Servicio de Software Factory para los OEE en 4 Lotes - ID 410299

OT 12 - Desarrollo del Face Recognition Core a partir del Análisis de librerías de reconocimiento facial y prueba de vida

EVIDENCIAS DE CONTROL DE CALIDAD

CENTRO DE DESARROLLO SOSTENIBLE SA

09/AGOSTO/2024

ÍNDICE

1. INTRODUCCIÓN **3**
2. GLOSARIO DE TÉRMINOS **3**
3. FUNCIONALIDADES ENTREGADAS **4**
RF1 – Aplicación Móvil Integrada a la API 4
RF2 – Detección de rostro 4
RF3 – Identificación de rostro y aprendizaje del modelo 4
RF4 – Prueba de vida activa 4
RF5 – Autenticación de la API 4
RF6 – (OnBoarding) Enrolamiento y registro de huella facial 4
RF7 – Autenticación/Verificación de usuarios (userapps) 4
4. PRUEBAS DE ASEGURAMIENTO DE CALIDAD **4**
PRUEBAS FUNCIONALES 4
5. EVIDENCIA Y REPORTE DE EJECUCIÓN DE LAS PRUEBAS **4**
PRUEBAS FUNCIONALES 4
6. CORRECCIONES Y PRUEBAS REALIZADAS EN ETAPA DE VERIFICACIÓN Y CICLO DE PRUEBAS **25**
7. HISTORIAL DE VERSIONES **31**

1. *INTRODUCCIÓN*

En el marco de los servicios de Software Factory para los OEE en 4 Lotes - ID 410299, Orden de Trabajo (OT) Nro 12, titulada *"Desarrollo del Face Recognition Core a partir del Análisis de librerías de reconocimiento facial y prueba de vida"*, se llevaron a cabo las pruebas y control de calidad de las funcionalidades desarrolladas.

Este documento presenta las evidencias de pruebas y control de calidad llevadas a cabo para la OT Nro 12, e incluye en las siguientes secciones: Glosario de términos, Funcionalidades, Pruebas de Aseguramiento de Calidad, Evidencia y Reporte de Ejecución de las Pruebas y Conclusiones.

2. *GLOSARIO DE TÉRMINOS*

- **OEE**: Organismos y Entidades del Estado.
- **SII**: Sistema de Intercambio de Información, plataforma de interoperabilidad, que actúa como canal entre las instituciones y organizaciones del Estado para compartir información.
- **MITIC**: Ministerio de Tecnologías de la Información y Comunicación.
- Usuarios: Aquellos que posean credenciales dentro de la plataforma y según sus permisos, pueden realizar consultas de datos.
- **API**: Interfaz de Programación de Aplicaciones, es un conjunto de funciones y procedimientos que permite integrar sistemas, permitiendo que sus funcionalidades puedan ser reutilizadas por otras aplicaciones o software.
- **Face recognition**: El reconocimiento facial es una tecnología potencialmente capaz de comparar un rostro humano a partir de una imagen digital contra una base de datos de rostros e identificar a una persona.
- **Huella facial**: Es el código numérico al cual se convierte un rostro luego de aplicarse una fórmula matemática sobre el mismo.
- **OnBoarding (digital)**: es el proceso de identificación no presencial, que ayuda a las entidades a agilizar la incorporación de nuevos clientes y a los usuarios a realizar los trámites desde cualquier lugar. En el onboarding se vinculan clientes nuevos o antiguos a los servicios de manera 100% digital.
- **Porcentaje de similitud**: Indica que tan probable es que la persona en la foto tipo selfie corresponda con el número de cédula de la persona registrada en el Departamento de Identificaciones de la Policía Nacional u otra fuente utilizada.
- **Aplicación cliente**: Es la aplicación que se conecta y utiliza los servicios de la API, en este caso la FaceRecognitionCore, un ejemplo de aplicación cliente es PortalPY.
- **Userapps**: Son los usuarios que las aplicaciones clientes dan de alta mediante funciones para realizar el OnBoarding (o enrolamiento). Son el resultado de un proceso de OnBoarding exitoso

3. *FUNCIONALIDADES ENTREGADAS*

RF1 – Aplicación Móvil Integrada a la API
RF2 – Detección de rostro
RF3 – Identificación de rostro y aprendizaje del modelo
RF4 – Prueba de vida activa
RF5 – Autenticación de la API
RF6 – (OnBoarding) Enrolamiento y registro de huella facial
RF7 – Autenticación/Verificación de usuarios (userapps)

4. *PRUEBAS DE ASEGURAMIENTO DE CALIDAD*

El objetivo de las pruebas del sistema es asegurarse de que el sistema cumple con los requisitos funcionales.

Para la OT en cuestión se abarcaron las siguientes categorías de pruebas:

- Pruebas Funcionales

PRUEBAS FUNCIONALES

Las pruebas funcionales no se ocupan de cómo se produce el procesamiento, sino de si éste ofrece los resultados correctos o tiene algún fallo.

- Objetivo: Comprobar que el sistema funciona acorde a los requerimientos funcionales.
- Alcance: Se realizarán las pruebas de todos los requerimientos funcionales solicitados.

5. *EVIDENCIA Y REPORTE DE EJECUCIÓN DE LAS PRUEBAS*

PRUEBAS FUNCIONALES

Detalles de la Prueba:

- **Fecha de Ejecución:** del 6 al 9 de agosto de 2024
- **Herramienta:** Plantilla de casos de prueba
- **Ambiente de prueba:**
  - Backend docs: https://face-recognition.cds.com.py/docs
  - Websocket: ws://face-recognition.cds.com.py/ws

Contenido de la plantilla de casos de prueba:

- **Id:** Cada caso de prueba debe tener un identificador (código) único.
- **RF:** Requerimiento funcional al que pertenece el caso de prueba
- **Descripción:** Descripción del caso de prueba, indicando sus elementos, funcionalidades y acciones a ser ejercidas en el caso de prueba.
- **Área Funcional / Subproceso:** Describe el área funcional, subproceso o módulo al que está asociado el caso de prueba. La intención es poder dividir los casos de prueba según la estructura jerárquica y **casos de uso** del sistema que se está probando y los procesos que este soporta.
- **Precondiciones:** Condiciones que deben cumplirse para poder llevar a cabo las pruebas en el escenario esperado.
- **Datos de entradas:** Se especifica cada entrada que se requiere para ejecutar el caso de prueba. Estas entradas pueden ser valores o datos de entrada, y también acciones (por ejemplo presionar un botón). Deben identificarse los archivos o bases de datos involucrados.
- **Estado de Caso de Prueba:** Se indica si pasó o no con éxito la ejecución de la prueba llevada a cabo para verificación del caso. OK implica que pasó la ejecución de la prueba.
- **Observaciones:** Observaciones relacionadas con la ejecución de los casos y de los resultados que se obtuvieron u otros comentarios relacionados.

ID	RF	Descripción Caso de Prueba	Área funcional	Precondiciones	Datos/Acción de Entrada	Estado Caso	Observación
1	RF1	1. Permitir el enrolamiento de la persona, donde la misma deberá proporcionar su: a. Número Documento de Identidad (CI) b. Fecha de Nacimiento c. Correo Electrónico 2. En el enrolamiento se deberá capturar una secuencia de imágenes de la persona realizando una acción, que deberá ser enviada a la API para registrar la huella facial y prueba de vida de la persona 3. Permitir el ingreso de las personas previamente enroladas a la aplicación a través de la verificación o reconocimiento facial en el Face Recognition Core	Aplicación Móvil	Identidad electrónica del Portal Paraguay	CI=3636134 Fecha nac=16/09/93 Email=dantepane@cds.com.py	OK	**Imágenes 1 al 10.**
2	RF2	Desarrollar un método para detección de rostros que reciba una fotografía y retorne la misma fotografía con un cuadro dibujado alrededor del rostro detectado. 1. Análisis Facial: Captura y análisis de la imagen del rostro, identificando	API /face-recognition/api/v1/usuarios/detect-faces/			OK	Análisis facial: **Imagen 15** Conversión de imagen a dato: **Imagen 14**

		los puntos de referencia faciales que son clave para distinguir un rostro 2. Conversión de la imágen a dato: transformación de la información analógica (un rostro) en un conjunto de información digital (datos) basado en los rasgos faciales de la persona para proceder a registrar su “huella facial” digital					
3	RF3	Identificar a una persona retornando un porcentaje de similitud de la fotografía recibida en relación a una o varias fotografías previamente registradas de la persona, asociadas al número de documento de identidad recibido. 1. Iniciar un proceso de aprendizaje donde se obtengan una o varias fotografías previamente registradas de la persona, obtener su huella facial, y asociarlas al número de documento de identidad. 2. Las fuentes de datos para el aprendizaje deberán ser proveídas por el MITIC. Las alternativas indicadas por la institución son las	API			OK	**Imagen 10.** Prueba de vida exitosa con porcentaje de similitud Para el proceso de aprendizaje, el modelo entrenado se guarda en el MongoDB. La fuente de datos es: **https://sii.paraguay.gov.py/frontend-identificaciones/api/persona/obtenerPersonaPorNroCedulaConFoto/123456**

		siguientes: a. MITIC: Base de datos de rostros registrados por el MITIC al momento de la creación de una cuenta de identidad electrónica. b. Policía Nacional: Bases de datos de rostros de la Policía Nacional los cuales son relevados al momento de la renovación o expedición de un documento de identidad o pasaporte (Accesibles mediante el SII).					
4	RF4	Incluir funciones de “prueba de vida” considerando el conteo de pestañeo, para detectar actividad vital activa y garantizar que la persona/usuario está presente y vivo durante la autenticación o enrolamiento.	API			OK	**Imágenes 1 al 10.**
5	RF5	1. Contemplar un mecanismo de autenticación de la API basado en tokens de acceso para su protección ante accesos no autorizados. 2. El token deberá tener una validez para un periodo de tiempo	API /face-recognition/api/v1/login/access-token		username=mitic password=test	OK	Revisar la "Documentación Técnica y Guía de Uso de la API". El token dura 8 días.

		configurable, almacenado en variables de entorno. 3. El token estará relacionado a la aplicación que solicita y contendrá el identificador de aplicación.					Las aplicaciones solicitantes se guardan en la tabla **seguridad.aplicacion**
6	RF6	1. Desarrollar un método para realizar el proceso de onBoarding de los usuarios de las aplicaciones clientes de la API, que contemplen los siguientes datos: a. El número de documento de identidad policial como el identificador único b. Fecha de nacimiento c. Correo electrónico d. Secuencia de imágenes tipo selfie 2. Los usuarios estarán relacionados a una institución. El identificador de institución se obtiene a partir del token de autenticación. 3. Realizar la validación de la identificación del rostro para obtener el porcentaje de similitud, y la validación de prueba de vida.	API		CI=3636134 Fecha nac=16/09/93 Email=dantepane@cds.com.py	OK	Los datos de onboarding se registran en la tabla **seguridad.usuario** Los usuarios se relacionan a una aplicación, las aplicaciones se ven en la tabla **seguridad.aplicacion** Para porcentaje de similitud ver imagen: **Imagen 10.** La cantidad de operaciones de onboarding se registran en la columna **cantidad** de la tabla **seguridad.estadistica**

		4. Registrar los datos del usuario recibidos como parámetro: a. El porcentaje de similitud b. La fecha/hora del enrolamiento c. Un estado (Activo por defecto) 5. Registrar la cantidad de veces que se realizan operaciones de on-boarding por las aplicaciones clientes 6. Todas las acciones deberán ser auditadas en estructuras de auditoría					Las acciones se auditan en la tabla **seguridad.auditoria**
7	RF7	1. Desarrollar un método para identificar o confirmar la identidad de una persona. Considerar como parámetros de entrada: a. Número de cédula de identidad b. Secuencia de imágenes tipo selfie 2. Realizar la validación de la identificación del rostro para obtener el porcentaje de similitud, y la validación de prueba de vida.	API		CI=3636134	OK	**Imágenes 10 y 11.** La cantidad de operaciones de onboarding se registran en la columna **cantidad** de la tabla **seguridad.estadistica** Las acciones se auditan en la tabla **seguridad.auditoria**

		3. Por cada identificación se deberá registrar: a. Fecha/hora de proceso de la identificación b. Porcentaje de similitud 4. Registrar la cantidad de veces que se realizan operaciones de validaciones por las aplicaciones clientes 5. El método deberá devolver como respuesta el porcentaje de similitud.					

Observación: *El diseño de las bases de datos estructurada y NO estructurada se encuentran en las imágenes 12 y 13.*

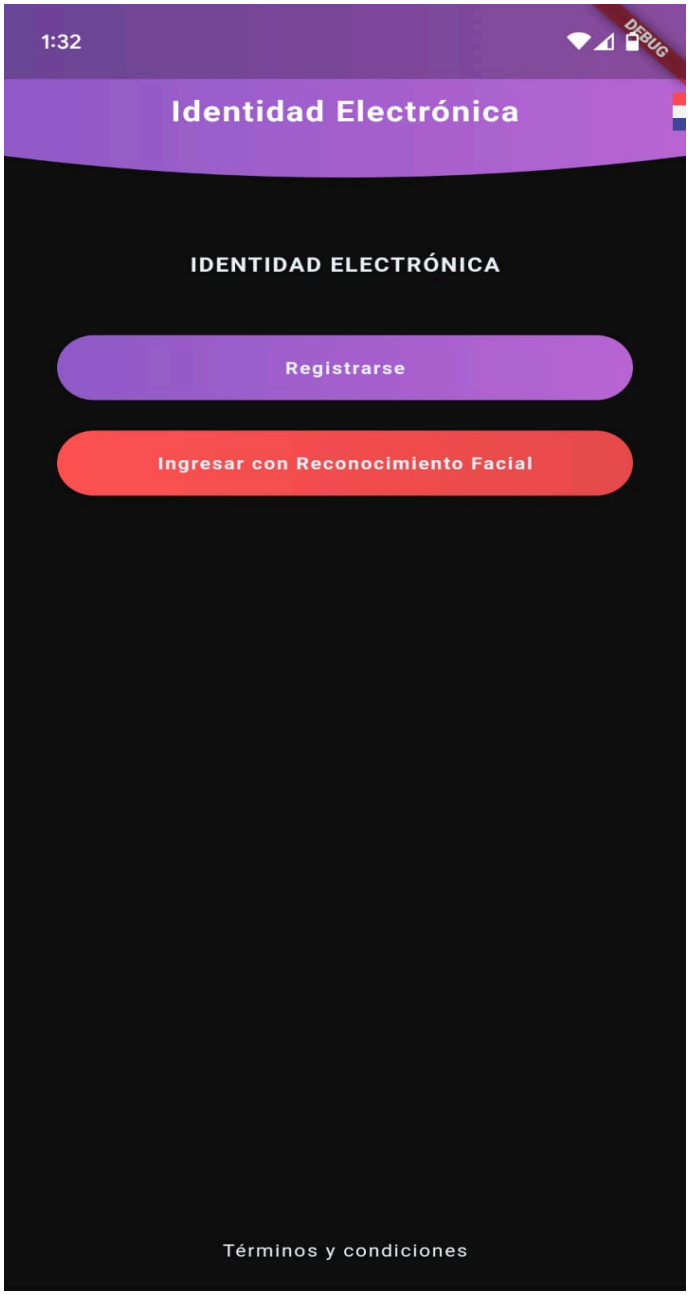


Imagen 1. *Inicio*

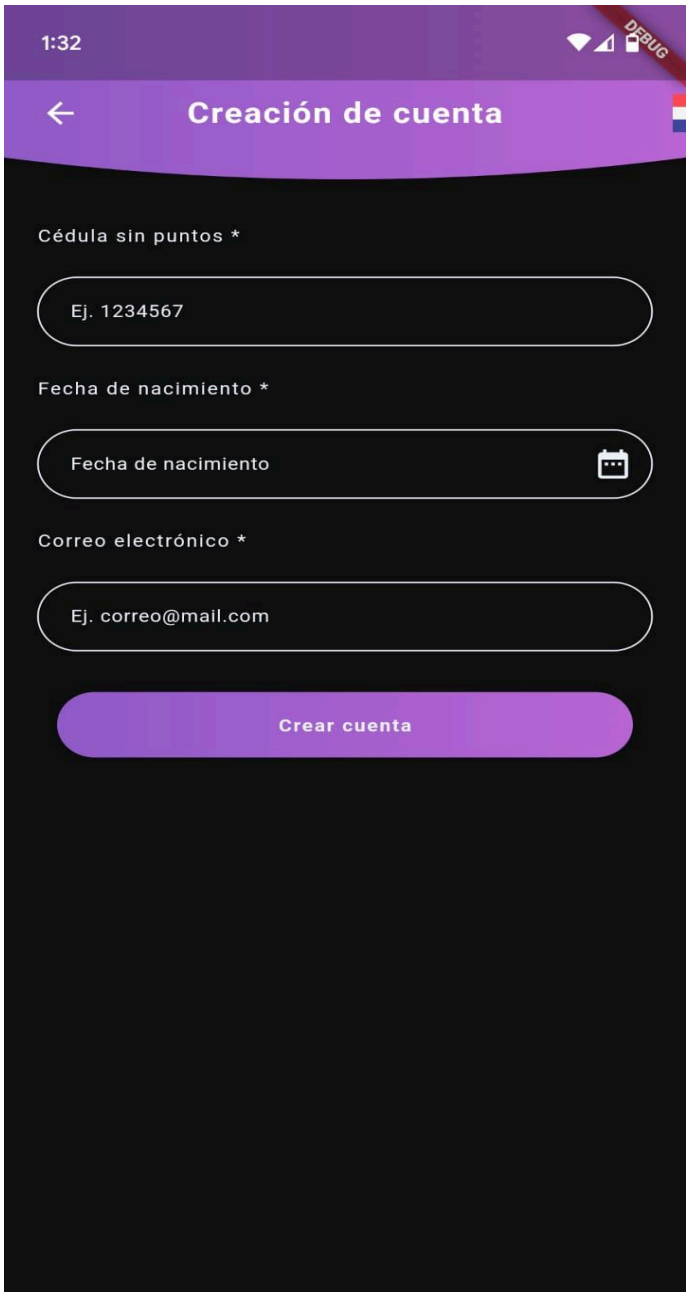


Imagen 2. *Enrolamiento*

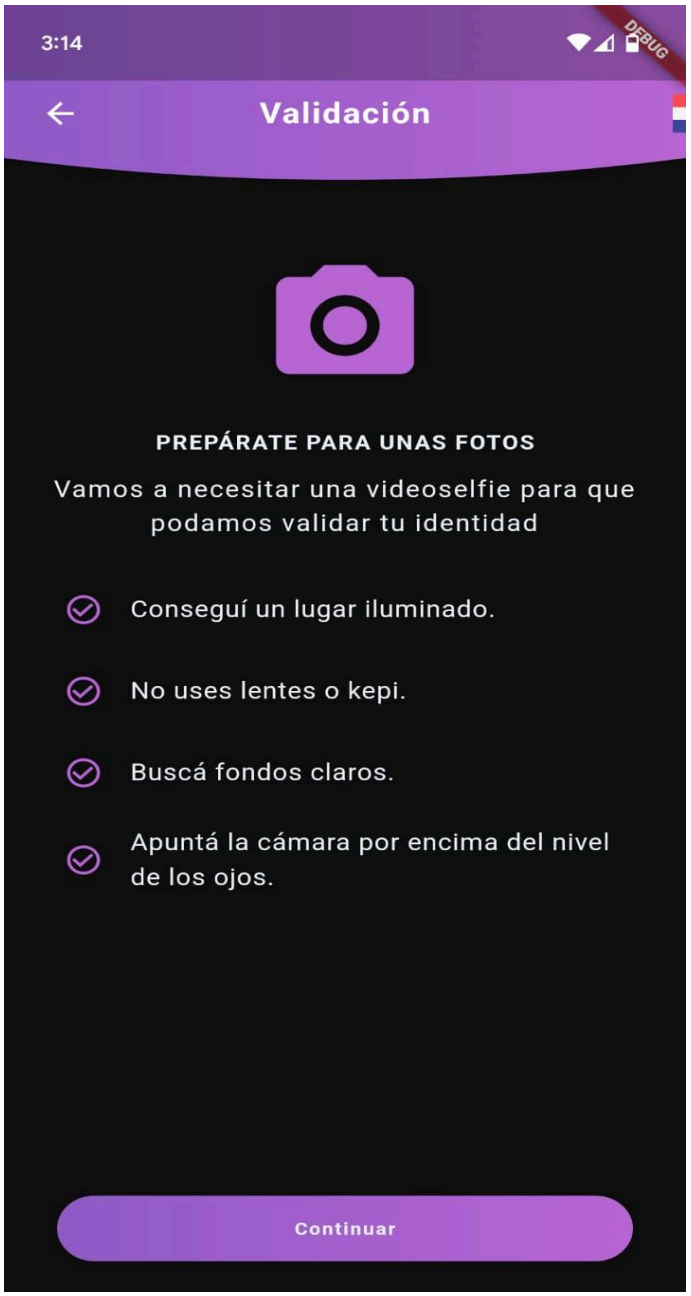


Imagen 3. *Validaciones*

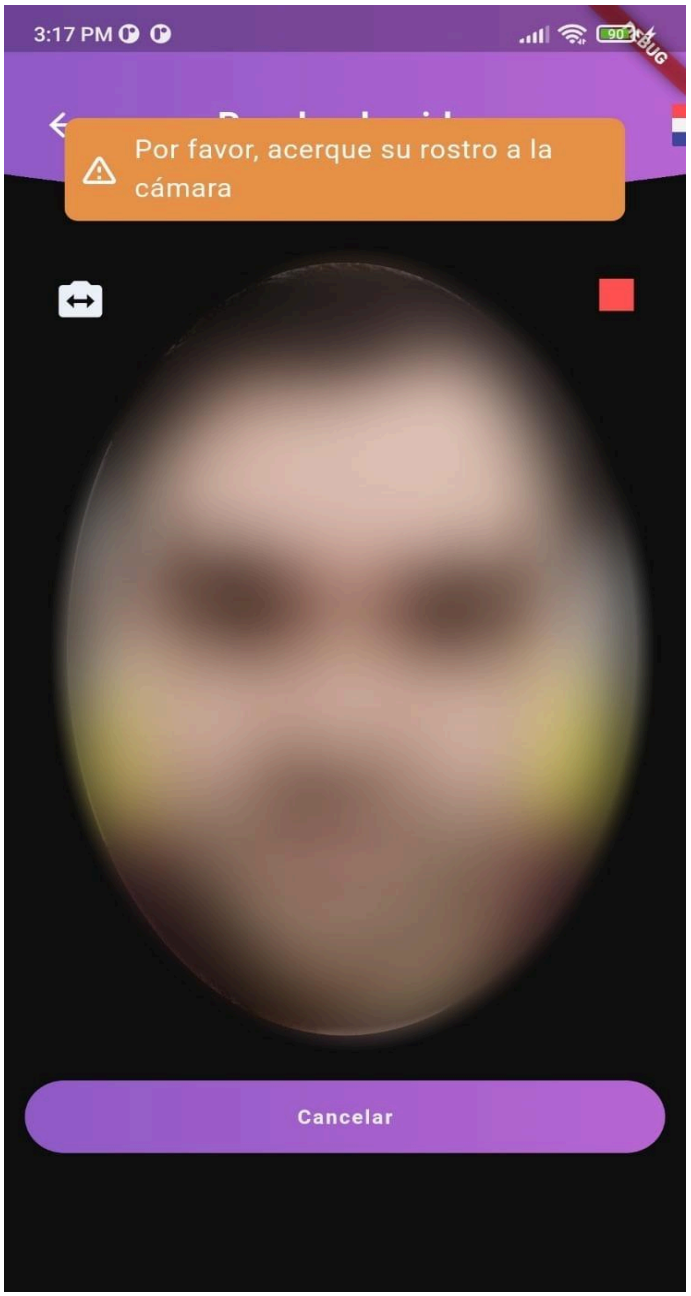


Imagen 4. *Prueba de vida, acerque su rostro a la cámara*

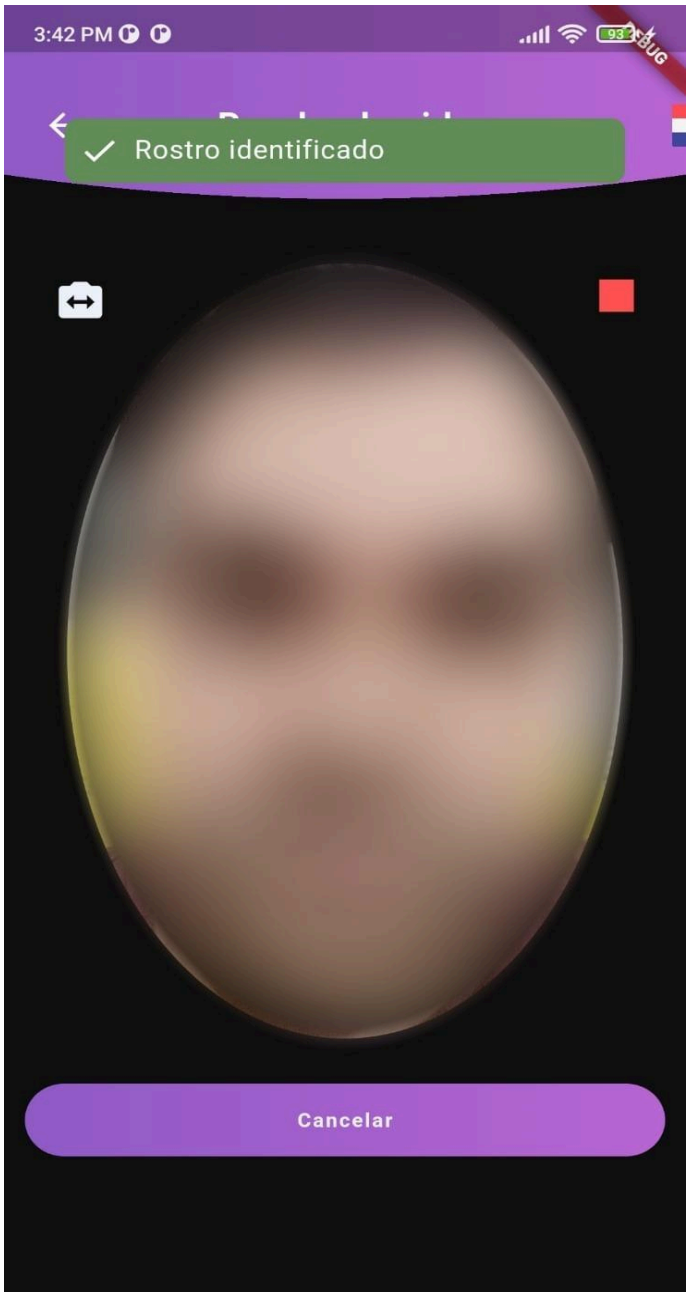


Imagen 5. *Prueba de vida, rostro identificado*

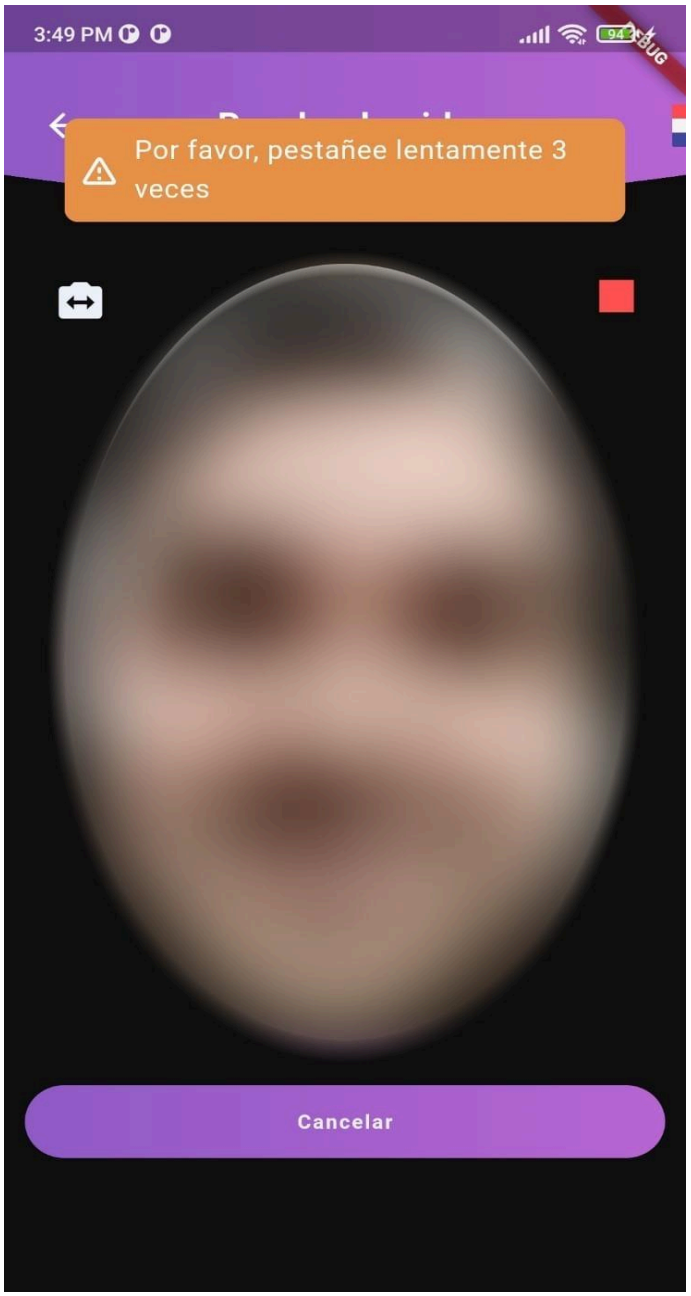


Imagen 6. *Prueba de vida, pestañee tres veces*

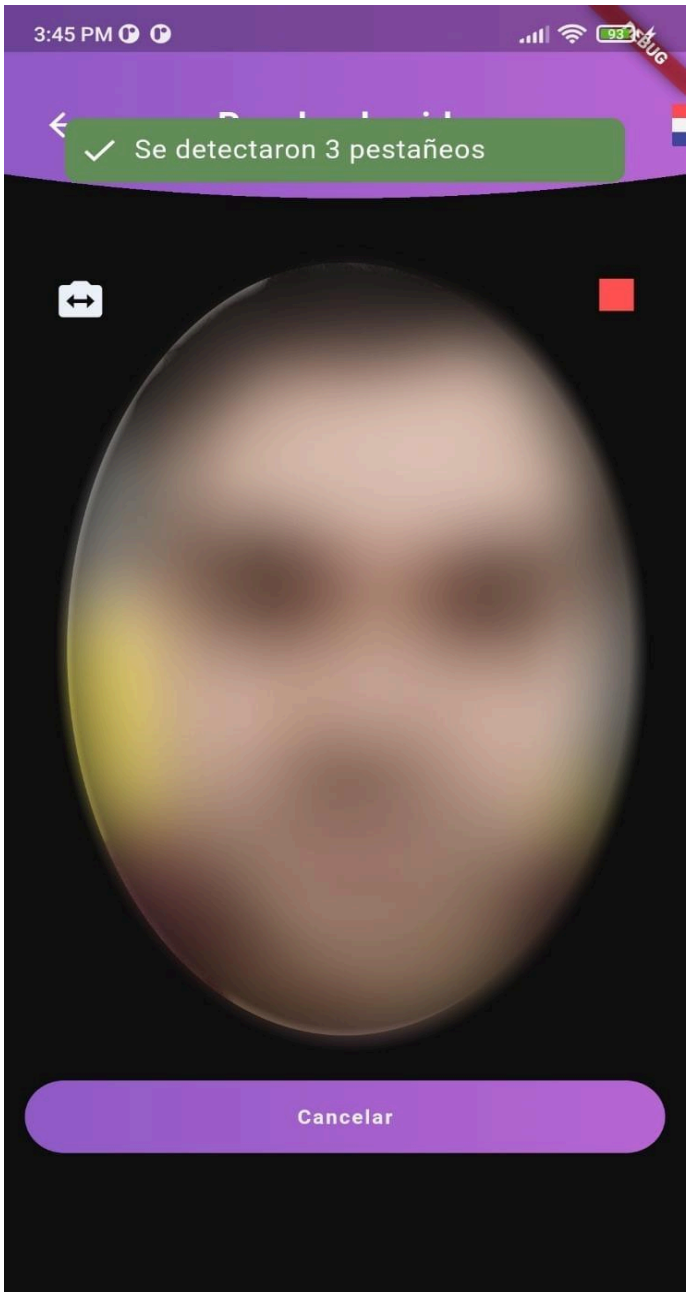


Imagen 7. *Prueba de vida, se detectaron pestañeos*

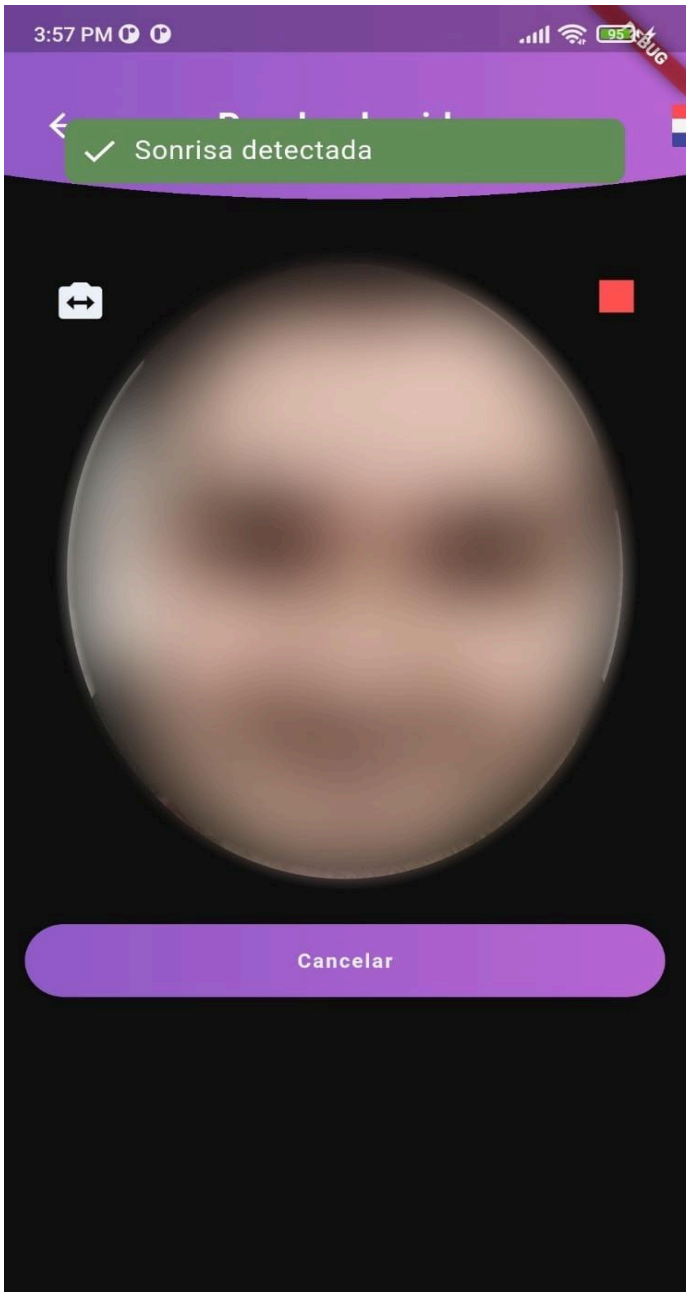


Imagen 8. *Prueba de vida, sonrisa detectada*

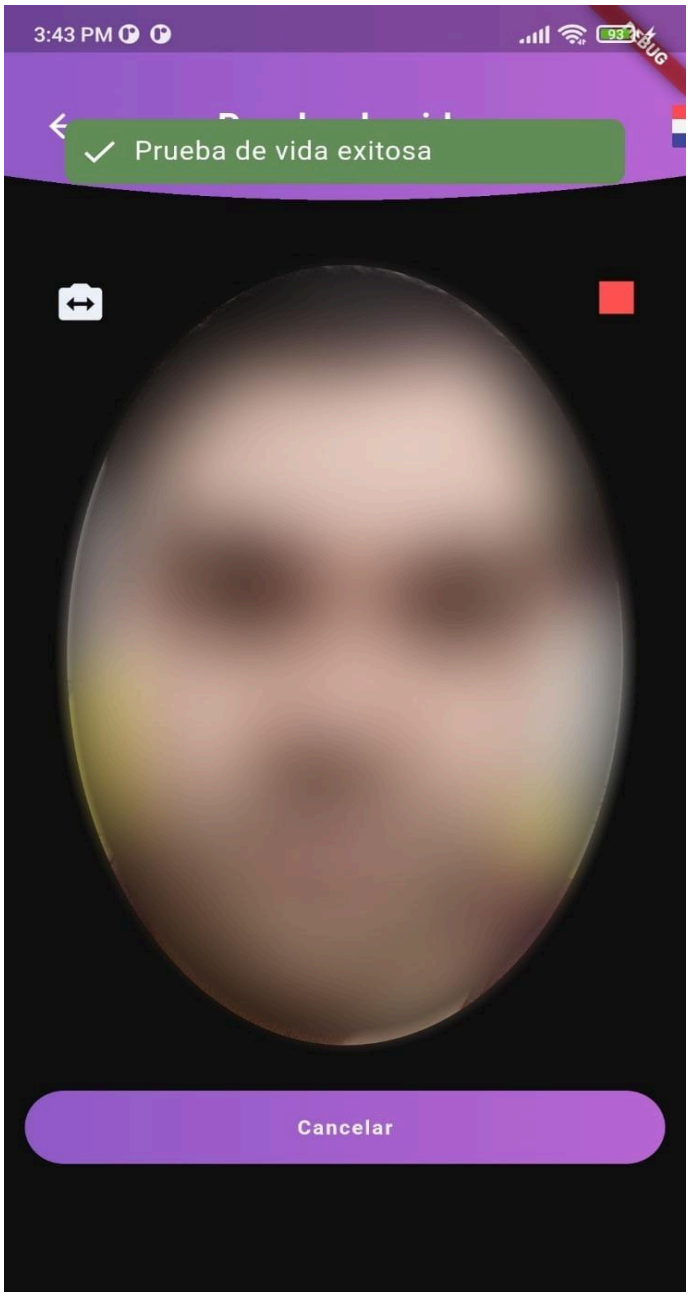


Imagen 9. *Prueba de vida exitosa*

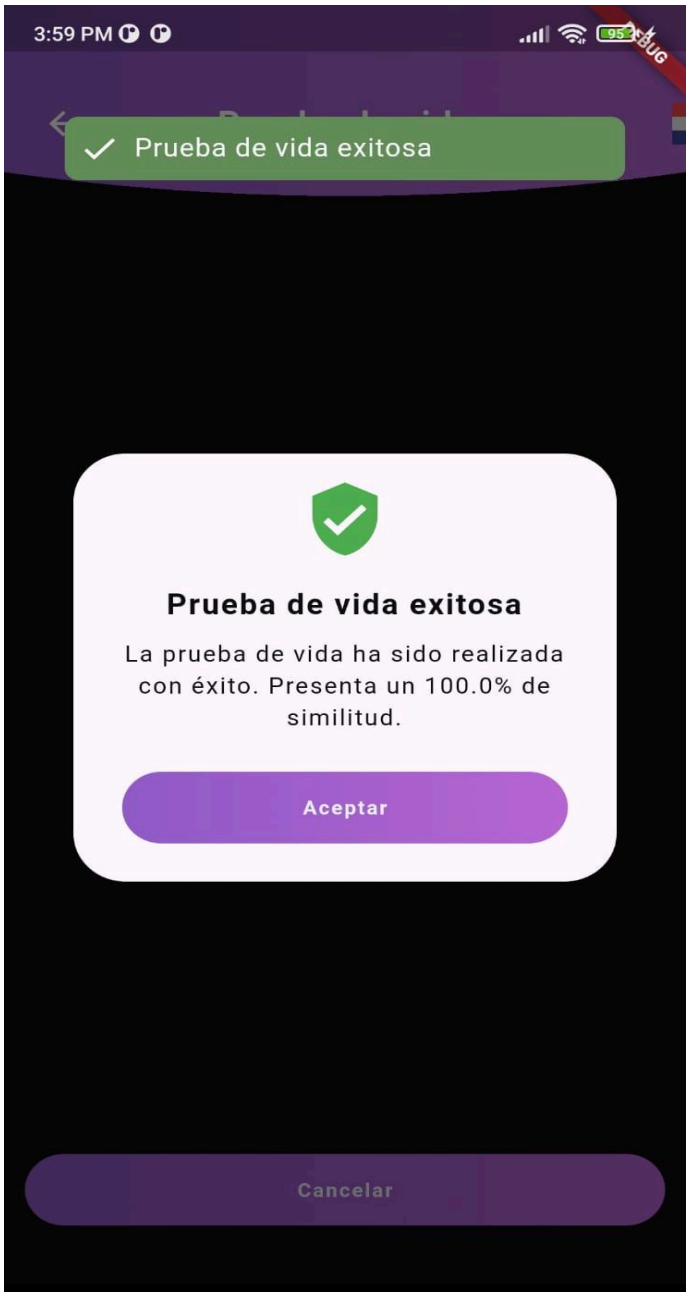


Imagen 10. *Prueba de vida exitosa con porcentaje de similitud*

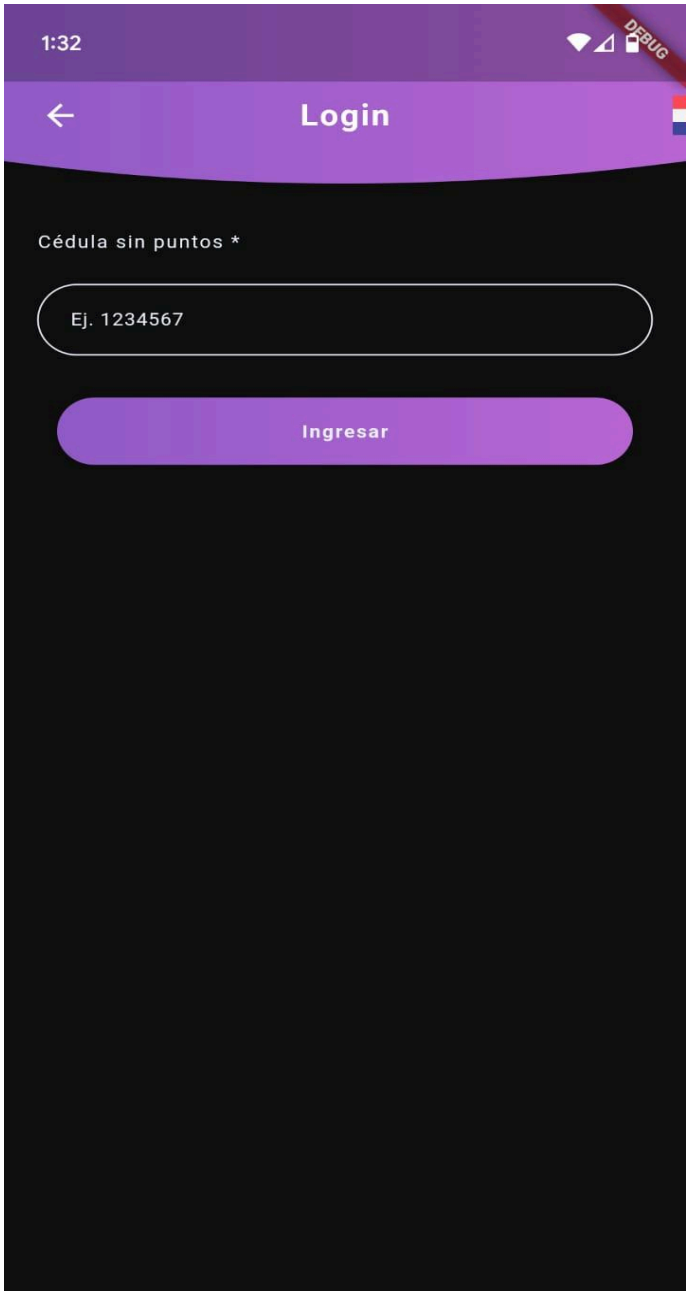


Imagen 11. *Ingresar con CI registrada*

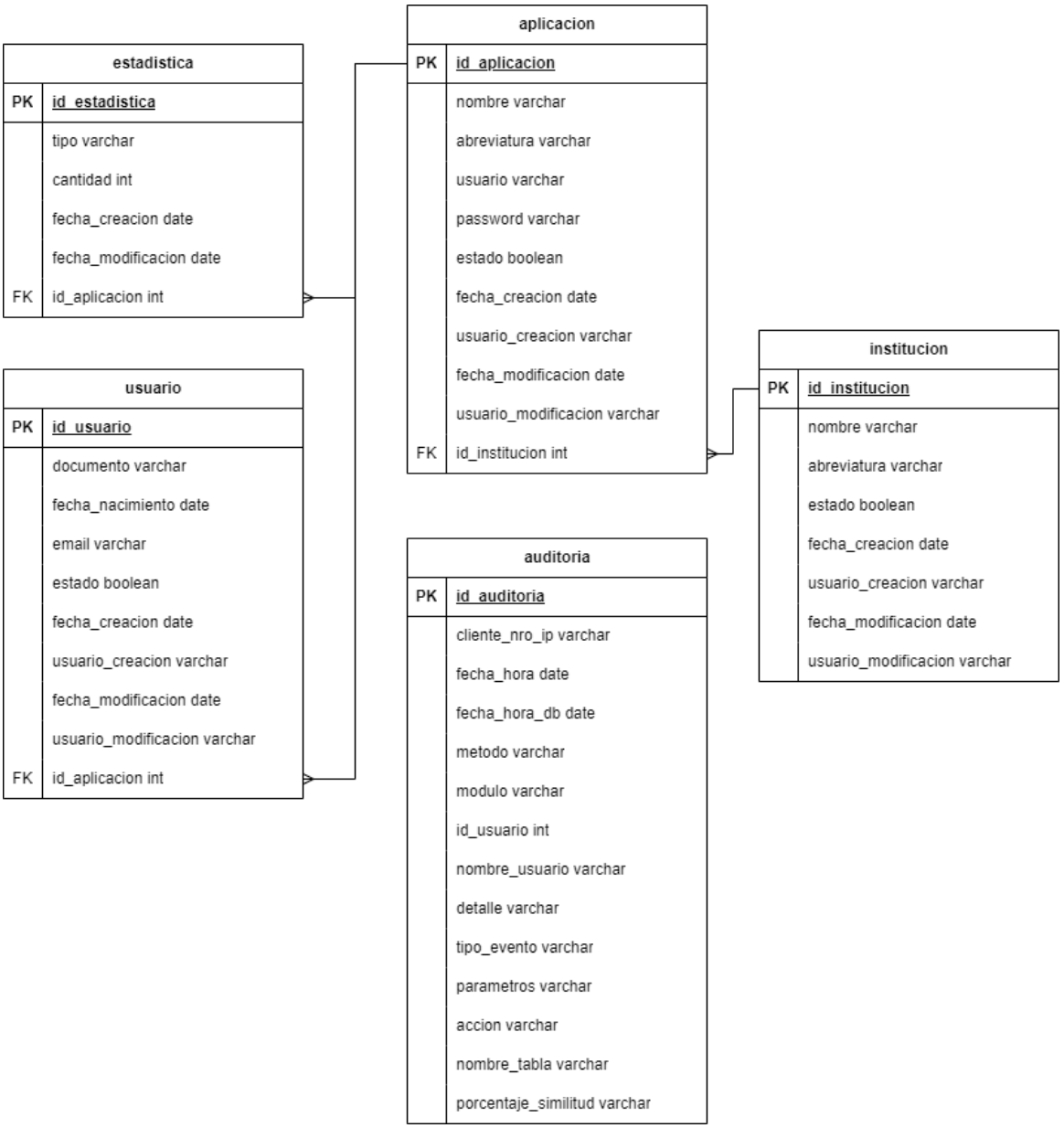


Imagen 12. *Diagrama de Entidad Relación (DER) relacionado a RNF3 - Diseño de Base de datos, RNF4 - Registros de auditoría y RNF6 - Documentación Técnica especificados en la OT*

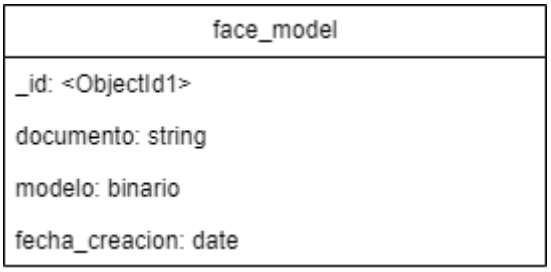


Imagen 13. *Diagrama de Base de datos no estructurada, relacionada a RNF3 - Diseño de Base de datos*

application.bin

```
80 04 95 4B 31 00 00 00 00 00 00 8C 21 73 6B 6C  €□•K1      Œ!skl
65 61 72 6E 2E 6E 65 69 67 68 62 6F 72 73 2E 5F  earn.neighbors.
63 6C 61 73 73 69 66 69 63 61 74 69 6F 6E 94 8C  classification”Œ
14 4B 4E 65 69 67 68 62 6F 72 73 43 6C 61 73 73  □KNeighborsClass
69 66 69 65 72 94 93 94 29 81 94 7D 94 28 8C 0B  ifier””) ”}”(Œ□
6E 5F 6E 65 69 67 68 62 6F 72 73 94 4B 01 8C 06  n neighbors”K□Œ□
72 61 64 69 75 73 94 4E 8C 09 61 6C 67 6F 72 69  radius”NŒ      algori
74 68 6D 94 8C 09 62 61 6C 6C 5F 74 72 65 65 94  thm”Œ  ball tree”
8C 09 6C 65 61 66 5F 73 69 7A 65 94 4B 1E 8C 06  Œ      leaf size”K□Œ□
6D 65 74 72 69 63 94 8C 09 6D 69 6E 6B 6F 77 73  metric”Œ       minkows
6B 69 94 8C 0D 6D 65 74 72 69 63 5F 70 61 72 61  ki”Œmetric para
6D 73 94 4E 8C 01 70 94 4B 02 8C 06 6E 5F 6A 6F  ms”NŒ□p”K□Œ□n jo
62 73 94 4E 8C 07 77 65 69 67 68 74 73 94 8C 08  bs”NŒ□weights”Œ□
64 69 73 74 61 6E 63 65 94 8C 0E 6E 5F 66 65 61  distance”Œ□n fea
74 75 72 65 73 5F 69 6E 5F 94 4B 80 8C 0B 6F 75  tures in ”K€Œ□ou
74 70 75 74 73 5F 32 64 5F 94 89 8C 08 63 6C 61  tputs 2d ”‰Œ□cla
73 73 65 73 5F 94 8C 15 6E 75 6D 70 79 2E 63 6F  sses ”Œ□numpy.co
72 65 2E 6D 75 6C 74 69 61 72 72 61 79 94 8C 0C  re.multiarray”Œ
5F 72 65 63 6F 6E 73 74 72 75 63 74 94 93 94 8C   reconstruct””Œ
05 6E 75 6D 70 79 94 8C 07 6E 64 61 72 72 61 79  □numpy”Œ□ndarray
94 93 94 4B 00 85 94 43 01 62 94 87 94 52 94 28  ””K …”C□b”‡”R”(
4B 01 4B 01 85 94 68 17 8C 05 64 74 79 70 65 94  K□K□…”h□Œ□dtype”
93 94 8C 02 55 36 94 89 88 87 94 52 94 28 4B 03  ””Œ□U6”‰ˆ‡”R”(K□
8C 01 3C 94 4E 4E 4E 4B 18 4B 04 4B 08 74 94 62  Œ□<”NNNK□K□K□t”b
89 43 18 36 00 00 00 35 00 00 00 34 00 00 00 31  ‰C□6   5   4   1
00 00 00 32 00 00 00 38 00 00 00 94 74 94 62 8C     2   8   ”t”bŒ
02 5F 79 94 68 16 68 19 4B 00 85 94 68 1B 87 94  □ y”h□h□K …”h□‡”
52 94 28 4B 01 4B 0A 85 94 68 20 8C 02 69 38 94  R”(K□K
89 88 87 94 52 94 28 4B 03 68 24 4E 4E 4E 4A FF  ‰ˆ‡”R”(K□h$NNNJÿ
FF FF FF 4A FF FF FF FF 4B 00 74 94 62 89 43 50  ÿÿÿJÿÿÿÿK t”b‰CP
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Imagen 14. *Print de archivo binario de una huella facial que se almacena en la base de datos no estructurada MongoDB*

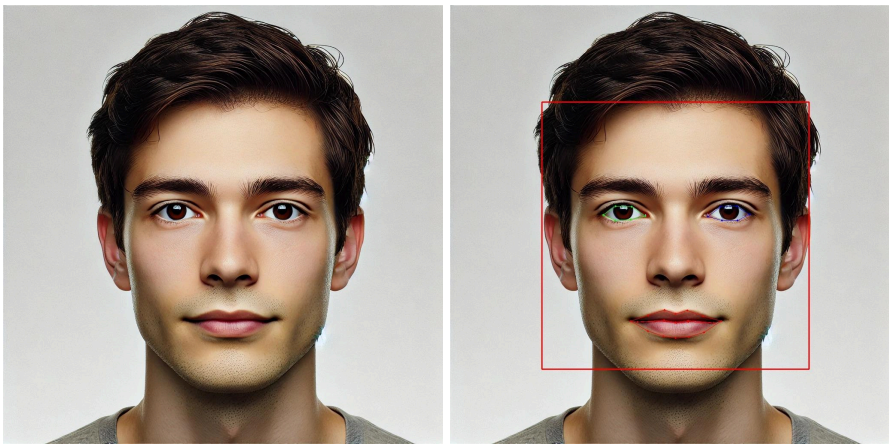


Imagen 15. *Ejemplo de captura y análisis de la imagen del rostro, identificando los puntos de referencia faciales para distinguir un rostro*

6. CORRECCIONES Y PRUEBAS REALIZADAS EN ETAPA DE VERIFICACIÓN Y CICLO DE PRUEBAS

Se llevaron a cabo pruebas con 10 personas que proporcionaron voluntariamente su número de documento y foto tipo selfie.

Link al archivo con los resultado:
https://repo.mitic.gov.py/mitic/reconocimiento-facial/face-recognition-api/-/blob/cds-2024-ot12/documentacion-OT12/ResultadoDePruebas.xlsx

A continuación un resumen de los pedidos y sus estados.

Contenido de la plantilla de estado de ajustes/mejoras solicitadas:

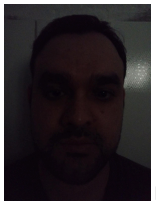
- **Id:** Cada caso de prueba debe tener un identificador (código) único
- **Descripción del pedido:** pedido realizado en etapa de verificación y primer ciclo de prueba de la OT en cuestión
- **Comentario:** comentario u observaciones sobre los ajustes realizados
- **Estado:** Se indica si se pudo realizar o no el pedido
- **Commit:** Se indica el commit relacionado en el repositorio indicado para la OT

ID	Descripción del pedido	Comentarios sobre el Ajuste/Mejora	Estado	Commit
1	Mejoras y/o ajustes en los parámetros de SMILE_THRESHOLD y BLINKS_THRESHOLD	Se modificó el parámetro de SMILE_THRESHOLD para detectar la sonrisa de más personas. Con el nuevo número establecido se detectó la sonrisa de todas las personas que colaboraron con las pruebas. No identificamos necesidad de mejora para el parámetro BLINKS_THRESHOLD.	HECHO	4afc61136d3afcb26493ce1e4adefa1104c2fde9
2	Mejorar o verificar el reconocimiento de imagen obtenida de la cédula, contra sí misma, ya que en vez de alcanzar un 100% alcanzó un 95.96%	Realizamos ajustes que nos permitieron obtener la similitud del 100% comparando la imagen obtenida de la cédula contra sí misma. El inconveniente se daba porque se rotaba la imagen, el ajuste consiste en identificar los casos donde la imagen no necesita rotación y por lo tanto se devuelve la imagen original sin rotar.	HECHO	31a0edff224f5ca96e1f589e8fd514c8abd0ff1d
3	Ajustar el Threshold distance para login, y que sea configurable desde el archivo .env.	Se han realizado los cambios necesarios para parametrizar el Threshold del login.	HECHO	217db1e3de0345bbc52f2e43438ecfe9c4dacef f

4	Agregar al .env todas la variables posibles referentes a ajustes de detección de rostro.	Se han agregado todas las variables de ajuste de detección de rostro al archivo .env.	HECHO	217db1e3de0345bbc52f2e43438ecfe9c4dacef f 04d81be1aabdf2f6e6b09af04d88828a9d5a37 6b f33d151405a6511c2d4a47411364846a7e5312 8a
5	Ajustar las presiones de la librería face_recognition, para obtener un porcentaje de similitud más preciso.	Planilla de Pruebas: https://repo.mitic.gov.py/mitic/reconocimiento-facial/face-recognition-api/-/blob/cds-2024-ot12/documentacion-OT12/ResultadoDePruebas.xlsx Se realizaron varias modificaciones para intentar mejorar la presición. Entre los cambios más considerables tenemos la instalación de la librería deepface de python que permite comparar 2 fotos o imágenes y retornar un porcentaje de similitud entre ellas. También se agregó un endpoint para obtener y probar directamente los porcentajes de similitud que retornaban las distintas funciones (distintas librerías) que sirven para eso y se modificó la fórmula del porcentaje de similitud: ● **Dlib con face_recognition:** utiliza la distancia euclidiana para detectar la distancia entre rostros, en un rango de $[0,\ 2]$, siendo los valores cercanos a 0 los rostros similares	HECHO	A partir del commit b1224708a5ca954949e0d2f04a5d2ea9030d6d5b serían los cambios que realizamos para hacer estas pruebas.

		Formula distancia Euclidiana: $d = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$ **Fórmula para la similitud:** $s = (1 - \frac{d}{2}) \times 100$ En resumen: $d = 0 \Rightarrow s = 100\%$ de similitud $d = 0.6 \Rightarrow s = 70\%$ de similitud $d = 1 \Rightarrow s = 50\%$ de similitud $d = 2 \Rightarrow s = 0\%$ de similitud Por encima del 70% o un threshold <= 0.6 se considera que la foto de la persona coincide con la de la cédula. ● **Deepface:** similar a Dlib, ya que también usa la fórmula euclidiana, pero la diferencia está en que también te permite elegir hacer el cálculo de distancia usando la fórmula del coseno. En las pruebas nos percatamos que al comparar la foto de la cédula con la cédula misma no tira un 100% sino que son valores muy cercanos como 99.99%. En el endpoint mencionado se agregó la posibilidad de ir registrando las pruebas en un archivo CSV en el cual se guardan		

		varios parámetros de las imágenes que se suben, como se puede observar en la planilla de pruebas. La conclusión a la que se llegó con los datos que se fueron acumulando en el archivo CSV es que el porcentaje de similitud de las distintas funciones no se ve afectado por metadatos de la imagen como ser el tamaño, brillo o calidad, pero si se llegó a definir límites para procesos computacionales: ● Ancho y Alto recomendado: 720 x 900, mínimo de 450x700. El ancho debe ser menor al alto siempre, y la altura máxima debe ser de 2000px para procesar los cálculos con 6GB de RAM. Números mayor a 2000 tardan más exponencialmente. ● Tamaño recomendado: entre 60kb hasta 5mb. Lo más importante es no superar el alto (h) > 2000px para 6GB de RAM. ● Brillo recomendado: con que la persona se vea y distinga bien en la imagen es suficiente, sigue detectando rostro y similitud en brillos bajos pero genera falsos negativos.		

		recomendado  no recomendado (no se distingue rostro) ● Se recomienda que la persona se asemeje lo más posible a la imagen de su cédula. Por ejemplo, es preferible evitar tener barba si en la cédula aparece sin ella, o viceversa. Factores como la edad o el cambio de peso también pueden influir en el porcentaje de similitud. No obstante, en las pruebas realizadas, la precisión se mantenía por encima del 70% siempre que se tratara de la misma persona, independientemente de la edad. Además, concluimos que si la foto de la cédula es alterada, ya sea comprimiéndola o cambiando el formato (de JPG a PNG, por ejemplo), la similitud retornada no es del 100%, sino que presenta una diferencia promedio del 3%. Es decir, en estos casos, el sistema arroja un 97% de coincidencia en promedio.		

7. *HISTORIAL DE VERSIONES*

Versión	Fecha	Autor	Modificaciones
1.0	09/08/2024	Dante Pane	Versión original del documento
2.0	09/09/2024	Julia Talavera	Se agrega la sección 6 relacionada a las pruebas y correcciones posteriores a la entrega.

JULIO MANUEL PACIELLO CORONEL

Digitally signed by JULIO MANUEL PACIELLO CORONEL
DN: CN=JULIO MANUEL PACIELLO CORONEL, SERIALNUMBER=CI1490175, G=JULIO MANUEL, SN=PACIELLO CORONEL, OU=F2, O=CERTIFICADO CUALIFICADO DE FIRMA ELECTRONICA, C=PY
Reason: I am the author of this document
Location:
Date: 2024.09.09 23:23:55-04'00'
Foxit PDF Editor Version: 13.0.1

Ing. Julio Paciello Coronel
Coordinador Especialista TIC
Centro de Desarrollo Sostenible SA