliza OCR para validar el número de documento registrado.

```python
@app.route('/verify-profile', methods=['POST'])
def verify_profile():
    """Verificación de perfil CORREGIDA con comparación de Número de ID"""
    try:
        profile_url = request.form.get('profile_url')
        userid = request.form.get('userid')
        iddocument_file = request.files.get('iddocument')
        user_idnumber = request.form.get('idnumber', '')  # Número de ID del perfil de Moodle

        if not profile_url or not iddocument_file:
            return jsonify(success=False, message='Faltan datos: profile_url o iddocument'), 400

        logging.info(f"[PROFILE] Verificando perfil usuario {userid} - SISTEMA CORREGIDO")
        if user_idnumber:
            logging.info(f"[PROFILE] Número de ID del perfil Moodle: {user_idnumber}")

        # --- PASO 1: VALIDACIÓN DE CÉDULA ---
        id_card_result = None
        if id_card_detector is not None:
            try:
                # Leer imagen de cédula
                iddocument_file.seek(0)
                file_bytes = np.frombuffer(iddocument_file.read(), np.uint8)
                id_card_image = cv2.imdecode(file_bytes, cv2.IMREAD_COLOR)

                # Detectar si es un documento válido
                id_card_result = id_card_detector.detect(id_card_image)

                if not id_card_result['is_valid']:
                    return jsonify(
                        success=True,
                        verified=False,
                        message='La imagen proporcionada no corresponde a un documento de identidad válido. Por favor, capture una fotografía clara de su cédula o documento oficial.',
                        score=0.0,
                        id_card_detection=id_card_result
                    ), 200

                logging.info(f"[PROFILE] ✓ Cédula validada con confianza: {id_card_result['confidence']:.2%}")
                # Resetear archivo para siguiente procesamiento
                iddocument_file.seek(0)

            except Exception as e:
                logging.warning(f"[PROFILE] Error en validación de cédula: {str(e)}")
                # Continuar sin validación si hay error
                id_card_result = {'error': str(e)}
        else:
            logging.warning("[PROFILE] ID Card Detector no disponible, saltando validación")

        # --- PASO 2: Cargar imagen de perfil
        try:
            response = requests.get(profile_url, timeout=10, verify=False)
            response.raise_for_status()
            profile_image = convert_image_data_to_array(io.BytesIO(response.content))

            if profile_image is None:
                raise ValueError("Imagen de perfil inválida")

            logging.info(f"[PROFILE] Imagen de perfil cargada: {profile_image.shape}")
        except Exception as e:
            return jsonify(
                success=True,
                verified=False,
                message=f'Error cargando imagen de perfil: {str(e)}',
                score=0.0
```

Figura 29. Código /verify-profile

- /verify-with-profile – Verificación con perfil validado

  Realiza verificación facial usando la imagen en vivo, pero solo para usuarios que ya validaron previamente su documento de identidad.

```python
@app.route('/verify-with-profile', methods=['POST'])
def verify_with_profile():
    """
    Verificación facial en vivo comparando con perfil verificado del usuario - SISTEMA CORREGIDO
    Incluye verificación de número de ID si está disponible
    """
    try:
        # Obtener datos del request
        image_data = request.files.get('image')
        userid = request.form.get('userid')
        wwwroot = request.form.get('wwwroot')
        user_idnumber = request.form.get('idnumber', '')  # Número de ID del perfil

        if not all([image_data, userid, wwwroot]):
            return jsonify(success=False, message='Faltan datos: image, userid o wwwroot'), 400

        logging.info(f"[PROFILE-LIVE] Verificando imagen en vivo vs perfil para usuario {userid} - SISTEMA CORREGIDO")
        if user_idnumber:
            logging.info(f"[PROFILE-LIVE] Número de ID del perfil disponible: {user_idnumber}")

        # --- PASO 1: Anti-Spoofing ---
        try:
            antispoofing_result = detect_spoofing(image_data)

            if not antispoofing_result.get('is_real', False):
                logging.info(f"[PROFILE-LIVE] Usuario {userid} - Anti-spoofing falló")
                return jsonify(
                    success=True,
                    verified=False,
                    message='❌ Imagen detectada como falsificada',
                    antispoofing_result=antispoofing_result
                ), 200

            logging.info(f"[PROFILE-LIVE] Usuario {userid} - Anti-spoofing pasó ✓")

        except Exception as e:
            logging.exception("[PROFILE-LIVE] Error en anti-spoofing")
            return jsonify(
                success=True,
                verified=False,
                message=f'❌ Error en detección anti-spoofing: {str(e)}',
                antispoofing_result=False
            ), 200

        # --- PASO 2: Procesar imagen en vivo ---
        live_image = convert_image_data_to_array(image_data)
        if live_image is None:
            return jsonify(
                success=True,
                verified=False,
                message='❌ No se pudo procesar la imagen en vivo',
                antispoofing_result=antispoofing_result
            ), 200

        # --- PASO 3: Obtener imagen de perfil ---
        try:
            profile_url = f"{wwwroot}/user/pix.php/{userid}/f3.jpg"
            response = requests.get(profile_url, timeout=10, verify=False)
            response.raise_for_status()
            profile_image = convert_image_data_to_array(io.BytesIO(response.content))

            if profile_image is None:
                raise ValueError("Imagen de perfil inválida")

            logging.info(f"[PROFILE-LIVE] Imagen de perfil cargada desde {profile_url}")
        except Exception as e:
```

Figura 30. Codigo /verify-with-profile

4.1.7.7. Desarrollo del Plugin de Integración con Moodle

Para conectar el sistema de verificación facial con la plataforma Moodle, se desarrolló un plugin personalizado que actúa como puente entre el servidor Flask y las funcionalidades internas del módulo de cuestionarios.

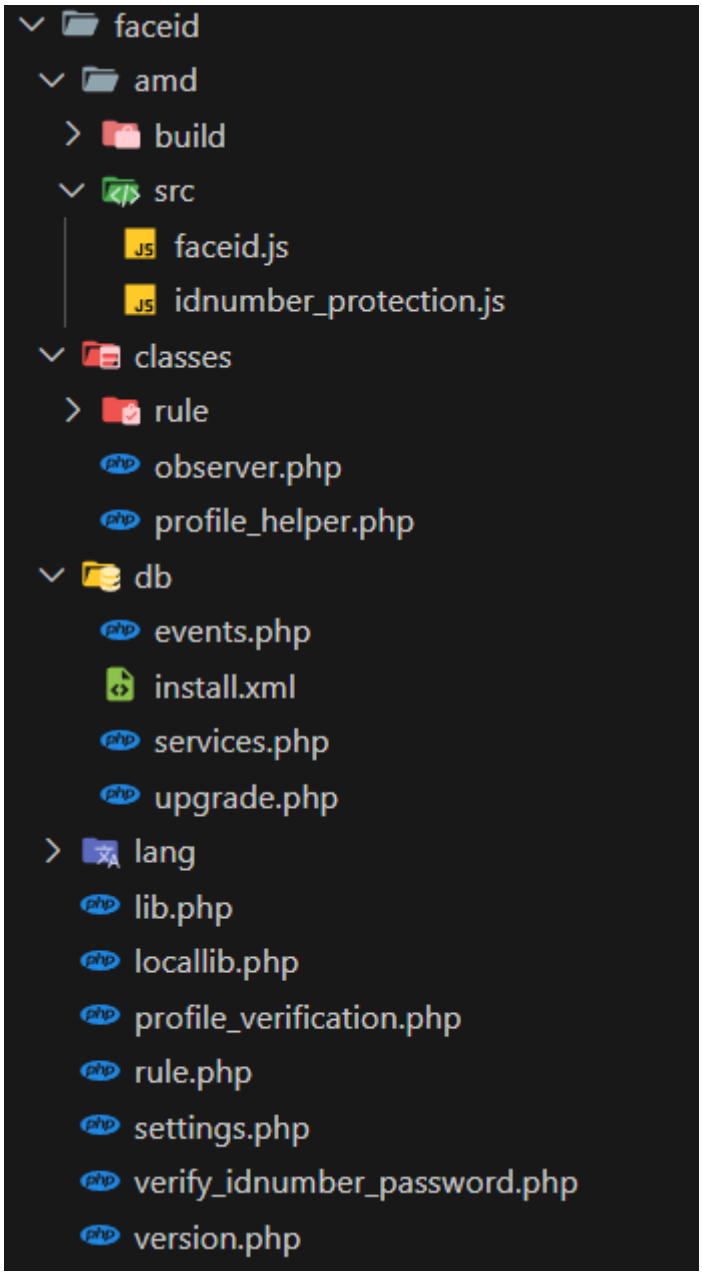


Figura 31. Estructura de directorios del plugin Moodle

- Archivo settings.php

Este Archivo es encargado de la configuración inicial al momento de instalar el plugin.

```php
<?php
// This file is part of Moodle - http://moodle.org/
defined('MOODLE_INTERNAL') || die();

if ($ADMIN->fulltree) {
    // Heading for server settings
    $settings->add(new admin_setting_heading(
        'quizaccess_faceid/serverheading',
        get_string('serversettings', 'quizaccess_faceid'),
        get_string('serversettingsdesc', 'quizaccess_faceid')
    ));

    // Server URL setting
    $settings->add(new admin_setting_configtext(
        'quizaccess_faceid/server_url',
        get_string('serverurl', 'quizaccess_faceid'),
        get_string('serverurldesc', 'quizaccess_faceid'),
        'http://127.0.0.1:5001',
        PARAM_URL
    ));

    // Connection timeout setting
    $settings->add(new admin_setting_configtext(
        'quizaccess_faceid/timeout',
        get_string('timeout', 'quizaccess_faceid'),
        get_string('timeoutdesc', 'quizaccess_faceid'),
        '10',
        PARAM_INT
    ));

    // SSL verification setting
    $settings->add(new admin_setting_configcheckbox(
        'quizaccess_faceid/verify_ssl',
        get_string('verifyssl', 'quizaccess_faceid'),
        get_string('verifyssldesc', 'quizaccess_faceid'),
        0
    ));

    // Heading for ID number protection
    $settings->add(new admin_setting_heading(
        'quizaccess_faceid/idnumberheading',
        get_string('idnumberprotection', 'quizaccess_faceid'),
        get_string('idnumberprotectiondesc', 'quizaccess_faceid')
    ));

    // ID number protection password
    $settings->add(new admin_setting_configpasswordunmask(
        'quizaccess_faceid/idnumber_password',
        get_string('idnumberpassword', 'quizaccess_faceid'),
        get_string('idnumberpassworddesc', 'quizaccess_faceid'),
        ''
    ));
}
```

Figura 32. Código Configuración administrativa del plugin FaceID

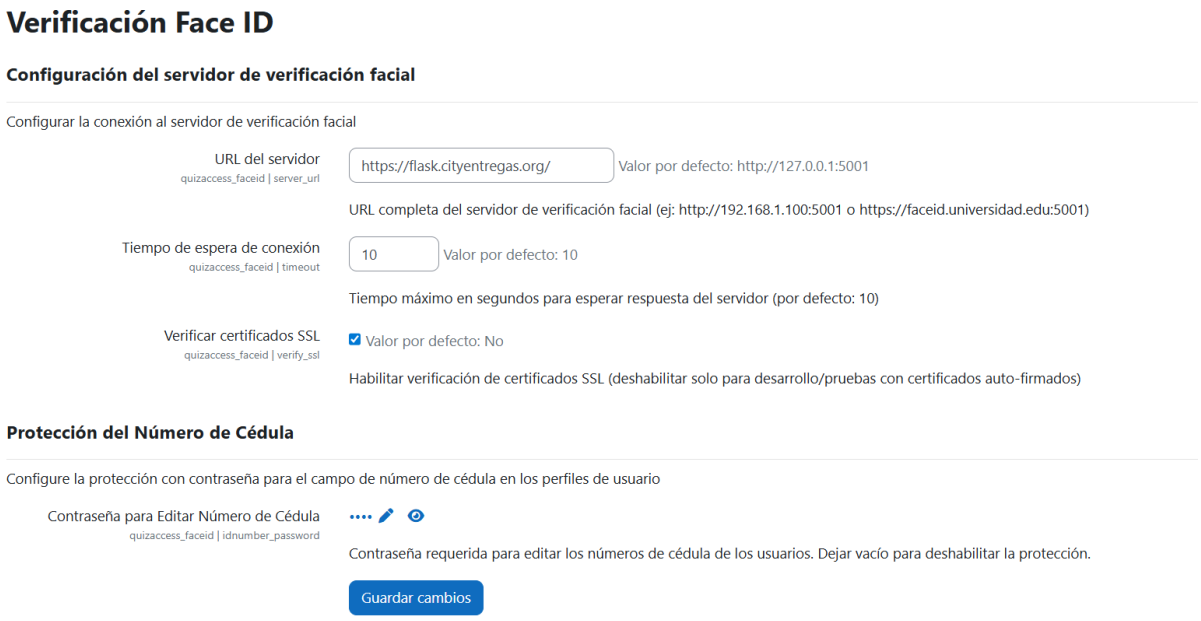


Figura 33. UI Configuración administrativa del plugin FaceID

- Archivo rule.php

Este apartado define la regla de acceso del plugin, implementando la lógica que valida si un estudiante puede iniciar o continuar un cuestionario.

```php
public static function make(\mod_quiz\quiz_settings $quizobj, $timenow, $canignoretimelimits) {
    global $DB;
    $rec = $DB->get_record('quizaccess_faceid', ['quizid' => $quizobj->get_quiz()->id]);
    return ($rec && $rec->enabled) ? new self($quizobj, $timenow) : null;
}
```

Figura 34. Función de Ciclo de Vida del Plugin

```php
public static function add_settings_form_fields(mod_quiz_mod_form $quizform, \MoodleQuickForm $mform) {
    // Create radio buttons for mutually exclusive options
    $radioarray = array();
    $radioarray[] = $mform->createElement('radio', 'faceid_verification_type', '',
        get_string('faceidenabled', 'quizaccess_faceid'), 'basic');
    $radioarray[] = $mform->createElement('radio', 'faceid_verification_type', '',
        get_string('faceid_with_profile', 'quizaccess_faceid'), 'with_profile');
    $radioarray[] = $mform->createElement('radio', 'faceid_verification_type', '',
        get_string('disabled', 'quizaccess_faceid'), 'disabled');

    $mform->addGroup($radioarray, 'faceid_verification_array',
        get_string('verification_mode', 'quizaccess_faceid'), array('<br/>'), false);
    $mform->addHelpButton('faceid_verification_array', 'verification_mode', 'quizaccess_faceid');
    $mform->setDefault('faceid_verification_type', 'disabled');

    // Add JavaScript to handle mutual exclusion
    global $PAGE;
    $PAGE->requires->js_amd_inline("
        require(['jquery'], function($) {
            $(document).ready(function() {
                // Handle radio button changes
                $('input[name=\"faceid_verification_type\"]').change(function() {
                    var selectedValue = $(this).val();
                    console.log('FaceID verification type changed to:', selectedValue);
                });
            });
        });
    ");
}
```

Figura 35. Función de Configuración del cuestionario

```php
public static function save_settings($quiz) {
    global $DB;

    // Get verification type from radio buttons
    $verification_type = isset($quiz->faceid_verification_type) ? $quiz->faceid_verification_type : 'disabled';
    $enabled = ($verification_type !== 'disabled') ? 1 : 0;

    $record = (object)[
        'quizid' => $quiz->id,
        'enabled' => $enabled,
        'verification_type' => $verification_type
    ];

    if ($DB->record_exists('quizaccess_faceid', ['quizid' => $quiz->id])) {
        $record->id = $DB->get_field('quizaccess_faceid', 'id', ['quizid' => $quiz->id]);
        $DB->update_record('quizaccess_faceid', $record);
    } else {
        $DB->insert_record('quizaccess_faceid', $record);
    }
}

public static function get_extra_settings($quizid) {
    global $DB;
    $rec = $DB->get_record('quizaccess_faceid', ['quizid' => $quizid]);
    if ($rec) {
        return [
            'faceid_enabled' => (int)$rec->enabled,
            'faceid_verification_type' => $rec->verification_type ?? 'disabled'
        ];
    }
    return ['faceid_verification_type' => 'disabled'];
}
```

Figura 36. Función Persistencia de Configuración

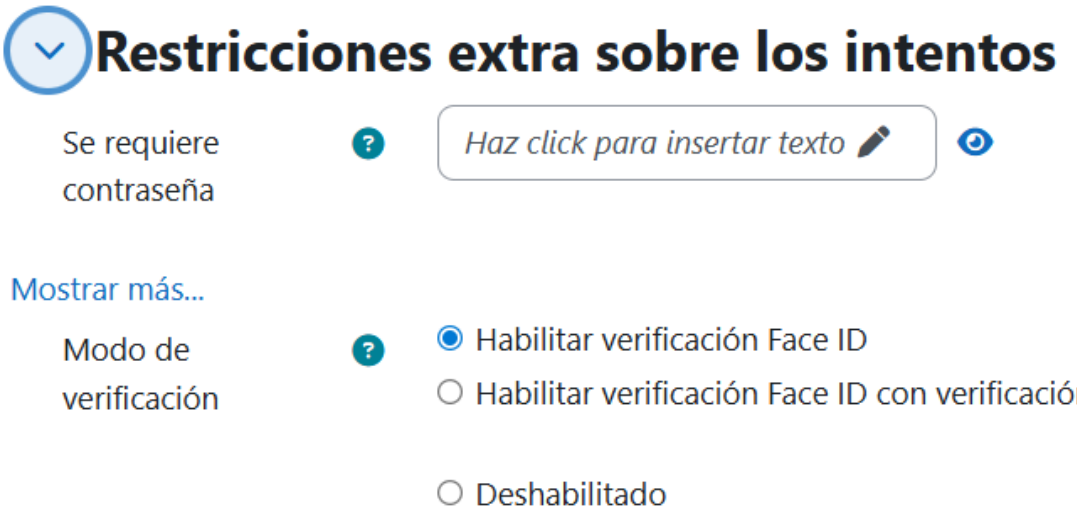
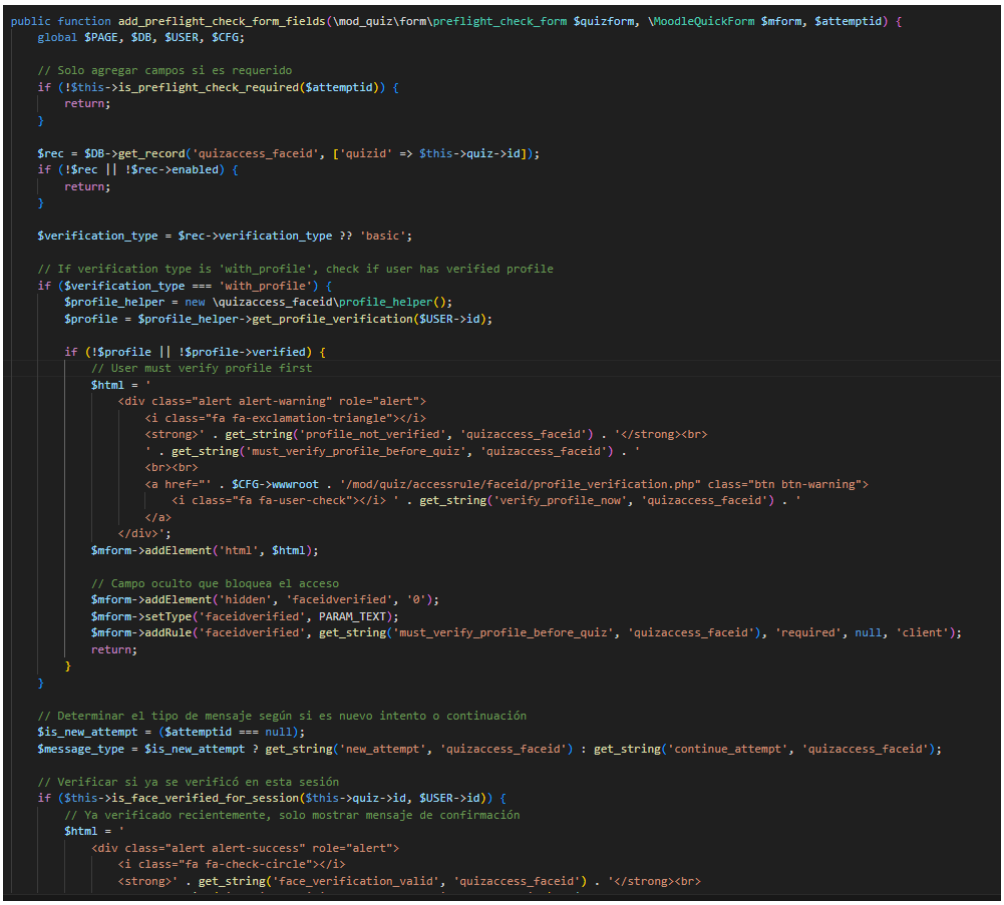


Figura 37. UI modos de verificación facial en el cuestionari

```php
public function is_preflight_check_required($attemptid) {
    global $USER;

    // Siempre requerir verificación facial, pero verificar si ya se hizo en esta sesión
    if ($attemptid === null) {
        // Nuevo intento - siempre requerir verificación
        return true;
    } else {
        // Continuar intento existente - verificar si ya se verificó en esta sesión
        return !$this->is_face_verified_for_session($this->quiz->id, $USER->id);
    }
}
```

Figura 38. Función Decisión de Verificación



```php
public function add_preflight_check_form_fields(\mod_quiz\form\preflight_check_form $quizform, \MoodleQuickForm $mform, $attemptid) {
    global $PAGE, $DB, $USER, $CFG;

    // Solo agregar campos si es requerido
    if (!$this->is_preflight_check_required($attemptid)) {
        return;
    }

    $rec = $DB->get_record('quizaccess_faceid', ['quizid' => $this->quiz->id]);
    if (!$rec || !$rec->enabled) {
        return;
    }

    $verification_type = $rec->verification_type ?? 'basic';

    // If verification type is 'with_profile', check if user has verified profile
    if ($verification_type === 'with_profile') {
        $profile_helper = new \quizaccess_faceid\profile_helper();
        $profile = $profile_helper->get_profile_verification($USER->id);

        if (!$profile || !$profile->verified) {
            // User must verify profile first
            $html = '
                <div class="alert alert-warning" role="alert">
                    <i class="fa fa-exclamation-triangle"></i>
                    <strong>' . get_string('profile_not_verified', 'quizaccess_faceid') . '</strong><br>
                    ' . get_string('must_verify_profile_before_quiz', 'quizaccess_faceid') . '
                    <br><br>
                    <a href="' . $CFG->wwwroot . '/mod/quiz/accessrule/faceid/profile_verification.php" class="btn btn-warning">
                        <i class="fa fa-user-check"></i> ' . get_string('verify_profile_now', 'quizaccess_faceid') . '
                    </a>
                </div>';
            $mform->addElement('html', $html);

            // Campo oculto que bloquea el acceso
            $mform->addElement('hidden', 'faceidverified', '0');
            $mform->setType('faceidverified', PARAM_TEXT);
            $mform->addRule('faceidverified', get_string('must_verify_profile_before_quiz', 'quizaccess_faceid'), 'required', null, 'client');
            return;
        }
    }

    // Determinar el tipo de mensaje según si es nuevo intento o continuación
    $is_new_attempt = ($attemptid === null);
    $message_type = $is_new_attempt ? get_string('new_attempt', 'quizaccess_faceid') : get_string('continue_attempt', 'quizaccess_faceid');

    // Verificar si ya se verificó en esta sesión
    if ($this->is_face_verified_for_session($this->quiz->id, $USER->id)) {
        // Ya verificado recientemente, solo mostrar mensaje de confirmación
        $html = '
            <div class="alert alert-success" role="alert">
                <i class="fa fa-check-circle"></i>
                <strong>' . get_string('face_verification_valid', 'quizaccess_faceid') . '</strong><br>
```

Figura 39. Código de generación de la interfaz de captura facial pre-cuestionario

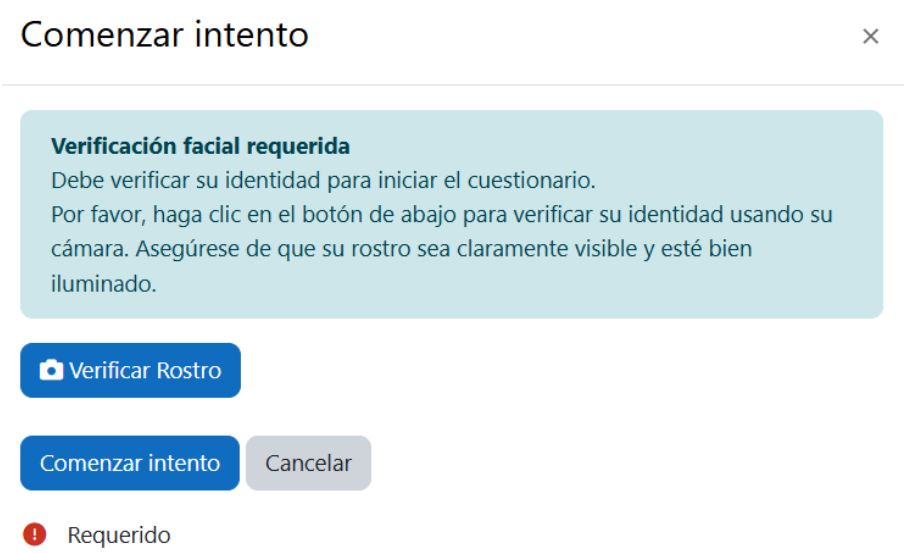


Figura 40. UI verificación facial previa al inicio del cuestionario

- Archivo profile_verification.php

Este archivo implementa la página de verificación de perfil del usuario. Permite la carga de documentos de identidad mediante formulario multipart/form-data, gestiona el almacenamiento de archivos en el sistema de filearea de Moodle, y

coordina la verificación biométrica del perfil enviando la foto de perfil del usuario y el documento cargado al servidor Flask para su comparación mediante reconocimiento facial.

```php
echo $OUTPUT->header();

// Display current verification status
echo $OUTPUT->heading(get_string('profileverification', 'quizaccess_faceid'), 2);

if ($profile) {
    $statusclass = $profile->verified ? 'alert-success' : 'alert-warning';
    $statusicon = $profile->verified ? '✅' : '❌';
    $statustext = $profile->verified ?
        get_string('verified', 'quizaccess_faceid') :
        get_string('not_verified', 'quizaccess_faceid');

    echo html_writer::div($statusicon . ' ' . $statustext, "alert {$statusclass}");

    if ($profile->verification_score > 0) {
        echo html_writer::div(
            get_string('verification_score', 'quizaccess_faceid',
                number_format($profile->verification_score, 3)),
            'small text-muted mb-2'
        );
    }

    if ($profile->last_verification) {
        echo html_writer::div(
            get_string('last_verification', 'quizaccess_faceid',
                userdate($profile->last_verification)),
            'small text-muted mb-3'
        );
    }
} else {
    echo html_writer::div('❌ ' . get_string('not_verified', 'quizaccess_faceid'), 'alert alert-warning');
}

echo html_writer::tag('hr', '');

// Upload form
echo html_writer::start_tag('form', array(
    'method' => 'post',
    'enctype' => 'multipart/form-data',
    'class' => 'mform'
));

echo html_writer::empty_tag('input', array('type' => 'hidden', 'name' => 'sesskey', 'value' => sesskey()));
```

Figura 41. Generación de la interfaz de estado de verificación del perfil

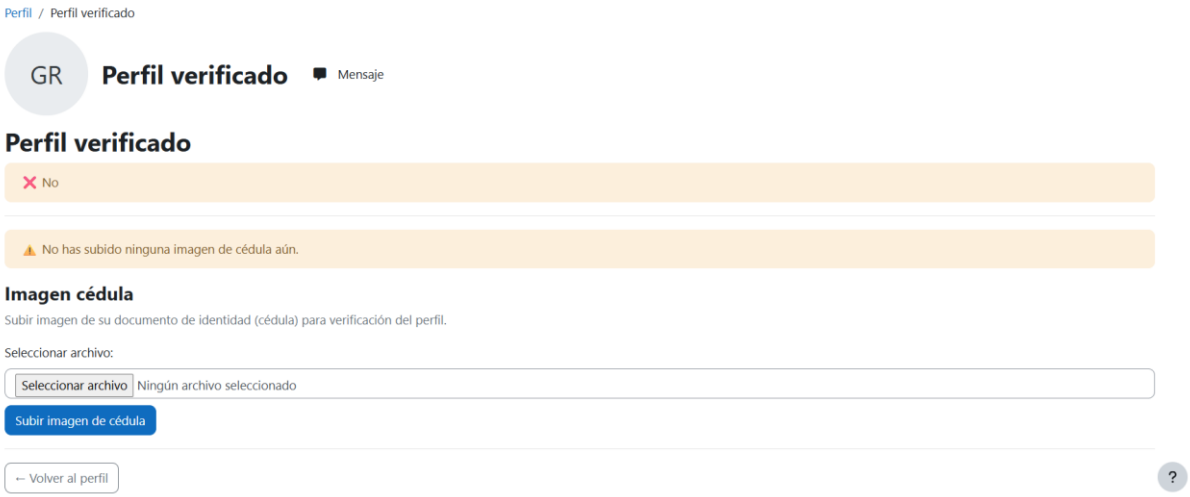


Figura 42. UI verificación de perfil con carga de documento de identidad

- Archivo verify_idnumber_password.php

Para fortalecer la seguridad del sistema, se implementó un mecanismo de protección del campo de número de identificación que bloquea su edición mediante validación por contraseña. Este control adicional previene modificaciones no autorizadas del número de ID registrado en el perfil del usuario.

```php
<?php
// This file is part of Moodle - http://moodle.org/

define('AJAX_SCRIPT', true);

require_once(__DIR__ . '/../../../../config.php');

require_login();
require_sesskey();

// Only process AJAX POST requests
if ($_SERVER['REQUEST_METHOD'] !== 'POST') {
    header('HTTP/1.1 405 Method Not Allowed');
    die();
}

$password = required_param('password', PARAM_RAW);

// Get configured password
$configured_password = get_config('quizaccess_faceid', 'idnumber_password');

// Response array
$response = [
    'success' => false,
    'message' => ''
];

// Check if password protection is enabled (password is configured)
if (empty($configured_password)) {
    $response['success'] = false;
    $response['message'] = get_string('idnumberpassword_notconfigured', 'quizaccess_faceid');
} else if ($password === $configured_password) {
    // Password matches
    $response['success'] = true;
    $response['message'] = get_string('idnumberpassword_correct', 'quizaccess_faceid');
} else {
    // Password does not match
    $response['success'] = false;
    $response['message'] = get_string('idnumberpassword_incorrect', 'quizaccess_faceid');
}

// Send JSON response
header('Content-Type: application/json');
echo json_encode($response);
```

Figura 43. Código de verificación de contraseña para desbloqueo del campo ID

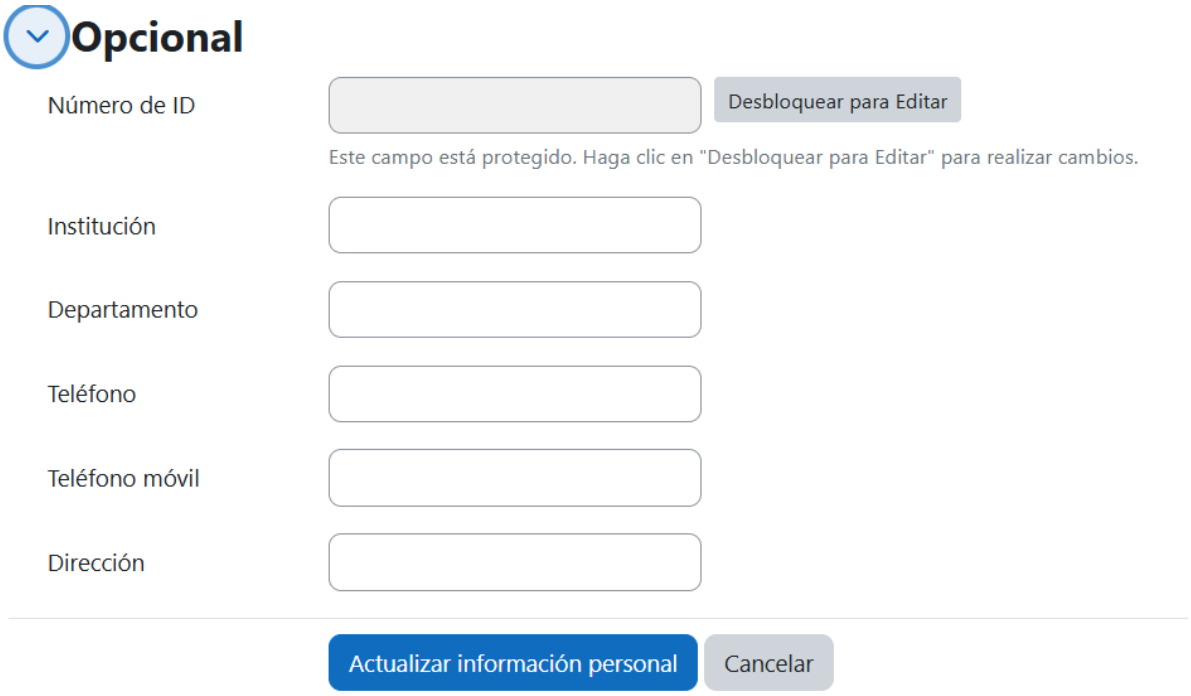


Figura 44. Vista del campo de número ID protegido en el perfil de usuario

- Archivo continuous_verify.php

Este archivo permite monitoreo continuo del estudiante mientras está realizando un quiz, verificando periódicamente que la persona frente a la cámara sigue siendo la misma que inició el examen.

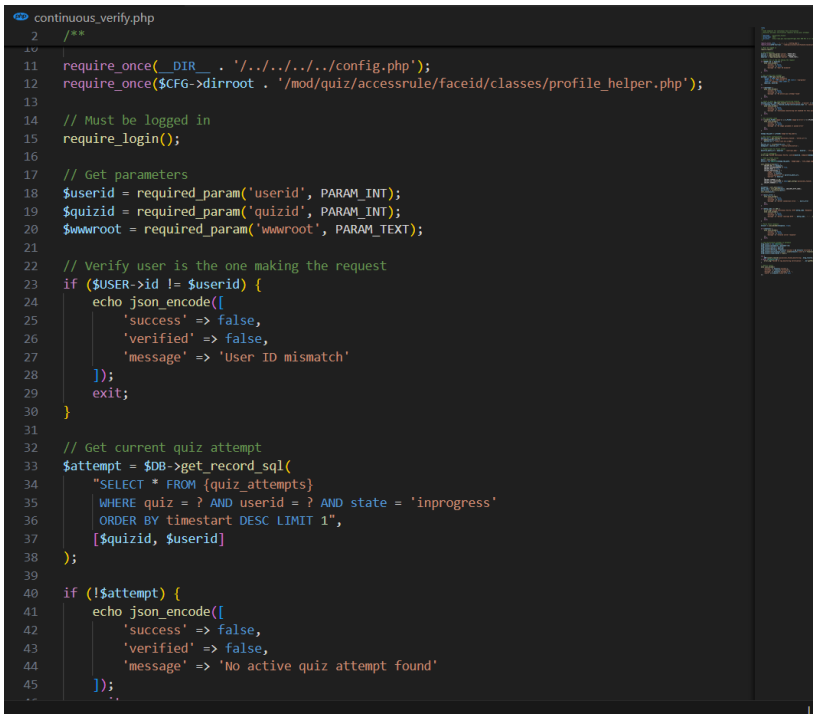


Figura 45. Archivo continuous_verify

- Archivo version.php

Este archivo define los metadatos del plugin requerido por Moodle, incluyendo el identificador del componente, número de versión, requisitos de compatibilidad (Moodle 3.9+), nivel de madurez del desarrollo y versión de lanzamiento.

```php
<?php
defined('MOODLE_INTERNAL') || die();

$plugin->component = 'quizaccess_faceid';
$plugin->version   = 2025091201;
$plugin->requires  = 2020061500; // Moodle 3.9 o superior
$plugin->maturity  = MATURITY_ALPHA;
$plugin->release   = 'v0.14';

```

Figura 46. Archivo de metadatos del plugin

4.1.8. Pruebas del sistema

En este apartado se presentan las pruebas realizadas al sistema de verificación facial, evaluando su funcionamiento integrado entre el plugin de Moodle y el servidor Flask. Las pruebas se ejecutaron en un entorno controlado para validar la correcta comunicación, el desempeño y el acceso seguro al cuestionario virtual.

4.1.8.1. Arquitectura del entorno de pruebas

La arquitectura del entorno de pruebas se implementó utilizando servicios en la nube de Amazon Web Services (AWS). Se desplegaron dos servidores independientes: uno para alojar Moodle y el plugin de verificación facial, y otro para ejecutar el servidor Flask encargado del procesamiento biométrico. Esta separación en instancias EC2 permitió simular un entorno distribuido, evaluar la comunicación entre los componentes y validar el funcionamiento del sistema en condiciones similares a un escenario real de uso.

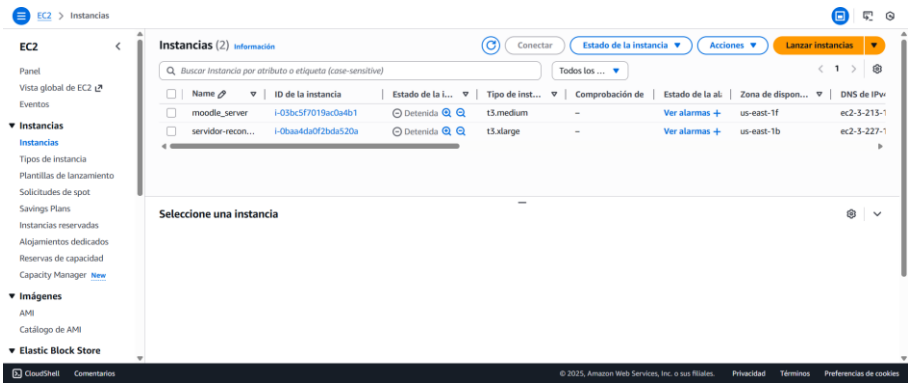


Figura 47. Instancias EC2 utilizadas para el entorno de pruebas del sistema de reconocimiento facial

Para asegurar que el servidor Flask se inicie automáticamente cada vez que la instancia AWS se encienda, se creó un servicio personalizado en *systemd* llamado face-recognition.service. Este servicio ejecuta el entorno virtual de Python y levanta el archivo principal del backend (face3_corrected.py) en el puerto 5001.

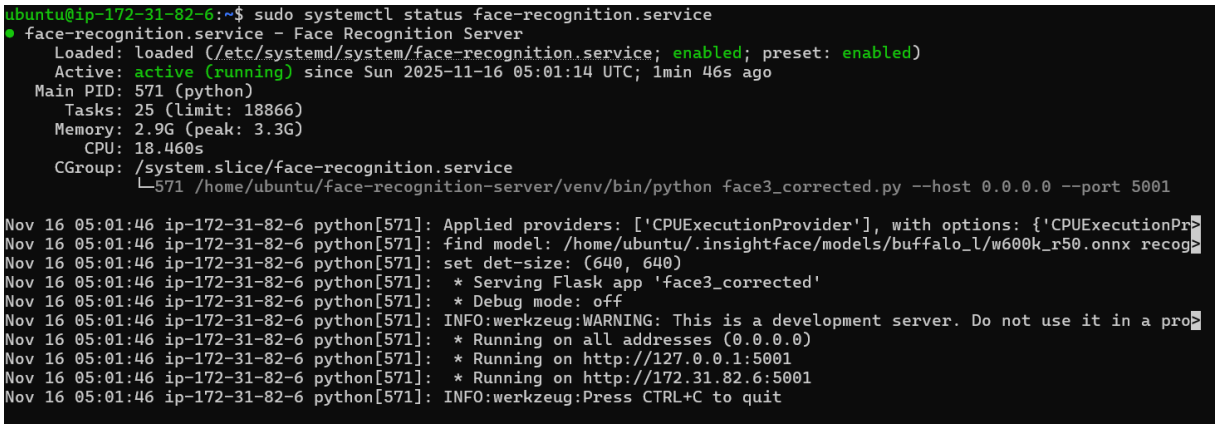


Figura 48. Ejecución del servicio systemd para iniciar automáticamente el servidor Flask

4.1.8.2. Pruebas funcionales del sistema

- Prueba verificación de perfil

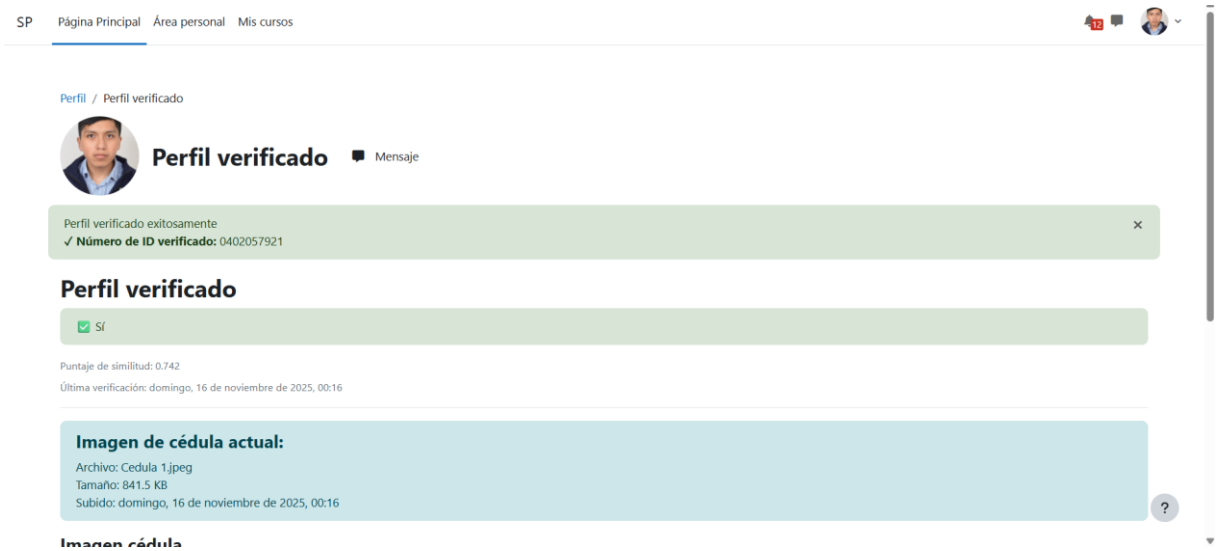


Figura 49. Resultado exitoso del proceso de verificación facial en el módulo de perfil de Moodle

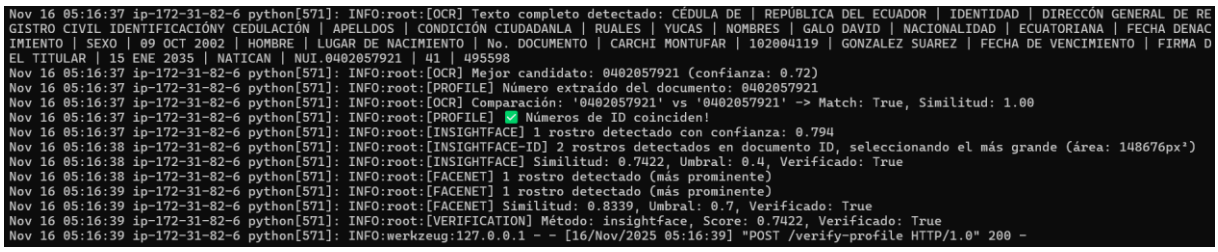


Figura 50. Registros del servidor Flask durante el proceso de verificación de perfil

- Prueba cámara

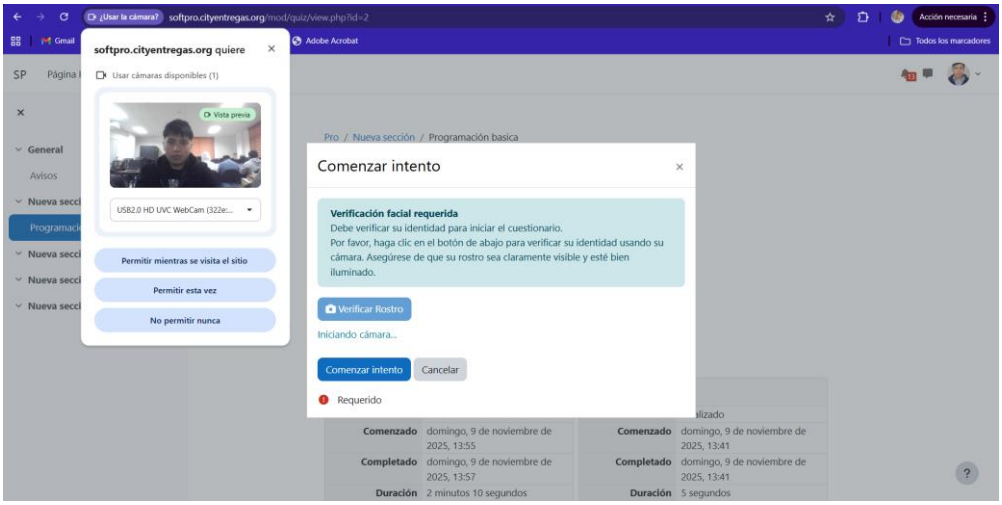


Figura 51. Solicitud de acceso a la cámara para la verificación facial antes de iniciar el cuestionario

- Prueba de verificación fallida al iniciar un cuestionario

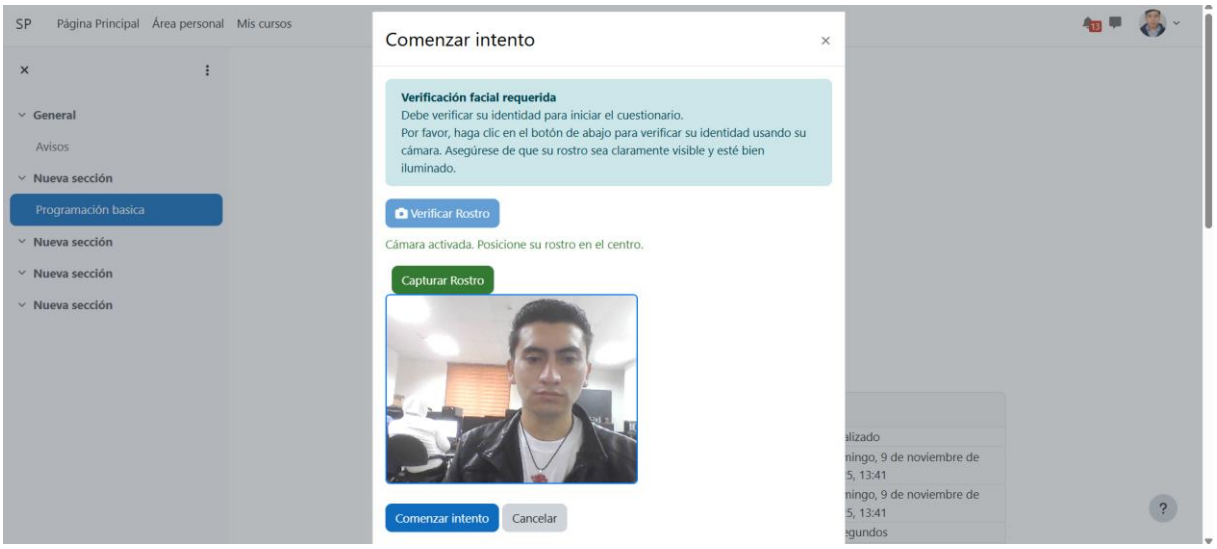


Figura 52. Captura previa del usuario no coincidente antes de intentar acceder

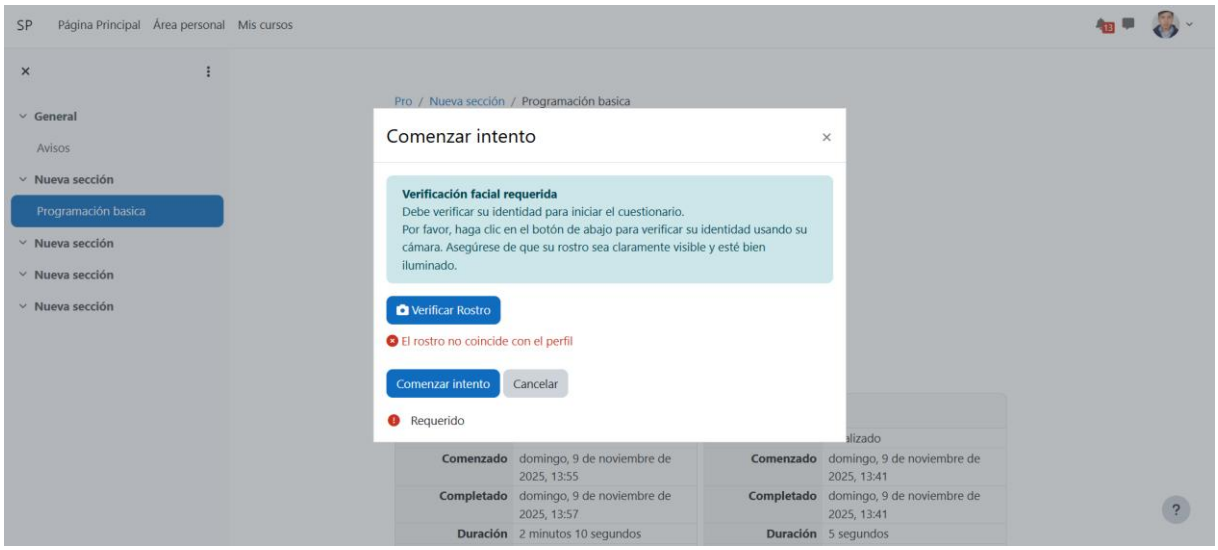


Figura 53. Mensaje en Moodle: "El rostro no coincide con el perfil"

```
Nov 17 13:38:01 ip-172-31-82-6 python[569]: INFO:root:[SERVER] Verificando usuario 3 - SISTEMA CORREGIDO
Nov 17 13:38:01 ip-172-31-82-6 python[569]: INFO:root:[SERVER] Usuario tiene número de ID registrado: 0402057921
Nov 17 13:38:01 ip-172-31-82-6 python[569]: INFO:root:[SPOOFING] Imagen cargada: (240, 320, 3)
Nov 17 13:38:01 ip-172-31-82-6 python[569]: INFO:root:[SPOOFING] Resultado: Real Face, Score: 0.997
Nov 17 13:38:01 ip-172-31-82-6 python[569]: INFO:root:[SERVER] ✓ Anti-spoofing aprobado
Nov 17 13:38:02 ip-172-31-82-6 python[569]: INFO:root:[INSIGHTFACE] 1 rostro detectado con confianza: 0.794
Nov 17 13:38:03 ip-172-31-82-6 python[569]: INFO:root:[INSIGHTFACE] 1 rostro detectado con confianza: 0.853
Nov 17 13:38:03 ip-172-31-82-6 python[569]: INFO:root:[INSIGHTFACE] Similitud: 0.5345, Umbral: 0.4, Verificado: True
Nov 17 13:38:03 ip-172-31-82-6 python[569]: INFO:root:[FACENET] 1 rostro detectado (más prominente)
Nov 17 13:38:03 ip-172-31-82-6 python[569]: INFO:root:[FACENET] 1 rostro detectado (más prominente)
Nov 17 13:38:03 ip-172-31-82-6 python[569]: INFO:root:[FACENET] Similitud: 0.6491, Umbral: 0.7, Verificado: False
Nov 17 13:38:03 ip-172-31-82-6 python[569]: INFO:root:[VERIFICATION] Método: insightface, Score: 0.5345, Verificado: False
Nov 17 13:38:03 ip-172-31-82-6 python[569]: INFO:werkzeug:127.0.0.1 - - [17/Nov/2025 13:38:03] "POST /verify HTTP/1.0" 200 -
```

Figura 54. Registros del servidor Flask mostrando similitud insuficiente

- Prueba de detección de múltiples personas

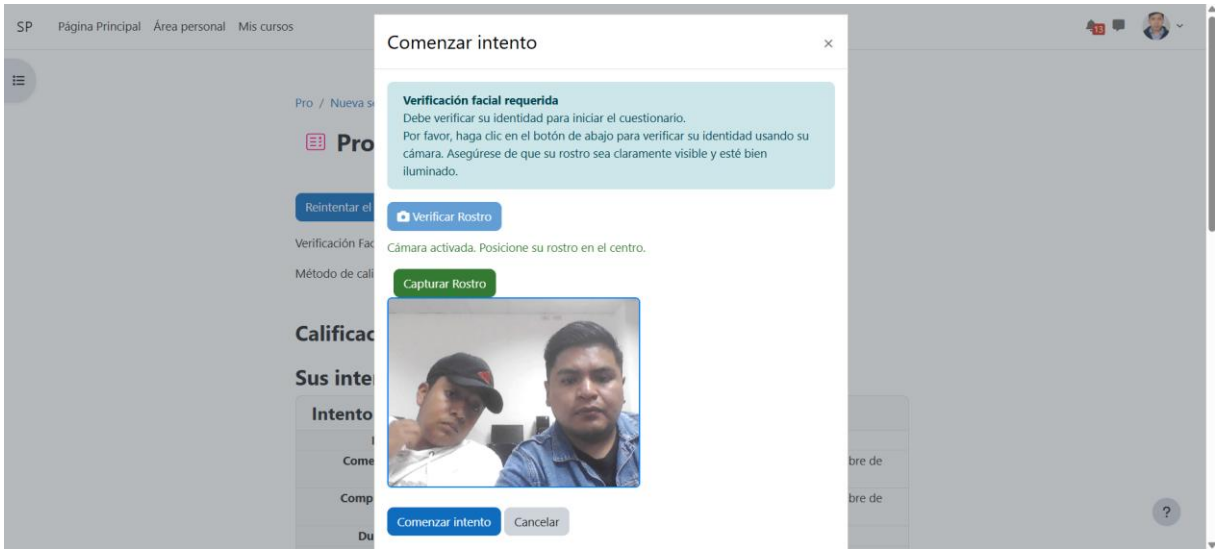


Figura 55. Imagen que contiene a 2 usuarios en el encuadre

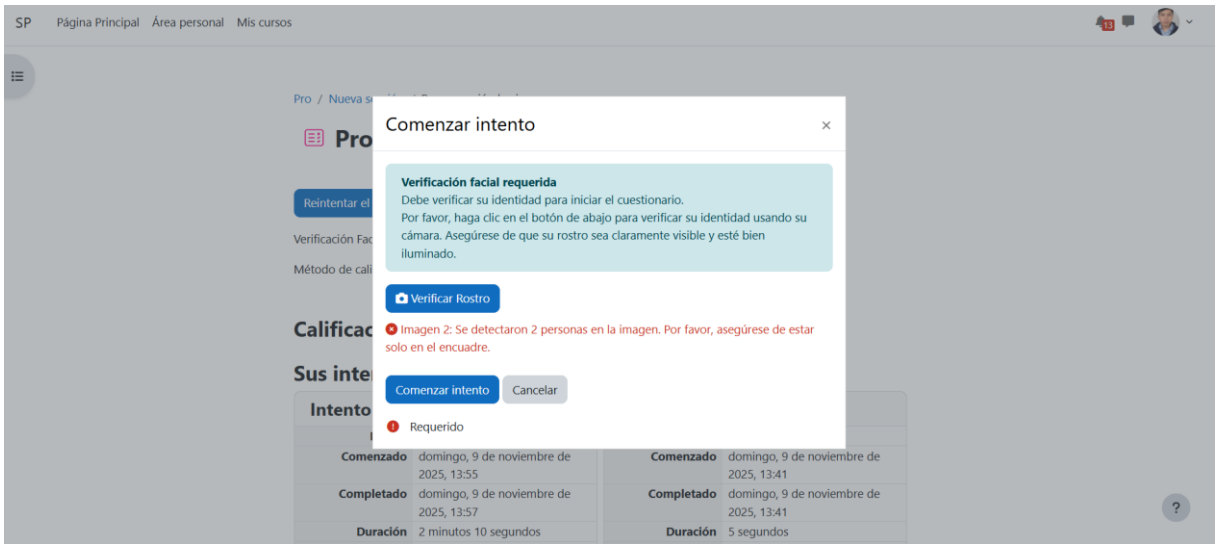


Figura 56. Imagen donde Moodle muestra: "Se detectaron 2 personas en la imagen"

```
Nov 17 14:47:35 ip-172-31-82-6 python[569]: INFO:root:[SERVER] Verificando usuario 3 - SISTEMA CORREGIDO
Nov 17 14:47:35 ip-172-31-82-6 python[569]: INFO:root:[SERVER] Usuario tiene número de ID registrado: 0402057921
Nov 17 14:47:35 ip-172-31-82-6 python[569]: INFO:root:[SPOOFING] Imagen cargada: (240, 320, 3)
Nov 17 14:47:35 ip-172-31-82-6 python[569]: INFO:root:[SPOOFING] Resultado: Real Face, Score: 0.532
Nov 17 14:47:35 ip-172-31-82-6 python[569]: INFO:root:[SERVER] ✓ Anti-spoofing aprobado
Nov 17 14:47:35 ip-172-31-82-6 python[569]: INFO:root:[INSIGHTFACE] 1 rostro detectado con confianza: 0.794
Nov 17 14:47:36 ip-172-31-82-6 python[569]: ERROR:root:[SECURITY] InsightFace detectó 2 rostros en la imagen
Nov 17 14:47:36 ip-172-31-82-6 python[569]: WARNING:root:[VERIFICATION] Fallo en imagen 2: Se detectaron 2 personas en la imagen. Por favor, asegúrese de es
tar solo en el encuadre.
Nov 17 14:47:36 ip-172-31-82-6 python[569]: WARNING:root:[SERVER] Verificación falló para usuario 3: Imagen 2: Se detectaron 2 personas en la imagen. Por fa
vor, asegúrese de estar solo en el encuadre.
```

Figura 57. Logs donde aparece: se detectaron 2 rostros

- Prueba detección de rostro falso

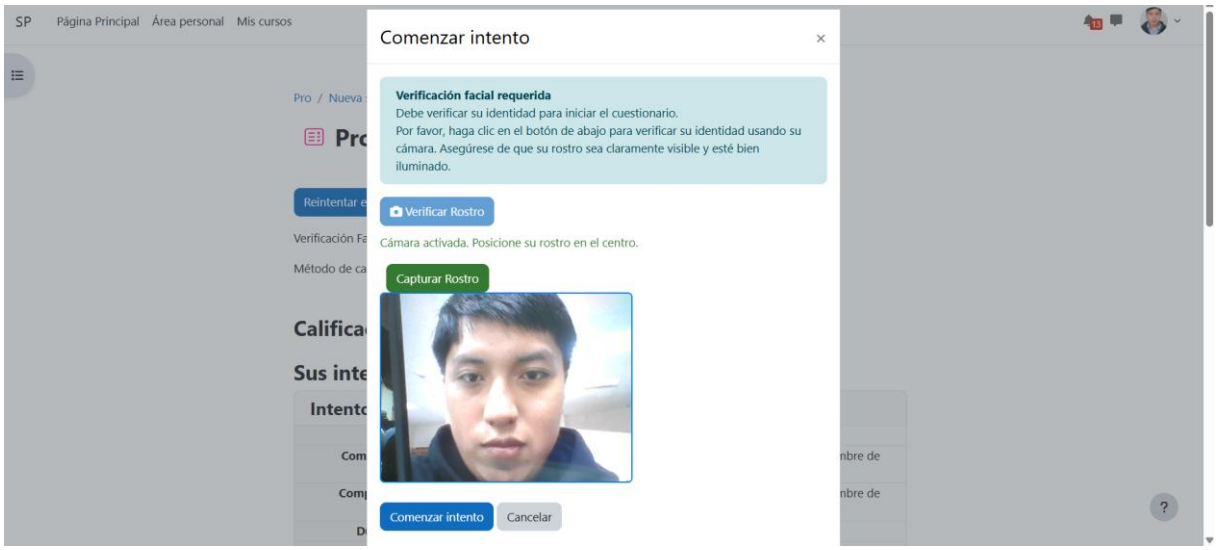


Figura 58. Imagen que contiene rostro falso en el encuadre

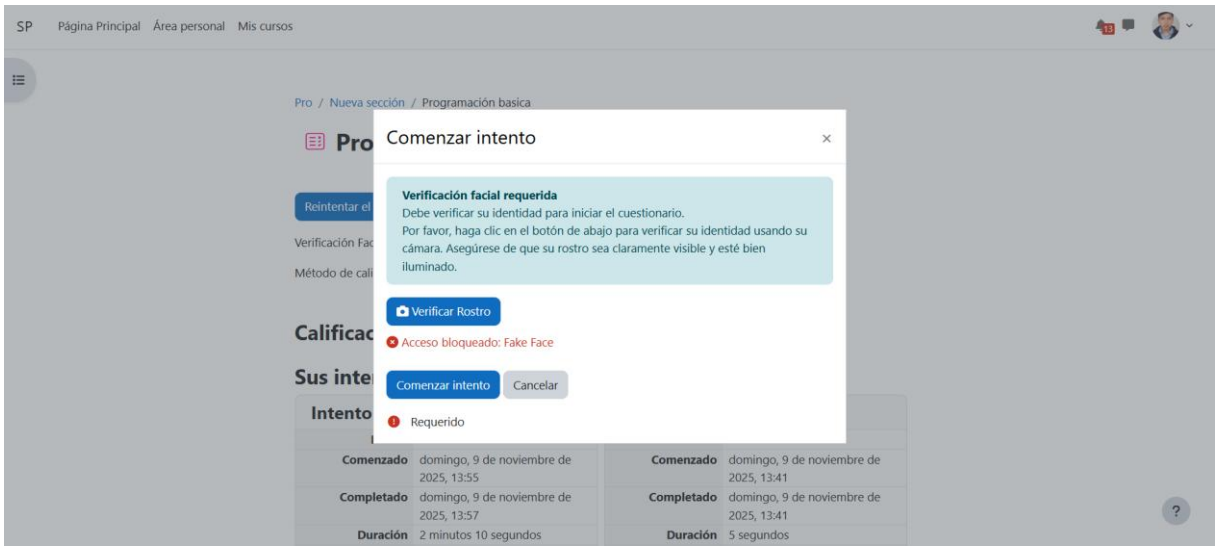


Figura 59. Imagen donde Moodle muestra: "Acceso bloqueado: Fake Face"

```
Nov 17 14:06:21 ip-172-31-82-6 python[569]: INFO:root:[SERVER] Verificando usuario 3 - SISTEMA CORREGIDO
Nov 17 14:06:21 ip-172-31-82-6 python[569]: INFO:root:[SERVER] Usuario tiene número de ID registrado: 0402057921
Nov 17 14:06:21 ip-172-31-82-6 python[569]: INFO:root:[SPOOFING] Imagen cargada: (240, 320, 3)
Nov 17 14:06:21 ip-172-31-82-6 python[569]: INFO:root:[SPOOFING] Resultado: Fake Face, Score: 1.000
Nov 17 14:06:21 ip-172-31-82-6 python[569]: INFO:werkzeug:127.0.0.1 - - [17/Nov/2025 14:06:21] "POST /verify HTTP/1.0" 200 -
```

Figura 60. Logs del servidor con: "Resultado: Fake Face, Score: 1.000"

- Pruebas de verificación exitosa al iniciar un cuestionario

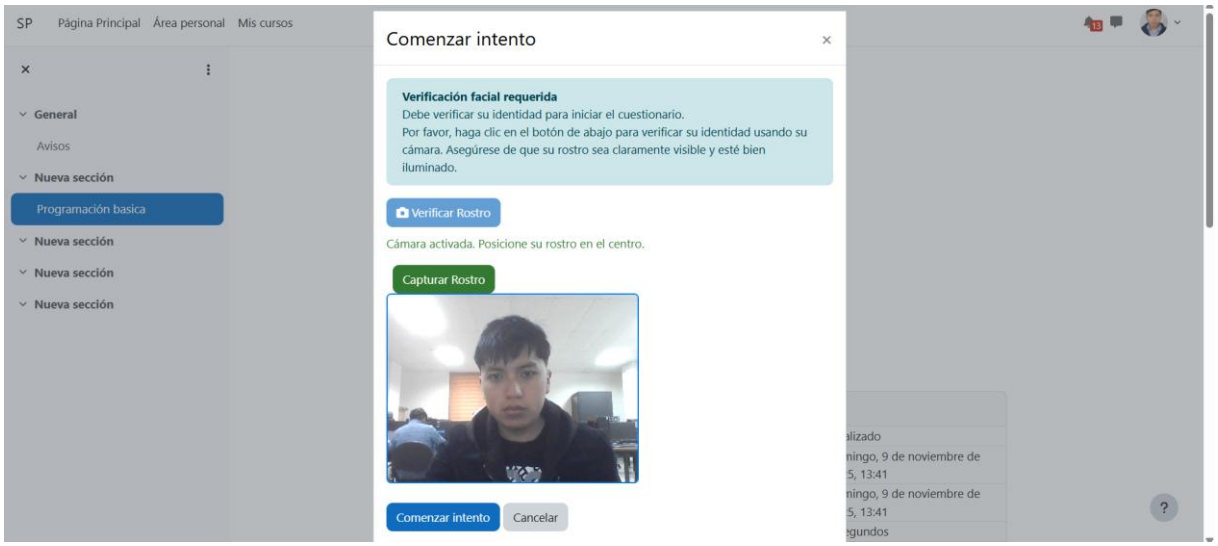


Figura 61. Imagen donde el usuario captura su rostro

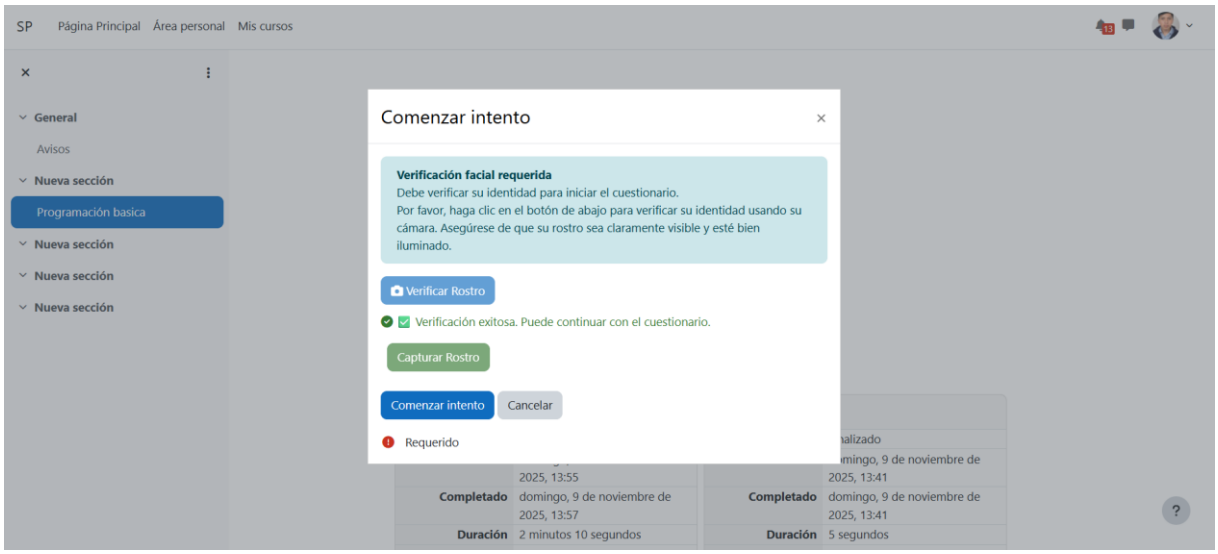


Figura 62. Imagen donde Moodle muestra "Verificación exitosa"

```
Nov 17 13:59:57 ip-172-31-82-6 python[569]: INFO:werkzeug:127.0.0.1 - - [17/Nov/2025 13:59:57] "POST /verify HTTP/1.0" 200 -
Nov 17 14:00:05 ip-172-31-82-6 python[569]: INFO:root:[SERVER] Verificando usuario 3 - SISTEMA CORREGIDO
Nov 17 14:00:05 ip-172-31-82-6 python[569]: INFO:root:[SERVER] Usuario tiene número de ID registrado: 0402057921
Nov 17 14:00:05 ip-172-31-82-6 python[569]: INFO:root:[SPOOFING] Imagen cargada: (240, 320, 3)
Nov 17 14:00:05 ip-172-31-82-6 python[569]: INFO:root:[SPOOFING] Resultado: Real Face, Score: 1.000
Nov 17 14:00:05 ip-172-31-82-6 python[569]: INFO:root:[SERVER] ✓ Anti-spoofing aprobado
Nov 17 14:00:06 ip-172-31-82-6 python[569]: INFO:root:[INSIGHTFACE] 1 rostro detectado con confianza: 0.794
Nov 17 14:00:06 ip-172-31-82-6 python[569]: INFO:root:[INSIGHTFACE] 1 rostro detectado con confianza: 0.805
Nov 17 14:00:06 ip-172-31-82-6 python[569]: INFO:root:[INSIGHTFACE] Similitud: 0.7666, Umbral: 0.4, Verificado: True
Nov 17 14:00:06 ip-172-31-82-6 python[569]: INFO:root:[FACENET] 1 rostro detectado (más prominente)
Nov 17 14:00:06 ip-172-31-82-6 python[569]: INFO:root:[FACENET] 1 rostro detectado (más prominente)
Nov 17 14:00:06 ip-172-31-82-6 python[569]: INFO:root:[FACENET] Similitud: 0.8080, Umbral: 0.7, Verificado: True
Nov 17 14:00:06 ip-172-31-82-6 python[569]: INFO:root:[VERIFICATION] Método: insightface, Score: 0.7666, Verificado: True
Nov 17 14:00:06 ip-172-31-82-6 python[569]: INFO:werkzeug:127.0.0.1 - - [17/Nov/2025 14:00:06] "POST /verify HTTP/1.0" 200 -
```

Figura 63. Logs correspondientes donde InsightFace y FaceNet confirman semejanza correcta

- Pruebas monitoreo continuo cada 5 minutos

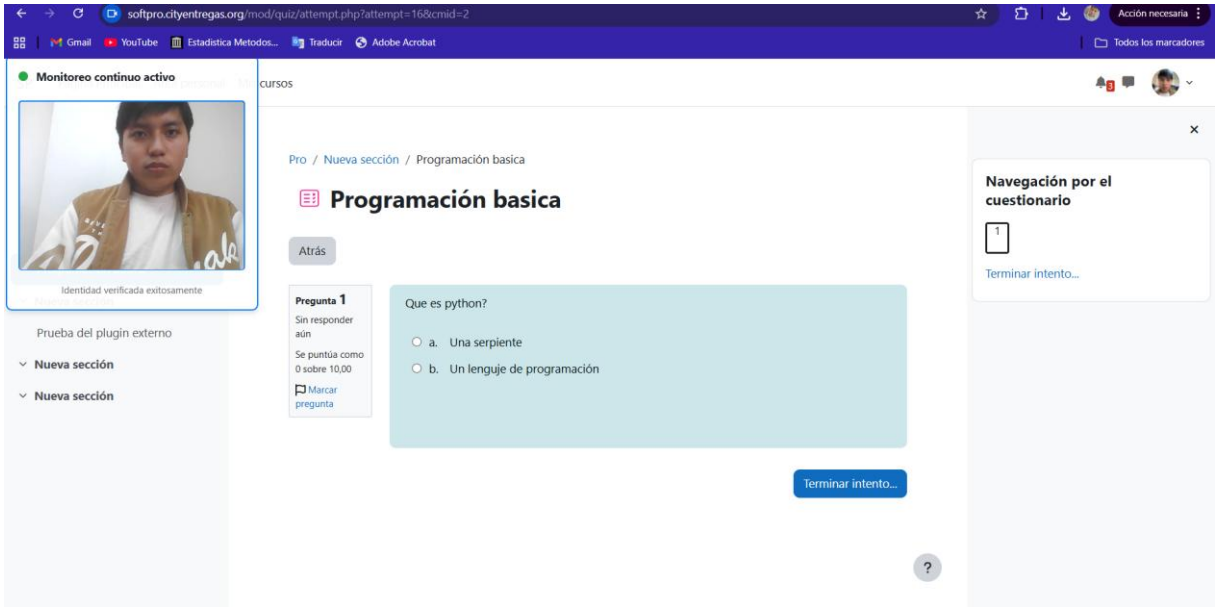


Figura 64. Monitoreo continuo verificación exitosa

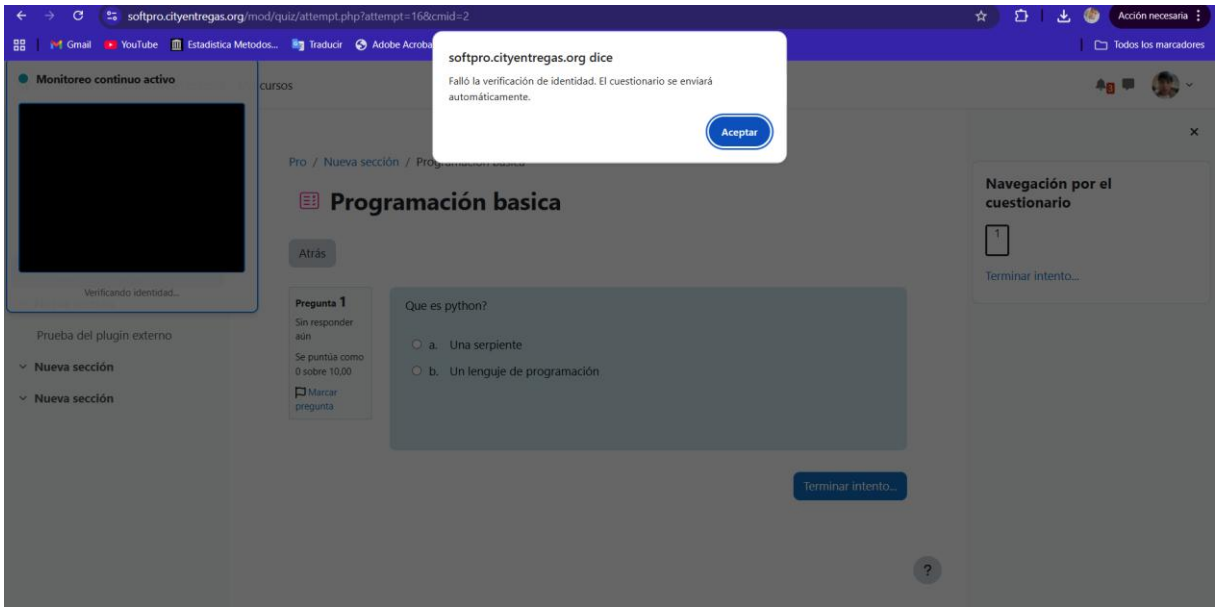


Figura 65. Monitoreo continuo verificación fallida

4.2. DISCUSIÓN

El sistema desarrollado en esta investigación integra tecnologías de reconocimiento facial para verificar la identidad de estudiantes en exámenes en línea. La solución combina InsightFace para el reconocimiento, RetinaFace para la detección facial, MiniFASNet para prevenir fraudes con fotografías, y se integra directamente con Moodle mediante un plugin. La presente propuesta aborda limitaciones de estudios previos de tres formas. Primero, incluye detección anti-spoofing para evitar fraudes con imágenes, respondiendo al problema identificado por Bergmans et al. (2021) donde sistemas comerciales fueron engañados con fotografías. Segundo, se integra completamente con Moodle, eliminando la necesidad de navegar entre plataformas diferentes. Tercero, usa modelos de aprendizaje profundo que funcionan mejor en condiciones variadas como las de los hogares estudiantiles. La arquitectura cliente-servidor es otra ventaja importante. Castro et al. (2024) documentaron que 25% de estudiantes en países en desarrollo no tienen equipos potentes. Al procesar todo en el servidor institucional, el sistema solo requiere que el estudiante tenga una cámara web y conexión a internet, democratizando el acceso. Los resultados de las encuestas revelaron información valiosa. El 47.1% de estudiantes conoce casos de suplantación de identidad, confirmando que el problema es real. El 69.8% confía en que el reconocimiento facial puede prevenir fraudes. Sin embargo, el 40.4% teme que el sistema no los reconozca correctamente, indicando que la principal preocupación es técnica, no ética. Solo el 25.4% expresó preocupación por invasión de privacidad.

Esta distribución de preocupaciones es diferente a lo reportado en otros países. En Estados Unidos y Europa, las objeciones éticas dominan, pero en este estudio las preocupaciones son principalmente operativas. Esto sugiere que, si el sistema funciona bien, será aceptado por los estudiantes. Los estudiantes prefieren una implementación gradual. El 39% indicó que el sistema debería usarse primero en exámenes finales o de grado, no en todas las evaluaciones. Esto coincide con las recomendaciones de García-Peñalvo et al. (2021) sobre adopción progresiva de tecnologías educativas. El sistema debe verse como parte de una estrategia más amplia de integridad académica, no como una solución aislada.

V. CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

- Se desarrolló un marco conceptual mediante revisión sistemática PRISMA que identificó las principales herramientas tecnológicas de reconocimiento facial en contextos educativos, incluyendo FaceNet, VGG-Face, OpenCV, MTCNN y RetinaFace, estableciendo una base teórica sólida sobre verificación de identidad en entornos virtuales.
- La revisión sistemática permitió identificar desafíos técnicos, éticos, sociales y legales de los sistemas de reconocimiento facial en educación, incluyendo imprecisión bajo condiciones no controladas, sesgos algorítmicos, brecha tecnológica y ausencia de legislación específica.
- El uso de encuestas estructuradas con 213 estudiantes permitió cuantificar percepciones, revelando un Índice de Conocimiento del 76.83%, Índice de Prevalencia del Fraude del 47.1%, e Índice de Confianza en la Tecnología del 69.8%, validando la problemática y la aceptación de soluciones tecnológicas.
- La selección de tecnologías de aprendizaje profundo (InsightFace/ArcFace, RetinaFace, MiniFASNet) se justifica por su robustez ante condiciones variables en entornos domésticos con iluminación deficiente y equipamiento limitado.
- La arquitectura cliente-servidor desarrollada aborda la brecha tecnológica al centralizar el procesamiento en el servidor Flask institucional, reduciendo los requisitos del estudiante a cámara web y conexión a internet.

5.2. RECOMENDACIONES

- Es recomendable integrar el sistema con otras capas de seguridad como navegadores seguros y análisis de patrones de respuesta, ya que enfoques híbridos son más efectivos.
- Es recomendable implementar programa de capacitación para docentes y estudiantes sobre funcionamiento del sistema
- Es recomendable que los estudiantes registren su fotografía de perfil en condiciones óptimas de iluminación, con fondo neutro, rostro centrado y sin accesorios que obstruyan el rostro (lentes oscuros, gorras, mascarillas), ya que

la calidad de la imagen base determina directamente la precisión del reconocimiento facial durante los exámenes.

- Para la implementación institucional del sistema de verificación facial en entornos de producción sin incluir monitoreo, se recomienda que el servidor donde se desplegará el componente Flask de procesamiento biométrico cuente con las siguientes especificaciones de hardware recomendadas: procesador Intel Xeon o AMD EPYC con 8 núcleos físicos o superior, memoria RAM de 16 GB o superior, almacenamiento SSD de 256 GB o superior, y tarjeta de red de 1 Gbps o superior, garantizando así tiempo de respuesta inferior a 2 segundos por verificación facial y capacidad de procesamiento para hasta 50 usuarios concurrentes.
- Para la implementación institucional del sistema de verificación facial en entornos de producción incluyendo monitoreo automático cada 5 minutos, se recomienda que el servidor donde se desplegará el componente Flask de procesamiento biométrico cuente con las siguientes especificaciones de hardware recomendadas: procesador Intel Xeon o AMD EPYC con 12 núcleos físicos o superior, memoria RAM de 24 GB o superior, almacenamiento SSD de 512 GB o superior (distribuyendo 256 GB para el sistema y aplicación, y 256 GB para logs, métricas históricas y datos de monitoreo), y tarjeta de red de 1 Gbps o superior, garantizando así tiempo de respuesta inferior a 2 segundos por verificación facial.

VI. REFERENCIAS BIBLIOGRÁFICAS

Admira, D. A., & Arnesia, D. (2021). Implementation of Face Recognition for Patient Identification Using the Transfer Learning Method. TEM JOURNAL - Technology, Education, Management, Informatics

Agencia Española de Protección de Datos (AEPD). (2025). Informe sobre el uso de sistemas de reconocimiento facial en empresas. Madrid: AEPD. https://www.aepd.es/prensa-y-comunicacion/notas-de-prensa/AEPD-informe-sistemas-reconocimiento-facial-empresas-seguridad-privada

Beraún Barrantes, J. G. (2021). Sistema de reconocimiento facial en línea para prevenir la suplantación y el plagio en el examen de admisión virtual en la Universidad de Huánuco 2020. *Repositorio Institucional UDH*. Sistema de reconocimiento facial en línea para prevenir la suplantación y el plagio en el examen de admisión virtual en la Universidad de Huánuco 2020

Bergmans, L., Haagen, M., & De Vos, T. (2021). Automated proctoring under scrutiny: A controlled cheating experiment. *Journal of Educational Technology & Society*, 24(3), 10-21. https://www.j-ets.net/collection/published-issues/24_3

Buolamwini, J., & Gebru, T. (2018). Gender shades: Intersectional accuracy disparities in commercial gender classification. *Proceedings of Machine Learning Research*, 81, 1-15. https://proceedings.mlr.press/v81/buolamwini18a.html

Castro, C., Silva, M., & Fernández, L. (2024). Brecha digital y acceso tecnológico en educación superior latinoamericana. *Revista de Educación y Tecnología*. https://tecnologia-educativa.com.ar/index.php/RET

Coronel Teanga, J. A. (2021). Sistema Inteligente de identificación facial para registro de asistencia estudiantil en la Universidad Ecotec. *Repositorio Digital ECOTEC*. https://repositorio.ecotec.edu.ec/handle/123456789/225

EDUCAUSE. (2022). Accessibility and proctoring: Principles and recommendations. *EDUCAUSE Learning Initiative*. https://library.educause.edu/resources/2022/3/accessibility-and-proctoring-principles-and-recommendations

Fernández, M., Rivas, L., & Romero, D. (2023). Desigualdad digital en la educación remota latinoamericana. *Educación y Sociedad*. https://periodicos.sbu.unicamp.br/ojs/index.php/vitis

Galindo Taype, J., Huaringa Gallardo, R., & Samaniego Canales, E. (2021). Reconocimiento facial para la identificación de los alumnos en exámenes finales. *Repositorio Institucional Universidad Continental*. https://repositorio.continental.edu.pe/handle/20.500.12394/10419

García-Peñalvo, F. J., Corell, A., Rivero-Ortega, R., & Rodríguez-Conde, M. J. (2021). Impact of the COVID-19 on higher education: An experience-based approach. *Information Technologies and Learning Tools*. https://journal.iitta.gov.ua/index.php/itlt/article/view/4507

Guerrero, J. (2021). Identidad y vigilancia digital en la universidad. *Repositorio Institucional UNAM*. https://repositorio.unam.mx/

Guerrero-Roldán, A. E., & Noguera, I. (2021). A model for aligning assessment with competences and learning activities in online courses. *The Internet and Higher Education*. https://doi.org/10.1016/j.iheduc.2021.100820

JaidedAI. (2020). EasyOCR: Ready-to-use OCR with 80+ languages supported. *GitHub*. https://github.com/JaidedAI/EasyOCR

Karim, R. A., Adnan, M., & Rahman, M. (2023). Student perceptions of online examination integrity in higher education. *International Journal of Educational Technology in Higher Education*. https://educationaltechnologyjournal.springeropen.com/articles/10.1186/s41239-022-00375-w

Labayen, M., Vea, R., Flórez, J., Aginako, N., & Sierra, B. (2021). Online student authentication and proctoring system based on multimodal biometrics technology. *IEEE Access*. https://ieeexplore.ieee.org/document/9420284

Lai, J. W., & Bower, M. (2021). How is the use of technology in education evaluated? A systematic review. *Computers & Education*. https://doi.org/10.1016/j.compedu.2019.03.009

Lee, M., & Shah, S. (2022). The trans exclusion of facial recognition systems in online education. *Gender & Technology Review*. https://pitt.libguides.com/lgbtq-studies/journals

Legarda, E., & Loaiza, M. (2022). Sistema de reconocimiento facial mediante aprendizaje profundo para control de asistencia universitaria. *Revista Tecnológica ESPOL*. https://www.rte.espol.edu.ec/index.php/tecnologica/article/view/893

Li, Y., & Jain, A. K. (2022). Beyond accuracy: Behavioral testing of face recognition systems. *IEEE Transactions on Biometrics, Behavior, and Identity Science*. https://ieeexplore.ieee.org/document/9693444

Manas, A. (2021). La vigilancia algorítmica en el aula. *Revista de Educación Crítica*. https://revistaeducacioncritica.org/

Marrugo Cogollo, J., & Castro Flórez, C. (2022). Sistema de reconocimiento facial para la gestión y el seguimiento de estudiantes ausentes (SEFAD). *Repositorio Universidad Piloto*. https://repository.unipiloto.edu.co/handle/20.500.12277/11396

Mendoza Nina, A. (2024). Sistema informático con reconocimiento facial para mejorar el control biométrico. *Repositorio Institucional UNAP*. https://repositorio.unap.edu.pe/handle/20.500.14082/23352

Moor, K., Silva, C., & Denning, T. (2023). Surveillance in the Zoom age. *Journal of Cybersecurity and Privacy*. https://www.mdpi.com/journal/jcp

Moubayed, A., Injadat, M., Shami, A., & Lutfiyya, H. (2020). Student engagement level in e-learning environment. *American Journal of Distance Education*. https://doi.org/10.1080/08923647.2020.1696140

Ozdamli, F., Aljarrah, A., Karagozlu, D., & Ababneh, M. (2022). Facial recognition system to detect student emotions and cheating in distance learning. *Sustainability*. https://doi.org/10.3390/su142013230

Palmett, A. M. (2020). Métodos inductivo, deductivo y teoría de la pedagogía crítica. *Petroglifos Revista Crítica Transdisciplinar*. https://dialnet.unirioja.es/servlet/articulo?codigo=7475143

Patiño, D. (2021). El rostro como frontera: Estudiantes y resistencia al reconocimiento facial. *Revista de Estudios Latinoamericanos*. https://www.cialc.unam.mx/reala/

Potluri, A., Dutta, S., & Chaudhuri, A. (2023). Attentive: An Intelligent Face-Based Monitoring System for Online Examinations. *Educational Data Mining*. https://educationaldatamining.org/edm2023/proceedings/

Resha, C., Kumar, V., & Singh, A. (2023). Multimodal authentication system for online examination proctoring. *Journal of Educational Computing Research*. https://journals.sagepub.com/home/jec

Rivarola, P., & Ortega, M. (2022). Evaluación digital inclusiva: Alternativas al proctoring. *Educación y Desarrollo Social*. https://revistas.unimilitar.edu.co/index.php/reds

Rodríguez, A., & Miller, T. (2022). When identity fails: Racial bias and academic penalties in AI proctoring. *Race and Technology Studies*. https://www.journals.uchicago.edu/toc/jla/current

Rodríguez, M., & Pérez, J. (2021). Métodos científicos de indagación y de construcción del conocimiento. *Revista EAN*. https://doi.org/10.21158/01208160.n82.2017.1647

Sakhipov, A., Ospanov, B., & Nurgaliyev, K. (2025). Implementation of facial recognition systems in Central Asian universities. *International Journal of Educational Technology*. https://link.springer.com/journal/41239

Segovia García, N. (2023). Aceptación de los exámenes supervisados a través de herramientas de e-proctoring. *Aloma*. https://doi.org/10.51698/aloma.2023.41.2.51-60

Shkodzinskyi, A., Sobolevskyi, D., & Koval, S. (2023). A comparative analysis of e-proctoring tools. *CEUR Workshop Proceedings*. https://ceur-ws.org/Vol-3382/paper1.pdf

Silva, R., & Cardoso, F. (2023). Reconhecimento facial e privacidade no ensino superior brasileiro. *Revista Direito & Tecnologia*. https://indexlaw.org/index.php/revistadireitotecnologia

Singh, A., Kumar, P., & Sharma, R. (2021). Digital transformation in higher education during COVID-19 pandemic. *International Journal of Information Management*. https://doi.org/10.1016/j.ijinfomgt.2021.102359

Singh, V., Sharma, N., & Patel, M. (2024). Active surveillance in online examinations using YOLOv5. *IEEE Transactions on Education*. https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=13

Ullah, F., Babar, M. A., & Shen, J. (2022). Security concerns in e-learning systems. *Journal of Computing in Higher Education*. https://doi.org/10.1007/s12528-021-09292-x

UNESCO. (2023). Global education monitoring report 2023: Technology in education. *UNESCO Publishing*. https://unesdoc.unesco.org/ark:/48223/pf0000385723

Villalobos Sánchez, B. S. (2025). Reconocimiento facial para prevenir suplantaciones en exámenes de admisión. *Repositorio UNTRM*. https://repositorio.untrm.edu.pe/handle/20.500.14077/4822

Yamada, F., Oliveira, P., & Silva, M. (2023). Implementação de proctoring facial em universidades públicas do Brasil. *Cadernos de Educação a Distância*. https://periodicos.uff.br/cedu

Yaman, M., & Koyuncu, İ. (2022). Academic integrity in online assessment. *Turkish Online Journal of Distance Education*. https://dergipark.org.tr/en/pub/tojde

Zhao, W., Chellappa, R., Phillips, P. J., & Rosenfeld, A. (2023). Face recognition: A literature survey. *ACM Computing Surveys*. https://dl.acm.org/journal/csur

VII. ANEXOS

Anexo 1. Certificado del abstract por parte de idiomas

UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI FOREIGN AND NATIVE LANGUAGES CENTER

ABSTRACT- EVALUATION SHEET

NAME: Galo David Ruales Yucas
DATE: Viernes, 16 de enero de 2026
Topic: "Reconocimiento facial para la verificación de identidad de estudiantes en exámenes en línea."

MARKS AWARDED **QUANTITATIVE AND QUALITATIVE**

VOCABULARY AND WORD USE	Use new learnt vocabulary and precise words related to the topic	Use a little new vocabulary and some appropriate words related to the topic	Use basic vocabulary and simplistic words related to the topic	Limited vocabulary and inadequate words related to the topic
	EXCELLENT: 2 ☐	GOOD: 1,5 ☐	AVERAGE: 1 ☐	LIMITED: 0,5 ☐
WRITING COHESION	Clear and logical progression of ideas and supporting paragraphs.	Adequate progression of ideas and supporting paragraphs.	Some progression of ideas and supporting paragraphs.	Inadequate ideas and supporting paragraphs.
De	EXCELLENT: 2 ☐	GOOD: 1,5 ☐	AVERAGE: 1 ☐	LIMITED: 0,5 ☐
ARGUMENT	The message has been communicated very well and identify the type of text	The message has been communicated appropriately and identify the type of text	Some of the message has been communicated and the type of text is little confusing	The message hasn't been communicated and the type of text is inadequate
	EXCELLENT: 2 ☐	GOOD: 1,5 ☐	AVERAGE: 1 ☐	LIMITED: 0,5 ☐
CREATIVITY	Outstanding flow of ideas and events	Good flow of ideas and events	Average flow of ideas and events	Poor flow of ideas and events
	EXCELLENT: 2 ☐	GOOD: 1,5 ☐	AVERAGE: 1 ☐	LIMITED: 0,5 ☐
SCIENTIFIC SUSTAINABILITY	Reasonable, specific and supportable opinion or thesis statement	Minor errors when supporting the thesis statement	Some errors when supporting the thesis statement	Lots of errors when supporting the thesis statement
	EXCELLENT: 2 ☐	GOOD: 1,5 ☐	AVERAGE: 1 ☐	LIMITED: 0,5 ☐
TOTAL/AVERAGE	9 - 10: EXCELLENT 7 - 8,9: GOOD 5 - 6,9: AVERAGE 0 - 4,9: LIMITED	TOTAL 9		

Anexo 2. Certificado de aprobación

EDUVirtual
UNIDAD DE TECNOLOGÍA EDUCATIVA

POLITÉCNICA
DEL CARCHI

CERTIFICO

Que el Señor **Galo David Ruales Yucas** con cédula de identificación N° **0402057921** estudiante de la carrera de Computación de la Universidad Politécnica Estatal del Carchi, trabajó en la Unidad de Tecnología Educativa en el desarrollo del proyecto de investigación **"Reconocimiento facial para la verificación de identidad de estudiantes en exámenes en línea"** en donde la Unidad ha brindado las facilidades para llevar a cabo la finalización del mismo.

La propuesta del proyecto que constituye el desarrollo un plugin para el LMS Moodle que permite el reconocimiento facial para la verificación de identidad de estudiante al momento de rendir una evaluación virtual, dicho trabajo ha sido socializado con el personal correspondiente el mismo contribuye alcanzar las metas propuestas dentro de la Unidad y a su vez mejorar en la calidad y seguridad de los exámenes en línea, por lo cual extendemos nuestros agradecimientos a la institución por los resultados obtenidos.

Es todo cuanto puedo certificar en honor a la verdad, facultando a el interesado hacer el uso del presente de forma que estime conveniente.

Dado y firmado en el cantón Tulcán, a los 15 días del mes de diciembre del año dos mil veinticinco.

Atentamente,

MSc. Marco Antonio Yandún Velastegui
Coordinador de la Unidad de tecnología educativa

Anexo 3. Repositorios código fuente del sistema

El código fuente completo del sistema de verificación facial desarrollado en esta investigación está disponible públicamente en los siguientes repositorios de GitHub:

Plugin Moodle - Quiz Access Rule FaceID

URL: https://github.com/Galo45/moodle-quizaccess-faceid-

Servidor Flask - Procesamiento Biométrico

URL: https://github.com/Galo45/faceid-flask-server-

Ambos repositorios incluyen documentación completa para facilitar la implementación y uso del sistema.