Universidad de Lima

Facultad de Ingeniería

Carrera de Ingeniería de Sistemas



FACEFIND LIMA - PERÚ UN SISTEMA DE DETECCIÓN DE PERSONAS DESAPARECIDAS MEDIANTE RECONOCIMIENTO FACIAL PARA CENTROS DE VIGILANCIA

Garay Poma, Kotler Aaron 20221044

Toribio Salcedo, Alejandro Daniel 20222540

Landa Flores, Marcelo Alejandro 20201118

Melchor Aponte, Franco Steven 20223718

Uceda Yacila, Gustavo Alejandro 20222596

Sección 754

Asesor Jorge Luis Irey Núñez

Lima - Perú

Junio de 2025



FACEFIND LIMA - PERÚ UN SISTEMA DE DETECCIÓN DE PERSONAS DESAPARECIDAS MEDIANTE RECONOCIMIENTO FACIAL PARA CENTROS DE VIGILANCIA

ÍNDICE DE CONTENIDOS

CAPITULO I: VISION GENERAL DEL PROYECTO	8
1.1 Propósito	8
1.2 Alcance	9
1.3 Características del Producto	10
1.4 Suposiciones y Restricciones	11
1.4.1 Suposiciones	11
1.4.2 Restricciones.	11
1.5 Participantes del Proyecto, Roles y Responsabilidades	11
1.6 Calendario del Proyecto y Roadmap	13
1.7 ODS y Justificación	14
CAPÍTULO II: BACKLOG DEL PRODUCTO	15
2.1 Mapas de Historias de Usuario	15
2.2 Priorización	16
2.3 Estimación de Historias	18
CAPÍTULO III: SPRINT 1	24
3.1 Objetivo del Sprint	24
3.2 Ítems a Implementar	24
3.3 Forma en que se Desarrollará el Sprint	24
3.4 Reportes	
3.4.1 Definition of Ready (DoR)	26
3.4.2 Definition of Done (DoD)	26
3.5 Diagrama de Casos de Uso de Sistema	27
3.6 Especificación de Casos de Uso	28
CAPÍTULO IV: SPRINT 2	35
4.1 Objetivo del Sprint 2	35
4.2 Ítems a Implementar	35
4.3 Forma en que se Desarrollará el Sprint 2	
4.4.1 Definition of Ready (DoR)	
4.4.2 Definition of Done (DoD)	
4.5 Diagrama de Casos de Uso	
4.6 Especificación de Casos de Uso	
CAPÍTULO V: DIAGRAMAS	
5.1 Diagrama de Clases	
5.2 Diagrama de Despliegue	
5.3 Diagrama de Base de Datos	
ANEXOS	
Anexo 1: Product Backlog.	
Anexo 2: Priorización técnica MoSCoW	

Anexo 3: Estimación de Historias de Usuario	54
Anexo 4: Roadmap	. 54
Anexo 5: Base de datos	
Anexo 6: Coevaluación	54
REFERENCIAS	. 55



ÍNDICE DE FIGURAS

Figura 1.1 Calendario del proyecto	13
Figura 1.2 Roadmap del proyecto.	14
Figura 2.1 Mapa de historias de usuario	15
Figura 2.2 Priorización de historias.	17
Figura 3.1 Diagrama de casos de uso del Sprint 1	27
Figura 3.2 Diagrama de secuencia del caso de uso "Registrar caso de persona desaparecia	da''
Figura 3.3 Mockup - "Registro de persona desaparecida"	29
Figura 3.4 Mockup - "Carga de fotografías de referencia"	30
Figura 3.5 Diagrama de secuencia del caso de uso "Procesar imágenes y generar embedo faciales"	_
Figura 3.6 Especificación del caso de uso "Actualizar embeddings al añadir nuevas imágenes"	34
Figura 4.1 Diagrama de casos de uso del Sprint 2	38
Figura 4.2 Diagrama de secuencia del caso de uso "Actualizar información de casos"	40
Figura 4.3 Mockup - "Editar Caso"	
Figura 4.4 Diagrama de secuencia del caso de uso "Gestionar coincidencias"	43
Figura 4.5 Diagrama de secuencia del caso de uso "Resolver coincidencias"	45
Figura 4.6 Mockup - "Validación de Detección"	45
Figura 4.7 Mockup - "Detalle de Caso"	46
Figura 4.8 Mockup - "Evidencias de Detecciones"	46
Figura 4.9 Diagrama de secuencia del caso de uso "Consultar información estadística"	47
Figura 4.10 Mockup - "Dashboard de Estadísticas".	48
Figura 4.11 Mockup - "Búsqueda Rápida de Casos"	48
Figura 5.1 Diagrama de Clases de FaceFind Lima - Perú.	49
Figura 5.2 Diagrama de Despliegue de FaceFind Lima - Perú	50
Figura 5.3 Diagrama de Base de Datos Relacional de FaceFind Lima - Perú	52

ÍNDICE DE TABLAS

Tabla 1.1 Participantes del Proyecto	11
Tabla 2.1 Estimación de Historias de Usuario	18
Tabla 3.1 Especificación del caso de uso "Registrar caso de persona desaparecida"	28
Tabla 3.2 Especificación del caso de uso "Procesar imágenes y generar embeddings fac 31	iales".
Tabla 3.3 Especificación del caso de uso "Actualizar embeddings al añadir nuevas imág 33	genes".
Tabla 4.1 Especificación del caso de uso "Actualizar información de casos"	39
Tabla 4.2 Especificación del caso de uso "Gestionar coincidencias"	42
Tabla 4.3 Especificación del caso de uso "Resolver coincidencias"	44
Tabla 4.4 Especificación del caso de uso "Consultar información estadística"	47

CAPÍTULO I: VISIÓN GENERAL DEL PROYECTO

1.1 Propósito

FaceFind tiene como propósito mejorar los esfuerzos de búsqueda y localización de personas desaparecidas en Lima, Perú, mediante la aplicación de tecnologías de reconocimiento facial. Este sistema busca ser un complemento para las instituciones y familias, facilitando la comparación de imágenes y videos que permitan identificar a las personas reportadas como no ubicadas, todo ello a través de una interfaz sencilla, rápida y accesible.

Según el Instituto de Democracia y Derechos Humanos de la PUCP (IDEHPUCP, 2024), en los primeros dos meses del año se reportaron 3 295 personas desaparecidas en el país, de las cuales más del 54 % eran adolescentes entre los 12 y 17 años. Esta tendencia representa un riesgo creciente para grupos vulnerables y evidencia la necesidad de reforzar los sistemas de respuesta inmediata.

Además, hasta octubre de 2024 se contabilizaron más de 14 900 personas desaparecidas a nivel nacional, y solo unas 7 300 habían sido localizadas, lo que indica que aproximadamente la mitad seguía sin ser ubicada (IDEHPUCP, 2024). Un reporte periodístico también señala que entre 2021 y 2024 se interpusieron unas 74 900 denuncias por desaparición, de las cuales más del 50 % permanecen abiertas (Infobae, 2024).

Frente a este panorama, FaceFind Lima - Perú se plantea como una solución tecnológica orientada a reducir los tiempos de búsqueda y aumentar la precisión en la identificación. A través de su pipeline de trabajo, el sistema permite registrar información y fotografías, generar vectores de características faciales (embeddings) y facilitar la comparación posterior con evidencia visual. Está diseñado para integrarse en procesos institucionales o comunitarios, con el objetivo de mejorar las tasas de localización, especialmente en los primeros días críticos tras una desaparición.

1.2 Alcance

El presente proyecto plantea una solución tecnológica para optimizar la gestión de casos de personas desaparecidas mediante reconocimiento facial en centros de vigilancia. La plataforma web estará dirigida a instituciones públicas encargadas de seguridad ciudadana (por ejemplo, Centrales de Seguridad Integrales) y permitirá centralizar desde la creación del caso hasta la confirmación y cierre de alertas.

Registro y Gestión de Casos

- Registro inicial de datos personales y físicos: nombre, fecha de nacimiento, sexo, fecha y lugar de desaparición.
- Carga de fotografías en tres ángulos (perfil izquierdo, frontal y perfil derecho) con validación de formatos y tamaños.
- Edición y reemplazo de imágenes y actualizaciones de datos en casos existentes.

Procesamiento de Imágenes y Embeddings

- Procesamiento automático de archivos JPG y PNG, con verificación de calidad y peso máximo permitido.
- Extracción de embeddings faciales dimensionalmente validados para comparaciones biométricas.
- Actualización automática de vectores tras cada nueva foto o reemplazo.

Detección y Validación Automática

- Análisis de imágenes y video (soporte multi rostros) para detectar y separar cada cara.
- Validación de consistencia entre embeddings de un mismo caso y bloqueo de errores fuera de rango.
- Priorización de alertas según porcentaje de similitud biométrica (> 95 % alta, 85-95 % media, < 85 % baja).

Alertas y Monitoreo

- Generación de alertas en tiempo real con orden descendente de prioridad.
- Búsqueda rápida por nombre o alias y visualización de historial de alertas.
- Mapa interactivo con geolocalización de detecciones confirmadas.

Auditoría y Seguridad

- Registro exhaustivo en logs de auditoría de todas las operaciones críticas (creación, edición, cierre de casos, notificaciones).
- Esquema de roles y permisos que garantiza acceso controlado a cada función, cumpliendo Ley N.º 29733 y principios éticos de IA.

Reportes y Exportación

- Dashboard mínimo con métricas: alertas generadas, coincidencias confirmadas, falsos positivos y tiempo promedio de detección.
- Exportación de reportes en PDF, Word y Excel, con configuración de rangos de fechas y filtros básicos.

1.3 Características del Producto

- 1. **Detección en Tiempo Real**: Procesamiento de vídeo en directo con tasa de actualización configurable.
- 2. **Alta Precisión de Reconocimiento**: Uso de modelos de última generación para minimizar falsos positivos y negativos.
- 3. **Configuración Dinámica de Cámaras**: Posibilidad de añadir, eliminar o ajustar parámetros de cámaras IP desde la interfaz.
- 4. **Umbrales Personalizables**: Ajuste fino del nivel de confianza necesario para generar una alerta.
- 5. **Notificaciones Multicanal**: Envío de alertas por correo electrónico y SMS a los operadores.
- 6. **Dashboard Intuitivo**: Visualización en tiempo real de la actividad de monitoreo y estadísticas de desempeño.
- 7. **Seguridad** y **Privacidad**: Autenticación de usuarios, roles y permisos, así como encriptación de la base de datos.
- 8. **Escalabilidad**: Arquitectura modular que permite añadir nuevos centros de vigilancia y aumentar la base de datos sin degradar el rendimiento.
- 9. **Soporte de Idioma**: Interfaz en español (predeterminado) y posibilidad de traducir a otros idiomas.

1.4 Suposiciones y Restricciones

Las suposiciones establecen condiciones o premisas que se consideran verdaderas para orientar la planificación y la toma de decisiones, aun cuando no hayan sido completamente verificadas. Por su parte, las restricciones delimitan el marco en el que el proyecto debe desarrollarse, definiendo las barreras y límites que deberán gestionarse durante todo el ciclo de vida. A continuación se presentan las asunciones y las limitaciones consideradas para FaceFind Lima - Perú:

1.4.1 Suposiciones

- Se asume que las cámaras IP instaladas están operativas y cuentan con ancho de banda suficiente para transmisión de video en tiempo real.
- Se asume que la base de datos de personas desaparecidas está actualizada periódicamente y contiene imágenes de calidad adecuada para el reconocimiento.
- Se asume que el personal de vigilancia ha sido capacitado en el uso básico del sistema y en la interpretación de alertas generadas.

1.4.2 Restricciones

- El sistema requiere una conexión de red estable y de baja latencia para el procesamiento continuo de video en vivo.
- Las cámaras deben ser compatibles con protocolos POE/IP y ofrecer transmisión en formatos estándar (por ejemplo, RTSP).
- El rendimiento y la capacidad de respuesta del sistema están limitados por la potencia de cómputo del servidor y la eficiencia del modelo de reconocimiento facial.
- Las alertas solo se generarán cuando el nivel de confianza de la coincidencia supere el umbral configurado, lo cual puede variar según la calidad de la imagen.

1.5 Participantes del Proyecto, Roles y Responsabilidades

Tabla 1.1 *Participantes del Proyecto*

Participantes	Rol	Funciones y Responsabilidades	
---------------	-----	-------------------------------	--

		1
		Posee la responsabilidad de guiar y
		proteger el proceso ágil, asegurándose de
		que se sigan las prácticas y ceremonias
		establecidas. Gestiona el flujo de trabajo,
		garantizando la coherencia en la
Garay Poma, Kotler Aaron	Scrum Master	planificación, ejecución y revisión de
		sprints. Además, detecta y elimina de
	LT	forma proactiva los obstáculos que
<u>C</u>	$\backslash U \cap$	puedan afectar al equipo, y facilita la
	1	comunicación con los usuarios mediante
	1. 4.	la organización de reuniones y eventos
		orientados a recoger feedback y resolver
		dudas.
	7 0 4	
	\checkmark	Cuenta con la función de definir y
		priorizar el backlog del producto,
		estableciendo los requisitos y criterios de
		aceptación para cada entrega. Comunica
Toribio Salcedo, Alejandro	Product Owner	la visión del proyecto a todas las partes
Daniel		interesadas, valida los incrementos
. 1		resultantes y realiza ajustes en función
		de la retroalimentación de usuarios y
0	MCMI	stakeholders. Se encarga de alinear las
C_{λ}	0 141 6	expectativas del negocio con las
(4)	m-	capacidades técnicas, optimizando el
*	VIIA E	valor de cada iteración.
		Este rol lo cumplen de manera
		colaborativa un grupo autoorganizado de
Landa Flores, Marcelo		3 a 9 profesionales que diseñan,
Alejandro		desarrollan y validan los componentes
		del sistema mediante entregas
Uceda Yacila, Gustavo	Team Developer	incrementales de software. Participan en
L	l .	!

Alejandro	la estimación y descomposición de	
	tareas, colaboran en revisiones de código	
Melchor Aponte, Franco	y pruebas de integración, y mantienen	
Steven	altos estándares de calidad, promoviendo	
	la mejora continua de procesos y	
	herramientas.	

1.6 Calendario del Proyecto y Roadmap

Una hoja de ruta es un esquema estratégico que presenta, de forma resumida, las entregas y etapas clave de un proyecto. Disponer de este plan al inicio del desarrollo de software es esencial para sincronizar los objetivos, tiempos y responsabilidades de todo el equipo. El cronograma de actividades para el desarrollo e implementación de FaceFind Lima - Perú se presenta a continuación sobre el calendario y roadmap propio del proyecto:

Figura 1.1Calendario del proyecto

Calendario del proyecto



Figura 1.2 *Roadmap del proyecto*



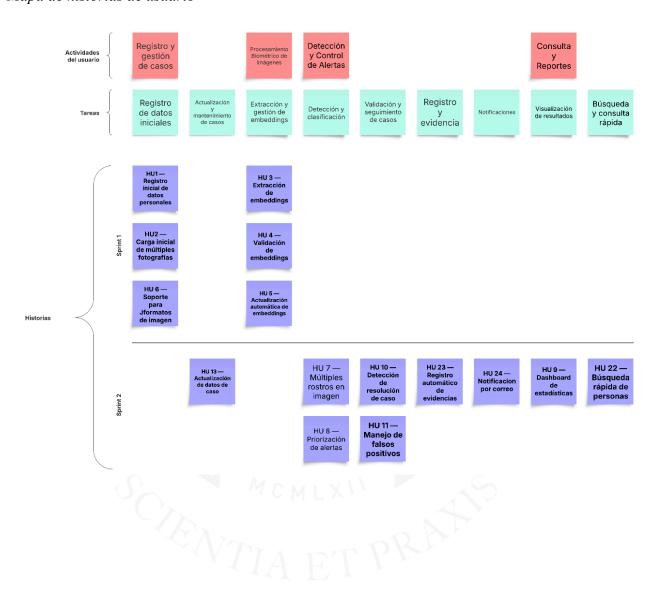
1.7 ODS y Justificación

FaceFind Lima - Perú tiene como objetivo ayudar a encontrar personas desaparecidas usando cámaras y tecnología de inteligencia artificial. Es una herramienta que puede ser muy útil para apoyar a las familias, la policía y las comunidades cuando alguien se pierde. Esta idea también está relacionada con tres Objetivos de Desarrollo Sostenible (ODS) de la ONU. Contribuye al ODS 16, porque ayuda a mejorar la seguridad y apoya el trabajo de las instituciones que cuidan a las personas. También se relaciona con el ODS 11, ya que hace que las ciudades sean lugares más seguros para todos. Y está conectado con el ODS 9, porque utiliza tecnología e innovación para resolver un problema real. En resumen, este proyecto muestra que la tecnología puede servir para ayudar a las personas y hacer un cambio positivo en la sociedad.

CAPÍTULO II: BACKLOG DEL PRODUCTO

2.1 Mapas de Historias de Usuario

Figura 2.1 *Mapa de historias de usuario*



2.2 Priorización

La priorización de las Historias de Usuario (HU) se llevó a cabo utilizando la técnica MoSCoW, que permite identificar y clasificar las funcionalidades según su relevancia para el desarrollo del producto. Este método agrupa las historias en cuatro categorías:

- Must have (Debe tener): Incluye funcionalidades críticas y fundamentales que aseguran la viabilidad del producto. En nuestro proyecto, aquí se priorizaron aspectos esenciales como el registro detallado de personas desaparecidas, carga inicial de fotografías, extracción y validación de embeddings faciales, soporte para formatos de imagen estándar, manejo de múltiples rostros, consulta rápida por nombre, dashboard de estadísticas y registro automático de evidencias.
- Should have (Debería tener): Agrupa funcionalidades importantes pero no imprescindibles para el lanzamiento inicial del producto, y que pueden integrarse en una siguiente iteración. Se incluyeron funcionalidades relacionadas con actualización automática de embeddings, priorización de alertas, manejo de falsos positivos, detección de resolución de casos y notificaciones automáticas por correo electrónico.
- Could have (Podría tener): Estas funcionalidades añaden valor adicional al producto, aunque su implementación no afecta significativamente la operatividad básica del sistema. En esta categoría se clasificaron aspectos como el reemplazo de fotografías existentes, actualización de datos registrados, visualización geográfica de movimientos confirmados, generación de reportes y desactivación temporal de cámaras.
- Won't have (No tendrá por ahora): Son funcionalidades que quedan explícitamente fuera del alcance actual, ya sea por complejidad, recursos o prioridad estratégica. Estas incluyen análisis de múltiples cámaras simultáneas, control avanzado de roles y permisos, gestión de roles de usuario, capturas automáticas en video, configuración de alertas avanzadas y exportación masiva de casos.

La definición final de esta priorización responde directamente a los objetivos estratégicos y operativos del proyecto, garantizando un desarrollo ágil, eficiente y alineado a las expectativas iniciales del producto.

Figura 2.2 *Priorización de historias*

Should Have Must have · HU5: Actualización automática de embeddings - HU1: Registro detallado de persona desaparecida - HU2: Carga inicial de múltiples fotografías de referencia - HU8: Priorización de alertas - HU3: Extracción de vectores de Embeddings - HU10: Detección de resolución de caso - HU4: Validación de consistencia y calidad de los - HU24: Notificación por correo para casos críticos embeddings generados - HU6: Soporte para carga de imágenes en formatos estándar - HU7: Manejo de múltiples rostros simultáneos - HU9: Dashboard de estadísticas - HU22: Consulta rápida de persona por nombre - HU23: Registro automático de evidencias de detección **Could Have** Won't Have - HU14: Análisis de video de múltiples cámaras en paralelo - HU17: Aplicación de roles y permisos de usuario para - HU15: Visualización básica de movimientos detectados acceso seguro HU18: Gestión de roles de usuario (CRUD) - HU19: Capturar en video el avistamiento - HU21: Configuración de alertas por horarios o turnos - HU20: Desactivación temporal de una cámara en - HU25: Exportación masiva de casos resueltos

2.3 Estimación de Historias

Tabla 2.1 *Estimación de Historias de Usuario*

ID	Historia de usuario	Descripción	Puntos de historia
1	Registro inicial de datos personales	Como Administrador, quiero ingresar los datos personales completos de la persona desaparecida al crear un nuevo caso, para tener un registro detallado que facilite su identificación y búsqueda.	3
2	Carga inicial de múltiples fotografías de referencia	Como Administrador, quiero poder subir al menos 3 fotografías desde diferentes ángulos de una persona desaparecida al registrar un nuevo caso, para que el sistema tenga suficiente información visual inicial para realizar una detección efectiva.	5
3	Extracción de vectores de características faciales (Embeddings)	Como usuario, quiero que las imágenes de rostros sean procesadas para extraer vectores de características faciales únicos (embeddings), para que estos vectores puedan ser utilizados para comparaciones biométricas precisas.	13
4	Validación de consistencia y calidad de los embeddings generados	Como sistema, quiero validar que los embeddings generados sean coherentes y útiles para comparaciones biométricas, para que se asegure su calidad antes de ser utilizados en procesos de reconocimiento facial.	8

5	Actualización automática de embeddings	Como administrador del sistema, quiero que cuando se suban nuevas fotos de una persona desaparecida, su vector de características faciales se actualice automáticamente, para asegurar que las futuras búsquedas y detecciones usen siempre la información más reciente y precisa.	8
6	Soporte para carga de imágenes en formatos estándar	Como usuario, quiero poder subir imágenes en formatos comunes como JPG y PNG, para que puedan ser procesadas por el sistema de reconocimiento facial sin problemas de compatibilidad.	3
7	Manejo de múltiples rostros simultáneos	Como usuario, quiero que el sistema pueda detectar y analizar múltiples rostros de un video, para asegurar que ninguna posible coincidencia quede sin revisar en contextos con varias personas presentes.	13
8	Priorización de alertas	Como usuario, quiero que las coincidencias se clasifiquen por prioridad según su porcentaje de similitud (por ejemplo, > 95 % prioridad alta, > 85 % prioridad media), para poder atender primero las alertas con mayor probabilidad de acierto.	8
9	Dashboard de estadísticas	Como usuario o administrador del sistema, quiero disponer de un dashboard	5

		que muestre estadísticas como el número de alertas generadas, coincidencias confirmadas y tiempo promedio de detección, para poder monitorear el desempeño del sistema y tomar decisiones informadas.	
10	Detección de resolución de caso	Como usuario, quiero que cuando una persona desaparecida sea reportada como encontrada, se retire automáticamente de la búsqueda activa, para evitar recibir nuevas alertas innecesarias sobre esa persona en futuras detecciones.	3
11	Manejo de falsos positivos	Como analista de casos,quiero validar manualmente si la detección realizada por el sistema es válida o no, para poder compararla con las fotos entregadas por la familia y marcarla como falso positivo si corresponde, manteniéndose el registro en el sistema	3
12	Reemplazo de fotografías de referencia existentes	Como Administrador, quiero poder reemplazar una o más fotografías previamente subidas de una persona desaparecida. Para asegurar que el sistema siempre cuente con las imágenes más adecuadas, actualizadas o de mejor calidad para la detección	1
13	Actualización de datos de un caso existente	Como Analista, quiero poder editar o actualizar los datos de un caso de persona desaparecida ya registrado, para mejorar la	2

		búsqueda y precisión de los datos del desaparecido.	
14	Análisis de video de múltiples cámaras en paralelo	Como Administrador, quiero que la plataforma pueda procesar múltiples feeds de video de diferentes cámaras en paralelo, para maximizar la cobertura y probabilidad de detección en tiempo real para los Usuarios de Vigilancia y Analistas de Casos.	13
15	Visualización básica de movimientos detectados en mapa	Como Analista de Casos/Investigador, quiero visualizar en un mapa las ubicaciones geográficas donde una persona desaparecida ha sido detectada y confirmada múltiples veces, para obtener una comprensión inicial de sus posibles áreas de movimiento y patrones.	8
16	Generación y visualización de reportes de efectividad	Como Administrador, quiero generar reportes que muestren la efectividad del sistema de detección, para evaluar el rendimiento general, identificar áreas de mejora y justificar la inversión en el sistema.	5
17	Aplicación de roles y permisos de usuario para acceso seguro	Como administrador del sistema, quiero que el sistema aplique un esquema de roles y permisos para controlar qué acciones puede realizar cada usuario, para que cada usuario solo pueda acceder a las funcionalidades y datos acordes a su nivel	5

		de autorización, protegiendo la integridad y seguridad del sistema.	
18	Gestión de roles de usuario (CRUD)	Como administrador del sistema, quiero poder crear nuevos roles, editar los nombres y descripciones de los existentes, y eliminar roles que ya no se necesiten, para mantener una estructura de roles flexible y adaptada a las necesidades de la organización que utiliza el sistema.	3
19	Capturar en video el avistamiento	Como analista, quiero poder recibir capturas del video del momento del avistamiento para poder corroborar la similitud de la persona.	5
20	Desactivación temporal de una cámara en mantenimiento		5
21	Configuración de alertas por horarios o turnos	Como administrador, quiero configurar franjas horarias o turnos en los que se activen alertas de reconocimiento facial, para evitar saturación de alertas en horarios no operativos.	5
22	Consulta rápida de persona por nombre o alias	Como analista de casos, quiero buscar rápidamente personas desaparecidas por nombre o alias en la base, para validar coincidencias detectadas sin revisar todo	2

		el listado.	
23	Registro automático de evidencias de detección	Como administrador, quiero que cada detección confirmada se almacene junto a la evidencia (foto/video del avistamiento), para mantener respaldo histórico completo.	5
24	Notificación por correo para casos críticos	Como supervisor, quiero que el sistema envíe notificaciones por correo electrónico cuando se detecten coincidencias con prioridad alta, para actuar de inmediato aunque no esté conectado al dashboard.	1
25	Exportación masiva de casos resueltos	Como administrador, quiero poder exportar en Excel los casos de personas encontradas con fecha de resolución, para reportes institucionales o legales.	1

CAPÍTULO III: SPRINT 1

3.1 Objetivo del Sprint

Montar completamente la canalización de datos del sistema, desde el registro inicial de casos hasta la generación y validación de embeddings faciales. Este sprint establecerá los componentes esenciales del flujo de reconocimiento facial, lo que permitirá reconocer rostros e identificarlos posteriormente.

3.2 Ítems a Implementar

- Registro inicial de datos personales.
- Carga inicial de múltiples fotografías de referencia.
- Soporte para imágenes en formatos estándar (JPG, PNG).
- Extracción de vectores faciales (embeddings).
- Validación de la consistencia y calidad de los embeddings generados.
- Actualización automática de embeddings al añadir nuevas imágenes.

3.3 Forma en que se Desarrollará el Sprint

Es necesario mencionar que en el grupo no existen subgrupos o rangos , todos los integrantes del grupo trabajarán para cumplir un mismo objetivo. El sprint tendrá una duración (time-box) de cuatro semanas y se espera tener las funcionalidades básicas.

En el primer evento del sprint (Sprint Planning) se requerirá la participación del Scrum Master, Product Owner y el Development Team. El Sprint Planning tendrá una duración de 6 horas aproximadamente y se establecerá la priorización de ítems a realizar en el primer sprint, en otras palabras se definirá qué se hará y cómo se realizarán las tareas del equipo de trabajo. En primer lugar, se empezará con el registro inicial de datos personales, se diseñará e implementará la GUI y se harán las validaciones correspondientes; para posteriormente realizar la extracción de vectores faciales y que los vectores se asocien al usuario correspondiente. Cabe destacar que el ítem de extracción de vectores faciales es el core del sprint y del MVP, ya que nos permite extraer vectores únicos que nos permiten diferenciar rostros. Asimismo en esta reunión los participantes se reunirán de manera presencial en una

sala de juntas con el objetivo de incorporar un sentido de pertenencia entre todos, así como mejorar su relación entre ellos.

De igual forma, se tienen las reuniones diarias (Daily Scrum), se realizan al inicio del día del Sprint y el objetivo es coordinar e inspeccionar el progreso hacia el objetivo del Sprint , así como identificar problemas y clarificar dudas. La duración de cada reunión será de no más de 15 minutos y se espera que participe solo el equipo de desarrollo junto con la participación opcional del Scrum Master y el Product Owner. En cada reunión se actualizará el diario virtual del sprint, con el objetivo de documentar el avance del proyecto. Si bien es cierto, tenemos en claro que las reuniones diarias no son obligatorias, decidimos incluirla en el flujo de desarrollo para asegurar la calidad y apoyarnos mutuamente. Teniendo en cuenta de que las reuniones son diarias, consideramos que la mejor forma en que el equipo puede tener estas reuniones es de forma virtual , ya sea por la plataforma de Zoom o Google Meet.

Asimismo se hará el Sprint Review al final de las cuatro semanas con el objetivo de mostrar lo que se construyó en el sprint al Product Owner y stakeholders para obtener retroalimentación. La duración del evento será de tres horas como máximo, se espera que participe todo el equipo Scrum. En la reunión se espera que los stakeholders realicen preguntas del producto terminado en el primer sprint, por lo que todo el equipo deberá ir preparado. Del mismo modo, los integrantes del equipo comentarán los problemas que tuvieron y cómo llegaron a solucionarlos. Se espera que al momento de presentación del primer Scrum todas las funcionalidades estén validadas correctamente según los criterios de validación. Finalmente el Product Owner actualiza el Backlog para el segundo Sprint.

Por último, en el Sprint Retrospective que se hará inmediatamente después del Sprint Review, se reflexionará sobre el Sprint realizado y se propondrán propuestas de mejora para los siguientes Sprints. La duración del evento será de dos horas y se espera que participe el Development Team.

3.4 Reportes

3.4.1 Definition of Ready (DoR)

Se definieron criterios mínimos que toda historia de usuario debía cumplir para ser seleccionada dentro del Sprint. Estos criterios garantizaron la claridad y viabilidad técnica de cada ítem:

- La historia de usuario debe estar redactada en formato "Como [actor] quiero [acción] para [objetivo]".
- Debe contar con criterios de aceptación detallados.
- La complejidad debe estar estimada en puntos de historia.
- No debe presentar dependencias externas no resueltas.
- Debe estar priorizada dentro del backlog general.
- Identificación de riesgos técnicos y legales (Ley N. ° 29733 y uso ético de IA).

3.4.2 Definition of Done (DoD)

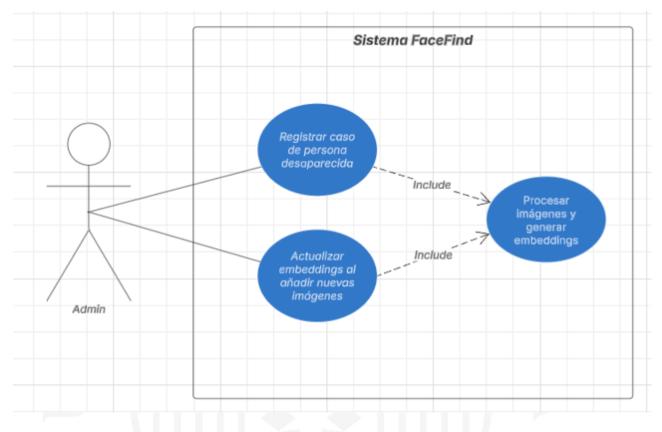
El equipo acordó que una historia sería considerada "terminada" únicamente si cumplía con los siguientes criterios:

- Desarrollo funcional completado.
- Validación manual o automatizada del flujo principal.
- Revisión por al menos un miembro del equipo.
- Registro de pruebas exitosas (cuando aplique).
- Código o flujo integrado en la versión actual del sistema.
- Documentación técnica y funcional completada (si corresponde).
- Cumplimiento de la Ley de Protección de Datos Personales del Perú.
- Aplicación de principios éticos de IA (consentimiento, explicabilidad, no discriminación).

3.5 Diagrama de Casos de Uso de Sistema

Figura 3.1

Diagrama de casos de uso del Sprint 1



3.6 Especificación de Casos de Uso

Tabla 3.1

Especificación del caso de uso "Registrar caso de persona desaparecida"

Nombre:	Registrar caso de persona desaparecida	
Autor:	Marcelo Landa	
Fecha:	20/06/2025	

Descripción: El Administrador registra un nuevo caso de persona desaparecida ingresando información personal y cargando al menos 3 imágenes del rostro.

Actores: Administrador

Historias de usuario: HU1, HU2, HU6

Precondiciones:

El administrador debe estar autenticado. El sistema debe estar operativo.

Flujo Normal:

- 1. El administrador accede a la opción "Registrar caso".
- 2. El sistema muestra el formulario con campos como nombre, edad, descripción, fecha de desaparición, y opción para cargar imágenes.
- 3. El administrador completa los campos y adjunta al menos 3 imágenes (1 de frente y 1 de cada perfil).
- 4. El sistema valida cada imagen (formato y tamaño permitido).
- 5. Si todo es válido, se guarda el caso en la base de datos.
- 6. Se registra la acción en la bitácora del sistema.
- 7. Se muestra un mensaje de éxito al usuario.

Flujo Alternativo:

- Formulario incompleto: el sistema muestra advertencia y no permite avanzar.
- Menos de 3 imágenes válidas: el sistema bloquea el registro.
- Imagen con formato/tamaño inválido: el sistema indica cuál imagen falló.
- Falla al guardar en base de datos: el sistema informa del error y permite reintentar.

Postcondiciones:

El nuevo caso queda registrado con su información e imágenes correspondientes. La acción es registrada en la bitácora.

Figura 3.2

Diagrama de secuencia del caso de uso "Registrar caso de persona desaparecida"

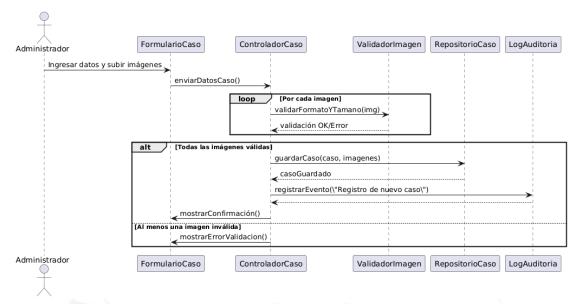


Figura 3.3

Mockup - "Registro de persona desaparecida"

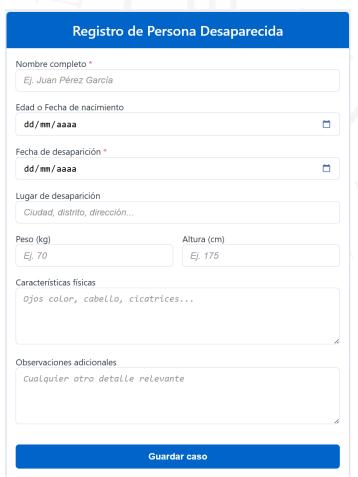


Figura 3.4

Mockup - "Carga de fotografías de referencia"



Tabla 3.2Especificación del caso de uso "Procesar imágenes y generar embeddings faciales"

Nombre:	Procesar imágenes y generar embeddings faciales
Autor:	Franco Melchor
Fecha:	20/06/2025

Descripción: El Administrador, tras registrar un caso, puede iniciar manualmente el procesamiento de las imágenes para generar los embeddings faciales asociados.

Actores: Administrador

Historias de usuario: HU3, HU4

Precondiciones:

El caso debe estar registrado y contar con al menos 3 imágenes válidas.

Flujo Normal:

- 1. El Administrador accede a la vista de detalles del caso.
- 2. Presiona el botón "Procesar imágenes".
- 3. El sistema carga las imágenes asociadas al caso.
- 4. Para cada imagen:
 - a. Detecta el rostro.
 - b. Genera el embedding facial.
- 5. El sistema almacena los embeddings en la base de datos.
- 6. Se muestra un mensaje de confirmación.

Flujo Alternativo:

- Imagen sin rostro: se descarta y se informa al Administrador.
- Calidad insuficiente: se notifica y se permite reemplazar.
- Error al generar embedding: se registra el error y se continúa con las demás imágenes.

Postcondiciones:

Los embeddings quedan registrados y vinculados al caso, listos para comparación en etapas posteriores.

Figura 3.5

Diagrama de secuencia del caso de uso "Procesar imágenes y generar embeddings faciales"

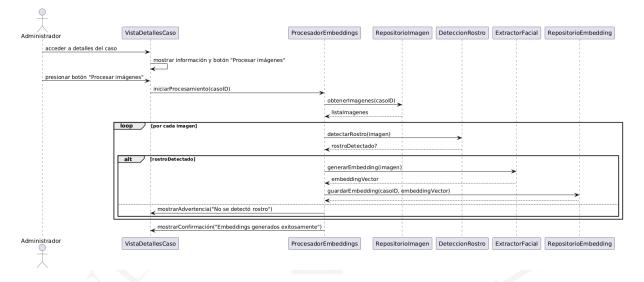


Tabla 3.3Especificación del caso de uso "Actualizar embeddings al añadir nuevas imágenes"

Nombre:	Actualizar embeddings al añadir nuevas imágenes
Autor:	Gustavo Uceda
Fecha:	21/06/2025

Descripción: Cuando se añaden nuevas imágenes al perfil de una persona desaparecida, el sistema actualiza automáticamente los embeddings asociados.

Actores: Administrador

Historias de usuario: HU5

Precondiciones:

El caso debe existir. Nuevas imágenes deben haber sido cargadas y validadas.

Flujo Normal:

- 1. El administrador selecciona un caso y accede a "Actualizar embeddings".
- 2. El sistema detecta las nuevas imágenes.
- 3. Para cada imagen:
 - a. Se genera el embedding.
 - b. Se guarda como versión nueva del embedding.
- 4. Se registra la acción en la bitácora.
- 5. El sistema notifica el éxito de la operación.

Flujo Alternativo:

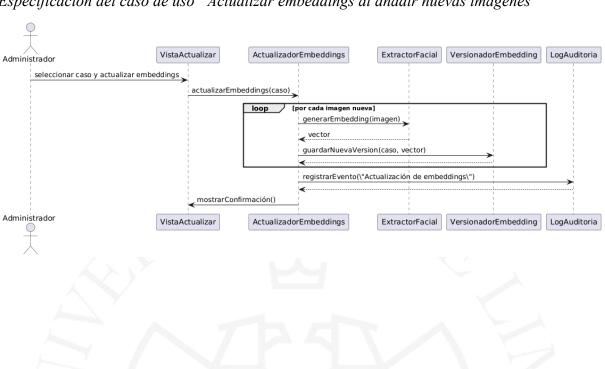
- Error en extracción: el sistema lo informa y permite reintentar.
- Falla en guardado: se muestra error interno y se permite reintentar.

Postcondiciones:

Los embeddings del caso son actualizados, y la acción queda registrada en la bitácora.

Figura 3.6

Especificación del caso de uso "Actualizar embeddings al añadir nuevas imágenes"



CAPÍTULO IV: SPRINT 2

4.1 Objetivo del Sprint 2

Implementar funcionalidades clave para completar el MVP, incluyendo el manejo de múltiples rostros, priorización de alertas, dashboard de estadísticas y gestión avanzada de casos. Estas mejoras permitirán optimizar la precisión del sistema y facilitar la supervisión de coincidencias relevantes.

4.2 Ítems a Implementar

- Manejo de múltiples rostros
- Priorización de alertas por similitud
- Dashboard de estadísticas
- Detección de resolución de caso
- Manejo de falsos positivos
- Consulta rápida de persona por nombre o alias
- Registro automático de evidencias de detección
- Notificación por correo para casos críticos
- Actualización de datos de un caso existente

4.3 Forma en que se Desarrollará el Sprint 2

Al igual que en el primer sprint, todos los integrantes del equipo trabajarán de manera colaborativa para alcanzar un objetivo común. Este sprint tendrá también una duración (time-box) de cuatro semanas y estará enfocado en completar el Producto Mínimo Viable (MVP) del sistema, entregando funcionalidades clave adicionales que complementan las del primer sprint.

El Sprint Planning inicial tendrá la participación activa del Scrum Master, Product Owner y el Development Team, y tendrá una duración aproximada de 6 horas. Durante esta sesión se revisará y priorizará qué funcionalidades se implementarán en el segundo sprint, así como la estrategia y los detalles técnicos para su desarrollo. Este sprint comenzará con la implementación del manejo de múltiples rostros simultáneos, asegurando que el sistema pueda detectar y gestionar adecuadamente varias personas en imágenes o videos.

Seguidamente, se desarrollará la funcionalidad de priorización de alertas por similitud biométrica, permitiendo identificar con rapidez las alertas más relevantes. Asimismo, se implementará un dashboard básico para visualización de estadísticas clave del sistema, incluyendo alertas generadas, coincidencias confirmadas y tiempos promedio de detección.

De igual forma, se abordará la detección y resolución automática de casos, incluyendo el manejo de falsos positivos para mejorar la precisión del sistema. Además, se desarrollará la función de búsqueda rápida de personas por nombre o alias, facilitando la gestión y revisión de casos activos. Finalmente, se implementará el registro automático de evidencias multimedia de cada detección confirmada y el envío automático de notificaciones por correo electrónico en coincidencias críticas, garantizando que el equipo esté informado oportunamente.

Al igual que en el primer sprint, se mantendrán las reuniones diarias (Daily Scrum), de una duración máxima de 15 minutos, con la participación obligatoria del Development Team y opcional del Scrum Master y Product Owner. Estas reuniones se realizarán virtualmente mediante plataformas como Zoom o Google Meet y se actualizará diariamente el registro virtual del avance del proyecto.

Al concluir el sprint, se realizará el Sprint Review con una duración máxima de tres horas, donde el equipo Scrum presentará las funcionalidades desarrolladas al Product Owner y stakeholders para obtener retroalimentación directa, validar la operatividad y asegurar que todas las funcionalidades cumplen con los criterios de aceptación definidos. Además, se discutirán y resolverán dudas sobre las implementaciones realizadas y se actualizará el Product Backlog para posibles iteraciones futuras.

Finalmente, el Sprint Retrospective se realizará inmediatamente después del Sprint Review, con una duración máxima de dos horas. Este evento permitirá al Development Team reflexionar sobre el trabajo realizado durante el segundo sprint, identificando oportunidades de mejora y proponiendo acciones concretas que puedan aplicarse en futuros ciclos de desarrollo, siempre enfocados en mejorar la calidad y efectividad del equipo.

4.4 Reportes

4.4.1 Definition of Ready (DoR)

Se definieron criterios mínimos que toda historia de usuario debía cumplir para ser seleccionada dentro del segundo Sprint . Estos criterios garantizaron la claridad y viabilidad técnica de cada ítem:

- La historia de usuario debe estar redactada en formato "Como [actor] quiero [acción] para [objetivo]".
- Debe contar con criterios de aceptación detallados.
- La complejidad debe estar estimada en puntos de historia.
- No debe presentar dependencias externas no resueltas.
- Debe estar priorizada dentro del backlog general.
- Identificación de riesgos técnicos y legales (Ley N. ° 29733 y uso ético de IA).

4.4.2 Definition of Done (DoD)

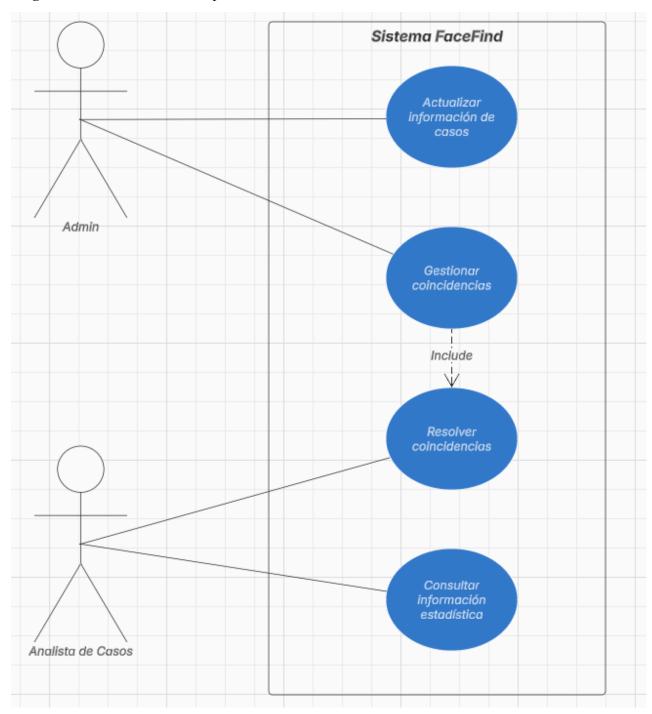
El equipo acordó que una historia sería considerada "terminada" únicamente si cumplía con los siguientes criterios:

- Desarrollo funcional completado.
- Validación manual o automatizada del flujo principal.
- Revisión por al menos un miembro del equipo.
- Registro de pruebas exitosas (cuando aplique).
- Código o flujo integrado en la versión actual del sistema.
- Documentación técnica y funcional completada (si corresponde).
- Cumplimiento de la Ley de Protección de Datos Personales del Perú.
- Aplicación de principios éticos de IA (consentimiento, explicabilidad, no discriminación).

Documentación técnica y funcional completada (si corresponde).

4.5 Diagrama de Casos de Uso

Figura 4.1Diagrama de casos de uso del Sprint 2



4.6 Especificación de Casos de Uso

Tabla 4.1

Especificación del caso de uso "Actualizar información de casos"

Nombre:	Actualizar información de casos	
Autor:	Marcelo Landa	
Fecha:	20/06/2025	

Descripción: Permitir que el Administrador modifique los datos registrados de un caso existente para mantener la información actualizada.

Actores: Administrador

Historias de usuario: HU13

Precondiciones:

Debe existir un caso previamente registrado. El Administrador debe estar autenticado.

Flujo Normal:

- 1. El administrador accede al listado de casos.
- 2. Selecciona el caso a modificar.
- 3. El sistema muestra el formulario con datos actuales.
- 4. El administrador edita los campos.
- 5. Guarda los cambios.
- 6. El sistema valida los datos y actualiza el registro.
- 7. Se registra la acción en la bitácora.
- 8. Se muestra un mensaje de éxito.

Flujo Alternativo:

- Errores de validación: el sistema muestra advertencias y bloquea guardado.
- Falla al guardar: el sistema informa del error y permite reintentar.

Postcondiciones:

El caso queda con la información actualizada y la acción queda registrada en la bitácora.

Figura 4.2

Diagrama de secuencia del caso de uso "Actualizar información de casos"

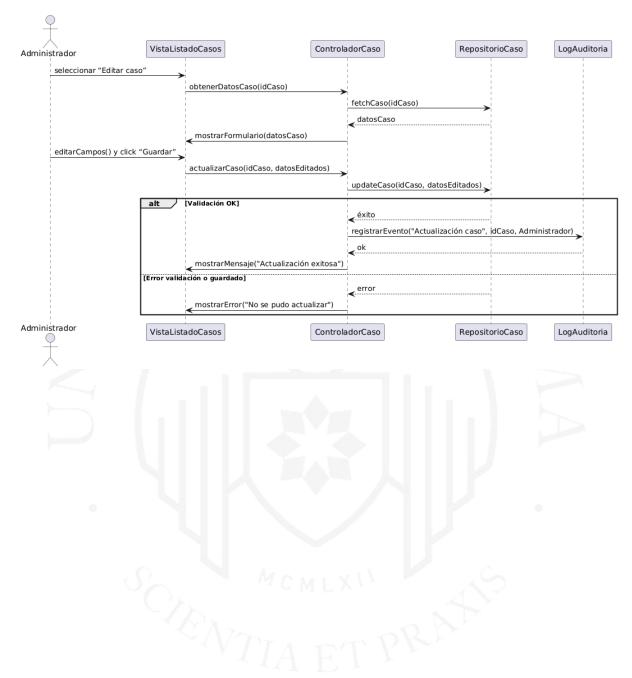


Figura 4.3

Mockup - "Editar Caso"

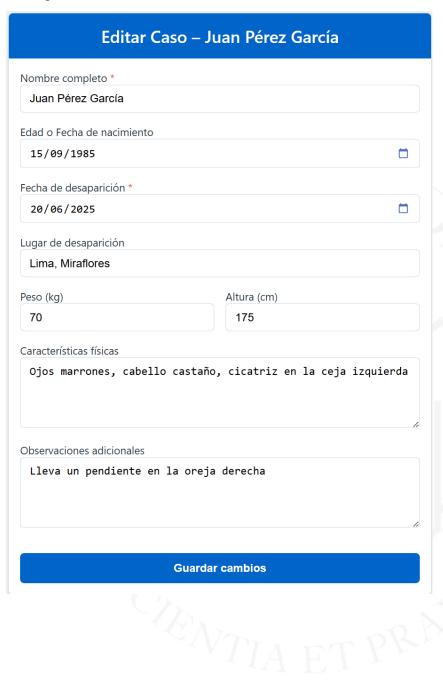


Tabla 4.2 *Especificación del caso de uso* "Gestionar coincidencias"

Nombre:	Gestionar coincidencias	
Autor:	Gustavo Uceda	
Fecha:	20/06/2025	

Descripción: El sistema analiza imágenes o vídeos en busca de rostros coincidentes con los casos registrados. El Administrador puede iniciar/detener el proceso y configurar parámetros como el umbral de similitud.

Actores: Administrador

Historias de usuario: HU7, HU8, HU24

Precondiciones:

Debe haber un flujo de video o imágenes habilitado. El sistema debe estar operativo.

Flujo Normal:

- 1. El Administrador accede a la opción "Gestión de coincidencias".
- 2. Configura parámetros como el umbral de similitud.
- 3. Inicia el proceso de detección.
- 4. El sistema analiza el feed de video o imágenes en tiempo real.
- 5. Extrae los embeddings faciales detectados.
- 6. Compara con la base de datos de personas desaparecidas.
- 7. Si se detecta una similitud superior al umbral, registra la coincidencia y la marca con prioridad alta.

Flujo Alternativo:

- Sin coincidencias: se registra evento sin notificaciones.
- Error de procesamiento: se informa automáticamente al Administrador.

Postcondiciones:

Las coincidencias detectadas se registran y quedan disponibles para su validación por el Analista.

Figura 4.4

Diagrama de secuencia del caso de uso "Gestionar coincidencias"

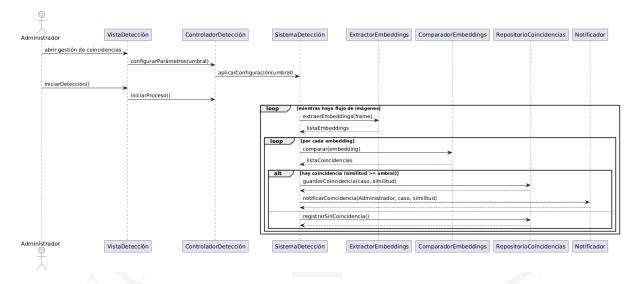


Tabla 4.3 *Especificación del caso de uso* "Resolver coincidencias"

Nombre:	Resolver coincidencias	
Autor:	Franco Melchor	
Fecha:	20/06/2025	

Descripción: Permitir al Analista o Administrador validar manualmente las coincidencias detectadas, marcar falsos positivos, resolver casos confirmados y registrar evidencia asociada.

Actores: Administrador, Analista de casos

Historias de usuario: HU11, HU10, HU23

Precondiciones:

Debe existir al menos una coincidencia detectada pendiente de revisión.

Flujo Normal:

- 1. El Analista accede al listado de coincidencias.
- 2. Visualiza detalles y evidencia.
- 3. Marca como coincidencia correcta o falso positivo.
 - a. Si es correcta: Guarda evidencia y permite marcar caso como resuelto.
 - b. Si es falso positivo: etiqueta y archiva.
- 4. El sistema actualiza el estado del caso y registra la acción.

Flujo Alternativo:

- Sin acción: la coincidencia permanece pendiente.
- Falla al guardar evidencia: el sistema alerta al Analista y registra error.

Postcondiciones:

La coincidencia queda validada o descartada, la evidencia queda almacenada y el estado del caso actualizado.

Figura 4.5

Diagrama de secuencia del caso de uso "Resolver coincidencias"

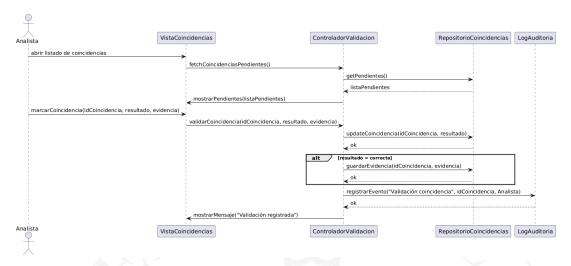


Figura 4.6 *Mockup - "Validación de Detección"*

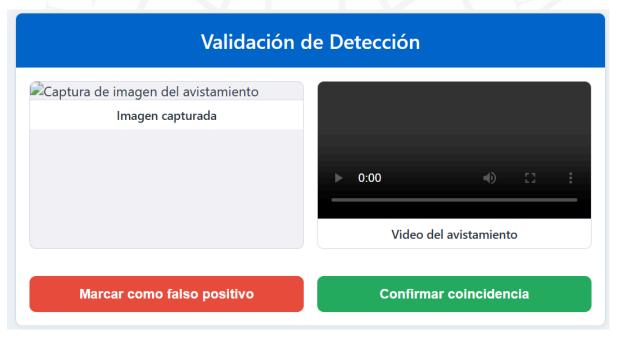


Figura 4.7

Mockup - "Detalle de Caso"

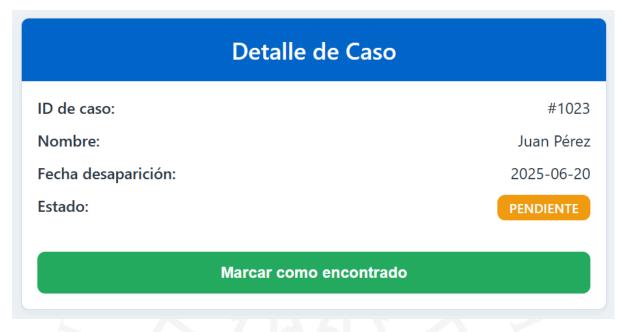


Figura 4.8 *Mockup - "Evidencias de Detecciones"*

Evidencias de Detecciones					
ID Detección	Fecha	Hora	Cámara	Evidencia	Acciones
#1024	2025-07-01	14:23	Entrada Principal	Evidencia 1	Ver Descargar
#1025	2025-07-01	15:05	Pasillo A	Evidencia	Ver Descargar

Tabla 4.4Especificación del caso de uso "Consultar información estadística"

Nombre:	Consultar información estadística
Autor:	Marcelo Landa
Fecha:	20/06/2025

Descripción: Permitir que el Analista consulte rápidamente casos por nombre o alias y visualicen estadísticas operativas desde un dashboard interactivo.

Actores: Analista de casos

Historias de usuario: HU9, HU22

Precondiciones:

Debe existir actividad registrada en el sistema y casos en la base de datos.

Flujo Normal:

- 1. El Analista accede al dashboard.
- 2. Visualiza métricas y gráficos de desempeño.
- 3. Utiliza filtros por nombre o alias.
- 4. El sistema muestra casos coincidentes y estadísticas detalladas.

Flujo Alternativo:

- Sin registros en rango: muestra el mensaje "No hay datos disponibles".
- Sin coincidencias de búsqueda: muestra "No se encontraron casos".

Postcondiciones:

Se muestran los resultados de consulta y las estadísticas solicitadas.

Figura 4.9

Diagrama de secuencia del caso de uso "Consultar información estadística"

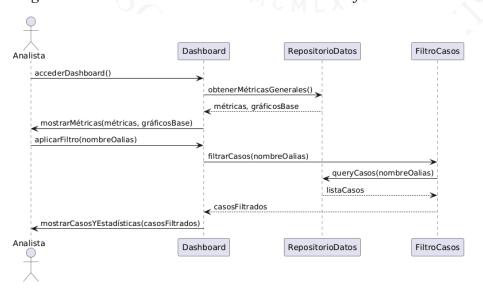


Figura 4.10

Mockup - "Dashboard de Estadísticas"



Figura 4.11 *Mockup - "Búsqueda Rápida de Casos"*

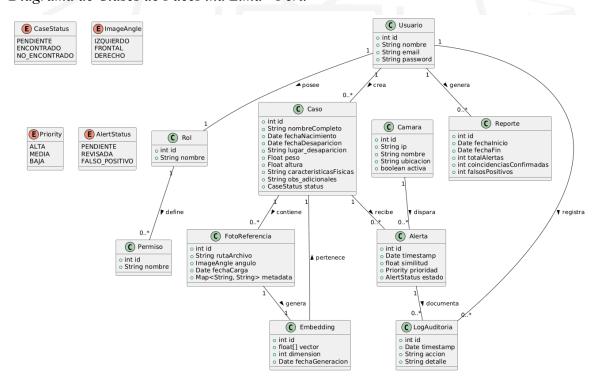
	Búsqueda Rápida de Casos				
Busca	r por nombre		Buscar		
	Nombre	Estado	Acciones		
Foto	María López	ACTIVO	Ver caso		
Foto	Carlos Méndez	CERRADO	Ver caso		

CAPÍTULO V: DIAGRAMAS

5.1 Diagrama de Clases

El diagrama de clases muestra el modelo conceptual para los dos primeros sprints del proyecto. Cada Usuario tiene un Rol específico que determina sus respectivos Permisos, controlando así el acceso al sistema. Los usuarios pueden crear y administrar Casos, cada uno con múltiples Fotos de Referencia asociadas, que incluyen metadatos detallados y generan un Embedding para reconocimiento facial. Las Cámaras generan Alertas al detectar posibles coincidencias con los casos registrados, clasificando estas alertas según prioridad y estado definidos claramente por enumeraciones. Además, el sistema permite generar Reportes estadísticos sobre el desempeño en la identificación de casos. Finalmente, todas las acciones relacionadas con las alertas quedan registradas en un Log de Auditoría, garantizando trazabilidad y transparencia operativa.

Figura 5.1Diagrama de Clases de FaceFind Lima - Perú



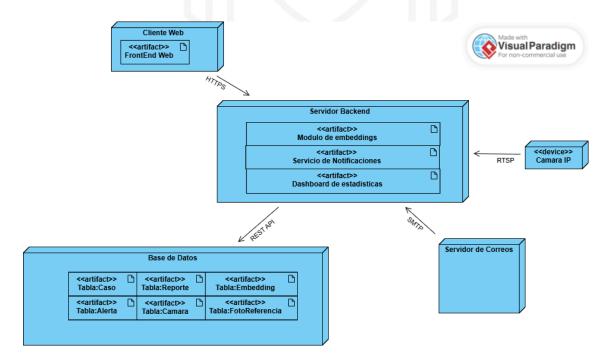
5.2 Diagrama de Despliegue

El diagrama de despliegue muestra la arquitectura física y lógica del sistema Face Find hasta los primeros dos sprints, diseñada para que tenga reconocimiento facial. La arquitectura se basa en un enfoque distribuido, donde los principales componentes se encuentran desplegados en distintos nodos físicos.

En el nodo de Cliente Web , se tiene al artefacto FrontEnd , el cual representa la interfaz gráfica, que los analistas utilizan para la gestión de casos , alertas ,etc. La comunicación con el servidor es mediante el protocolo HTTPS. El Servidor Backend es el núcleo de procesamiento del sistema y aloja el módulo de embeddings, encargado de procesar las imágenes para la generación de vectores faciales; el servicio de notificaciones, que gestiona el envío de alertas; y el dashboard de estadísticas, que provee visualización del rendimiento del sistema en tiempo real. La cámara es el dispositivo físico que captura imágenes en tiempo real , enviando secuencias de vídeo al servidor backend usando el protocolo RTSP. La base de datos contiene toda la información estructurada del sistema : casos, imágenes, embeddings, etc. Se encarga de proveer al servidor Backend REST. Finalmente el servidor de correos es el responsable de envío de notificaciones por correo electrónico y se activa cuando se detectan coincidencias de alta prioridad.

Figura 5.2

Diagrama de Despliegue de FaceFind Lima - Perú



5.3 Diagrama de Base de Datos

El sistema FaceFind ha sido diseñado como una herramienta tecnológica para apoyar la localización de personas desaparecidas mediante reconocimiento facial automatizado. En el centro del sistema se encuentra la entidad Caso, que representa a una persona desaparecida e incluye datos relevantes como nombre, fecha de nacimiento, lugar y fecha de desaparición, características físicas, observaciones, y un estado (CaseStatus) que puede ser PENDIENTE, ENCONTRADO o NO_ENCONTRADO. Según una de las reglas de negocio principales, un caso puede ser reabierto si se obtiene nueva información relevante, cambiando su estado de NO_ENCONTRADO a PENDIENTE para permitir su reintegración en el proceso de búsqueda activa.

Cada caso puede tener asociadas múltiples FotoReferencia (relación 1:N), que son imágenes del rostro tomadas desde diferentes ángulos definidos por el enum ImageAngle (IZQUIERDO, FRONTAL, DERECHO). Estas imágenes deben generar obligatoriamente un Embedding (relación 1:1), una representación matemática del rostro que permite su comparación automatizada. Esta estructura garantiza que toda imagen cargada al sistema tenga utilidad operativa. Una de las reglas de negocio más importantes establece que no se puede generar una alerta si no existe un embedding asociado a la foto referencial del caso.

El sistema utiliza cámaras (Camara) conectadas para realizar capturas en tiempo real. Cuando una cámara detecta una coincidencia entre una imagen capturada y un Embedding registrado, se genera automáticamente una Alerta. Cada alerta está relacionada tanto con el Caso como con la Camara que la originó (ambas relaciones son 1:N). Estas alertas incluyen una puntuación de similitud, una prioridad (Priority: ALTA, MEDIA o BAJA), y un estado de revisión (AlertStatus: PENDIENTE, REVISADA o FALSO_POSITIVO). Por regla del sistema, ninguna alerta puede ser incluida en un análisis o reporte si no ha sido revisada previamente, lo cual garantiza un filtro humano sobre el reconocimiento automatizado.

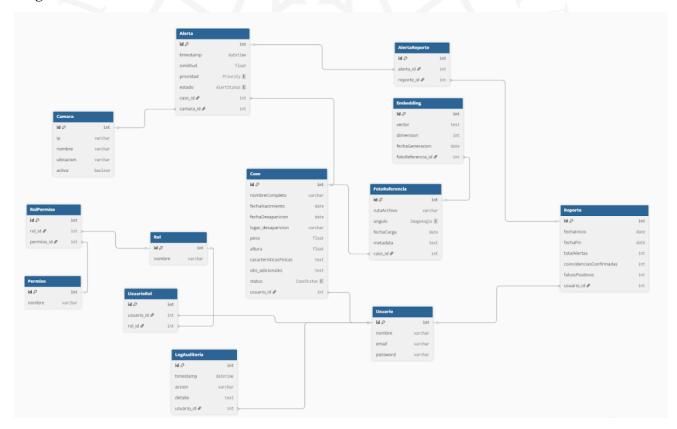
Las alertas pueden ser agrupadas en uno o más Reporte mediante la relación AlertaReporte (N:M), lo cual permite generar informes para análisis por zona, cámara o periodo de tiempo. Toda la actividad que ocurre en el sistema queda registrada en la entidad LogAuditoria, que almacena el tipo de acción realizada, el momento en que se ejecutó, y el Usuario responsable. Esto responde a la regla de trazabilidad total, una política de diseño que asegura la

posibilidad de auditar cualquier acción dentro del sistema, desde la carga de fotos hasta el cambio de estado de un caso.

Finalmente, el sistema de control de acceso se organiza en torno a Usuario, Rol y Permiso, con relaciones intermedias mediante las tablas UsuarioRol y RolPermiso. Esta arquitectura permite definir con precisión qué usuarios pueden ejecutar determinadas acciones, como cargar fotos, modificar estados de casos, o validar alertas. Un usuario puede tener múltiples roles, y cada rol puede agrupar diferentes permisos, lo que ofrece flexibilidad sin sacrificar la seguridad. El cumplimiento de estas reglas de autorización es fundamental en un sistema sensible como este, donde cada decisión puede impactar directamente en la vida de una persona desaparecida.

Figura 5.3

Diagrama de Base de Datos Relacional de FaceFind Lima - Perú





Anexo 1: Product Backlog

https://docs.google.com/document/d/19v9FWk4OCKtJ4XyA9xMR7ijrRCj21epE4ZXwqDBl 500/edit?usp=sharing

Anexo 2: Priorización técnica MoSCoW

https://online.visual-paradigm.com/share.jsp?id=343135323737362d31

Anexo 3: Estimación de Historias de Usuario

https://docs.google.com/spreadsheets/d/1HK1A5teqrotxwCvEo8O2j-Xk9D152i8SEXOPeh6y ECU/edit?usp=sharing

Anexo 4: Roadmap

https://docs.google.com/spreadsheets/d/1mmQVJaJsdzEqiZP-j3ow--ZqHcBKAau5/edit?usp=drive_link&ouid=110380282919676538975&rtpof=true&sd=true

Anexo 5: Base de datos

https://dbdiagram.io/d/6868c3d1f413ba3508649371

Anexo 6: Coevaluación

https://docs.google.com/spreadsheets/d/1AF1Ey7rhKdANhij4l_o_s9KFJ8APzB9yXL0zpZhC Dzo/edit?usp=sharing

REFERENCIAS

- IDEHPUCP. (2024). ¿Cuántos son y dónde están las personas desaparecidas? Una búsqueda sin respuestas. Instituto de Democracia y Derechos Humanos PUCP. https://idehpucp.pucp.edu.pe/boletin-eventos/cuantos-son-y-donde-estan-las-personas-desaparecidas-y-una-busqueda-sin-respuestas/
- IDEHPUCP. (2024). La catástrofe humanitaria de la desaparición en el Perú: Retos para la política de búsqueda de personas desaparecidas. https://idehpucp.pucp.edu.pe/boletin-eventos/la-catastrofe-humanitaria-de-la-desaparicion-en-el-peru-retos-para-la-politica-de-busqueda-de-personas-desaparecidas-28811/
- Infobae. (2024, 18 de abril). 3 295 personas desaparecidas en Perú: Alarmante cifra se registra solo en los dos primeros meses del 2024. https://www.infobae.com/peru/2024/04/18/3295-personas-desaparecidas-en-peru-alar-mante-cifra-se-registra-solo-en-los-dos-primeros-meses-del-2024/
- Sutherland, J., & Schwaber, K. (2020). Guía de Scrum (Spanish South/Latin American version; L. Salazar & M. López, Trad.). ScrumGuides.org. https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Spanish-Latin-South-American.pdf