Universidad ORT Uruguay
Facultad de Ingeniería

SafeCall: Sistema de Detección de Estafas Telefónicas mediante IA

Entregado como requisito para la obtención del título de Ingeniero en Sistemas

Andrés Montaño - 175409

Suren Keushkerian - 141534

Santiago Porro - 194499

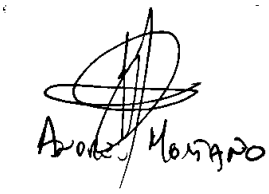
Tutor: Gastón Mousqués

Año: 2025

Declaración de autoría

Nosotros, Andrés Montaño, Santiago Porro y Surén Keushkerian, declaramos que el trabajo que se presenta en esa obra es de nuestra propia mano. Podemos asegurar que:

- La obra fue producida en su totalidad mientras realizábamos el proyecto de grado de Ingeniería en Sistemas;
- Cuando hemos consultado el trabajo publicado por otros, lo hemos atribuido con claridad;
- Cuando hemos citado obras de otros, hemos indicado las fuentes. Con excepción de estas citas, la obra es enteramente nuestra;
- En la obra, hemos acusado recibo de las ayudas recibidas;
- Cuando la obra se basa en trabajo realizado conjuntamente con otros, hemos explicado claramente qué fue contribuido por otros, y qué fue contribuido por nosotros;
- Ninguna parte de este trabajo ha sido publicada previamente a su entrega, excepto donde se han realizado las aclaraciones correspondientes.

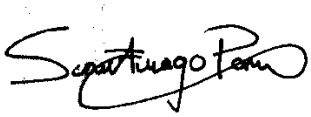


Andrés Montaño

16/10/2025

Surén Keushkerian

16/10/2025



Santiago Porro

16/10/2025

Dedicatoria

Andrés Montaño

A mi pareja Sofía y a mi hijo Guille, por apoyarme siempre.

Santiago Porro

Personalmente, quiero dedicarle el trabajo a mi madre Alicia, a mi padre Héctor y a mi hermana Victoria, por siempre confiar en mí, apoyarme y motivarme a seguir adelante con la carrera, incluso en los momentos difíciles.

Surén Keushkerian

Para mi esposa Noella y mis hijos Sara y Tomás, que supieron motivarme cuando me vieron agotado y agobiado durante todo el proceso. A mi madre Marta y mis hermanas Karla, Susana y Mayra, por alegrarse desde el primer momento cuando comencé la tesis. Especialmente a mi padre Zenón, que seguramente está orgulloso de esta etapa.

Agradecimientos

Nos gustaría agradecer a nuestro tutor, Gastón Mousqués, por su dedicación, paciencia y compromiso durante todo el proceso. Por siempre estar dispuesto a ayudarnos, orientarnos y compartir sus conocimientos.

Abstract

En Uruguay, las estafas telefónicas han crecido de manera alarmante en la última década, afectando especialmente a personas adultas mayores mediante modalidades cada vez más sofisticadas. SafeCall surge como una iniciativa innovadora orientada a proteger a los usuarios frente a estas amenazas, ofreciendo detección de fraudes y notificaciones a los afectados y a sus contactos.

El proyecto se desarrolló bajo un enfoque ágil, combinando fases de exploración y desarrollo iterativo que permitieron adaptar el trabajo a la complejidad y la incertidumbre del desafío. Este enfoque facilitó la identificación de las necesidades de los usuarios, la definición de prioridades y la entrega de resultados funcionales de manera continua, reflejando prácticas profesionales en gestión de proyectos tecnológicos.

El prototipo alcanzó sus objetivos, demostrando que es posible crear un sistema capaz de reconocer patrones de fraude de manera confiable. Más allá del prototipo en sí, el proyecto fortaleció las capacidades del equipo para enfrentar retos complejos, planificar de manera efectiva, colaborar bajo metodologías ágiles y aprender nuevas habilidades que les permitieron estar a la altura del desafío.

Entre los logros más destacados se encuentra la aplicación de metodologías profesionales de gestión, la resiliencia frente a problemas inesperados y la capacidad de desarrollar una solución que permite atacar una problemática relevante y actual, así como también el aprendizaje y aplicación de nuevas tecnologías por el equipo

Palabras Clave

Estafa, Vishing, Spoofing, Smishing, Clonación de voz, Procesamiento de Lenguaje Natural, Inteligencia Artificial, Aprendizaje Automático, Modelos Transformer, XLM-RoBERTa, DistilBERT, Ciberseguridad, Ingeniería de software, Detección de fraudes, Android, Mobile, Ajuste fino, Metodologías ágiles, Scrum, Kanban, Modelos preentrenados, Prototipo funcional.

Índice

1. Introducción

1.1 Contexto y problemática

Las estafas telefónicas en Uruguay han crecido de forma alarmante en la última década. Entre 2013 y 2024 se registró un aumento del 2.000%, pasando de 1.270 denuncias a más de 25.000 casos, lo que convirtió a este tipo de delito en el tercero más reportado del país en 2024. Este fenómeno afecta especialmente a personas adultas mayores, a través de modalidades como el "cuento del tío", el *vishing* (fraude por voz) y, más recientemente, la clonación de voz mediante inteligencia artificial, lo que incrementa su sofisticación y el riesgo para las víctimas. Además, se estima que existe un importante subregistro de casos, ya que muchas personas no denuncian por vergüenza o desconocimiento.

En este contexto surge SafeCall, un proyecto orientado a cubrir una brecha crítica en la ciberseguridad local. Las herramientas actualmente disponibles se enfocan principalmente en bloquear números o recibir reportes después de ocurrido el fraude, sin ofrecer una detección proactiva basada en el contenido de la llamada ni una especialización en patrones de estafa locales. La combinación entre la gravedad social del problema, los avances recientes en inteligencia artificial y un marco legal que promueve la innovación tecnológica crea una oportunidad para desarrollar una solución innovadora, socialmente útil y técnicamente desafiante.

1.2 Descripción general del proyecto

SafeCall es un sistema de detección de estafas telefónicas que utiliza inteligencia artificial para analizar el contenido de las llamadas. La solución consiste en una aplicación móvil Android, que una vez finalizada una llamada analiza el audio y determina si presenta indicios de fraude. En caso de detectar una posible estafa, el sistema notifica automáticamente al usuario y a contactos designados sobre el riesgo.

El proyecto se estructuró en dos grandes etapas de desarrollo. Durante la fase inicial de discovery, se aplicó la metodología ágil Kanban, que permitió gestionar de forma flexible las tareas de investigación, validación de requerimientos y definición del

alcance. En la etapa de *delivery*, orientada al desarrollo del producto, se adoptó la metodología Scrum, organizada en Sprints, lo que facilitó una planificación incremental, la colaboración constante entre los integrantes del equipo y la incorporación temprana de retroalimentación en cada ciclo.

Desde el punto de vista tecnológico, SafeCall se construyó sobre una arquitectura en la nube que permite escalabilidad y fácil mantenimiento. La solución incorpora técnicas de procesamiento de lenguaje natural (NLP, por sus siglas en inglés) y *machine learning* para adaptarse al idioma español y a patrones locales de fraude, asegurando una detección precisa de estafas telefónicas.

El diseño de la aplicación prioriza la performance y usabilidad, especialmente para adultos mayores, garantizando que la experiencia de uso sea clara, intuitiva y confiable.

1.3 Objetivos

Enfoque del prototipo

Dada la naturaleza exploratoria del proyecto, se optó por desarrollar un prototipo técnico y funcional basándonos en las definiciones de acuerdo con el Product School [1]. Esta decisión respondió a la existencia de incertidumbres tanto técnicas como de usuario que requerían validación previa antes de avanzar hacia un producto funcional. El propósito general fue evaluar la factibilidad técnica y la deseabilidad de un sistema capaz de detectar estafas telefónicas mediante inteligencia artificial, orientado a proteger a los usuarios, especialmente a los adultos mayores, de fraudes y engaños telefónicos.

Objetivos técnicos

Desde el punto de vista técnico, se buscó validar la posibilidad de aplicar modelos de procesamiento de lenguaje natural al análisis de conversaciones telefónicas en español, con el fin de detectar patrones característicos del fraude. El objetivo fue comprobar la viabilidad y precisión del enfoque, reduciendo la incertidumbre técnica y sentando las bases para el desarrollo futuro de un producto funcional y escalable.

Objetivos formativos y de gestión

En este plano, el proyecto tuvo como meta consolidar competencias profesionales vinculadas a la planificación, el trabajo colaborativo y la gestión ágil de proyectos tecnológicos. La aplicación combinada de Kanban en la etapa de descubrimiento y Scrum durante el desarrollo permitió simular la dinámica de un entorno profesional real, favoreciendo la adaptación continua y la integración de aprendizajes en cada iteración.

El objetivo fue reproducir la dinámica de trabajo de un entorno real, fomentando la colaboración, la distribución equitativa de tareas y el aprendizaje colectivo. Asimismo, se buscó profundizar en tecnologías de inteligencia artificial, particularmente en procesamiento de lenguaje natural y *machine learning*, áreas de alta relevancia que no habían podido ser abordadas en profundidad durante la formación académica.

Objetivos Orientados a la Experiencia del Usuario

El proyecto también se concibió con un enfoque social y emprendedor, orientado a crear una solución con potencial de impacto positivo en la sociedad uruguaya.

SafeCall apunta a proteger a los adultos mayores, uno de los grupos más vulnerables ante el fraude digital, contribuyendo a la concientización y prevención de estafas telefónicas mediante el uso de tecnología accesible e innovadora.

Además del componente técnico, se buscó evaluar la deseabilidad del producto [2], entendida como el grado en que la solución resulta atractiva, útil y confiable para los usuarios finales. Para ello, se realizó un proceso de identificación del perfil de usuario objetivo y de definición de las funcionalidades que aportaran valor real desde la perspectiva del público.

Estas actividades incluyeron encuestas, entrevistas exploratorias con potenciales usuarios, el análisis de casos reales de fraude y la observación de patrones de comportamiento y respuesta ante llamadas sospechosas. Los resultados permitieron orientar el diseño hacia una experiencia simple, comprensible y empática, especialmente adaptada a las necesidades y limitaciones tecnológicas de los adultos mayores.

Finalmente, el proyecto sienta las bases para una futura validación comercial y escalamiento, abriendo la posibilidad de evolucionar desde un prototipo técnico académico hacia un producto con valor social, sostenibilidad y potencial de adopción masiva.

1.4 Estructura del documento

El presente documento se organiza en once capítulos principales, además de los anexos.

En el capítulo 1, Introducción, se presenta el contexto general del proyecto, la problemática abordada, los objetivos planteados y la motivación detrás de su desarrollo.

El capítulo 2, Descripción del problema, profundiza en el fenómeno de las estafas telefónicas en Uruguay, sus tipos, su impacto social y económico, y la experiencia de las víctimas.

En el capítulo 3, *Product Discovery*, se detalla el proceso de descubrimiento del producto, la metodología utilizada, la investigación secundaria y primaria, así como las pruebas de concepto realizadas.

El capítulo 4, Definición del prototipo funcional, describe los requerimientos funcionales y no funcionales que guiaron el desarrollo del sistema.

En el capítulo 5, Conceptos de Inteligencia Artificial, se exponen los fundamentos teóricos aplicados, el procesamiento del lenguaje natural, el uso de modelos preentrenados y los resultados de las pruebas experimentales.

El capítulo 6, Arquitectura, explica la estructura general del sistema, incluyendo el diseño del *backend*, la aplicación móvil y las principales decisiones de arquitectura adoptadas.

En el capítulo 7, Calidad, se describen las estrategias de evaluación y pruebas, la calidad del código, y la gestión de la configuración del proyecto.

El capítulo 8, Gestión del proyecto, aborda la organización del trabajo, la metodología empleada, la planificación de *releases* y Sprints, y la identificación de riesgos.

En el capítulo 9, Conclusiones, se analiza el cumplimiento de los objetivos, las principales contribuciones del proyecto, las lecciones aprendidas y los desafíos enfrentados. Además se detalla el posible Trabajo futuro donde se presentan las posibles líneas de mejora técnica, expansión de funcionalidades.

2. Descripción del problema

2.1 Estafas telefónicas en Uruguay

Las estafas telefónicas representan uno de los fenómenos criminales de crecimiento más acelerado en Uruguay durante la última década. El informe oficial del Área de Estadística y Criminología Aplicada (AECA) del Ministerio del Interior documenta un incremento exponencial superior a 2.000% entre 2013 y 2024. [5] Las cifras oficiales en el informe muestran una evolución dramática: de 1.333 denuncias en 2013 (38,7 por cada 100.000 habitantes) se incrementó a 31.144 denuncias en 2024 (870,5 por cada 100.000 habitantes).

La progresión cronológica evidencia una aceleración constante del fenómeno: 1.926 denuncias en 2015 (55,6 por 100.000 hab.), 10.893 en 2020 (308,5 por 100.000 hab.), 14.747 en 2021 (416,2 por 100.000 hab.), 19.938 en 2022 (560,9 por 100.000 hab.), 23.418 en 2023 (656,6 por 100.000 hab.) hasta alcanzar las 31.144 denuncias en 2024. Este crecimiento sostenido no solo se refleja en valores absolutos, sino también en el posicionamiento relativo dentro del panorama delictivo nacional: las estafas ascendieron del décimo lugar en 2013 al tercer lugar en 2024 en el *ranking* de delitos más denunciados, representando uno de los saltos más pronunciados en el sistema de registros criminales.

Según el informe de AECA, el aumento se intensificó notoriamente a partir de 2016, coincidiendo con la masificación de tecnologías de comunicación, redes sociales, plataformas digitales de compraventa y canales electrónicos de pago, que ampliaron exponencialmente las oportunidades para estafas remotas o automatizadas. La pandemia de COVID-19 actuó como catalizador adicional, acelerando la digitalización de servicios y comunicaciones, lo que incrementó globalmente las oportunidades para actividades cibernéticas maliciosas.

La magnitud real del problema supera significativamente las estadísticas oficiales debido a un alto subregistro. El módulo de victimización de la Encuesta Continua de Hogares del Instituto Nacional de Estadística estimó que durante el segundo semestre de 2024 ocurrieron aproximadamente 26.900 estafas mensuales en Uruguay. Sin embargo,

solo el 11,7% de estos casos fueron denunciados formalmente y otro 16,2% reportados informalmente a la policía, revelando que la "cifra oculta" representa la mayoría de los casos reales.

El análisis territorial revela una distribución heterogénea pero generalizada en todo el país. Como revela el gráfico a continuación, Rocha presenta el perfil más afectado, manteniendo las tasas más elevadas durante todo el período analizado y alcanzando 1.290,5 denuncias por cada 100.000 habitantes en 2024. No obstante, el AECA advierte que estas variaciones territoriales deben interpretarse con cautela, ya que las estafas involucran frecuentemente interacciones remotas donde autor y víctima pueden encontrarse en ubicaciones geográficas diferentes.

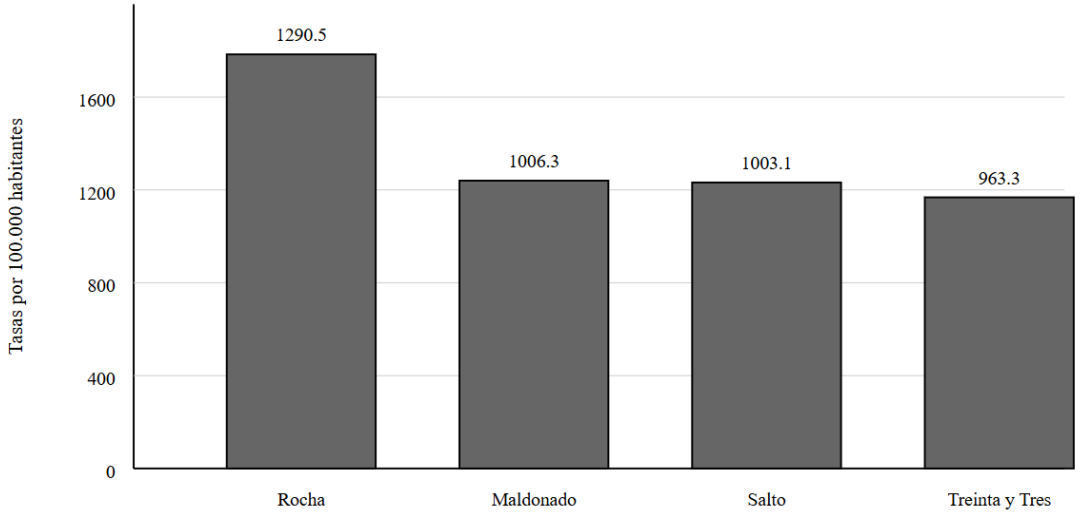


Ilustración 1 Tasa de denuncias por departamento (por cada 100.000 habitantes) - Año 2024

2.2 Tipos de Estafas telefónicas

Las estafas telefónicas en Uruguay han evolucionado considerablemente, diversificándose en múltiples modalidades que explotan tanto las vulnerabilidades tecnológicas como los aspectos psicológicos de las víctimas [5]. Las modalidades más frecuentes identificadas incluyen técnicas tradicionales adaptadas al contexto digital y nuevas metodologías que aprovechan las capacidades de la inteligencia artificial y la ingeniería social.

Estafas Bancarias

El *vishing* (*voice phishing*), término que combina "*voice*" (voz) y "*phishing*" (pesca de datos), representa la modalidad más común en Uruguay, constituyendo aproximadamente el 90% de los casos reportados al Banco de la República Oriental del Uruguay BROU [7]. Esta técnica combina llamadas telefónicas fraudulentas con información previamente obtenida mediante *phishing* (envío de correos o mensajes fraudulentos para obtener datos personales). Los estafadores se hacen pasar por empleados bancarios, en muchas ocasiones utilizando técnicas de *spoofing* (suplantación de voz o del número telefónico que permite mostrar en el identificador un número diferente al real) para que en el identificador del teléfono aparezca el número oficial de la entidad bancaria [8].

El proceso típico incluye dos fases: primero, los ciberdelincuentes obtienen información confidencial a través de correos electrónicos o sitios web fraudulentos; segundo, contactan telefónicamente a la víctima identificándose como personal del banco, utilizando mensajes alarmistas para que revele códigos SMS, tokens digitales o claves de acceso. Los estafadores emplean argumentos como "se ha detectado actividad sospechosa en su cuenta" o "necesitamos verificar sus datos por seguridad" para generar urgencia y reducir el pensamiento crítico de la víctima [9].

Con esa información, luego pueden acceder a las cuentas de las víctimas y así extraer dinero, o incluso también pedir préstamos o créditos a su nombre.

Modalidad "Cuento del Tío" Tecnológico

La tradicional estafa del "cuento del tío" se ha sofisticado incorporando elementos tecnológicos. Los delincuentes contactan a personas mayores haciéndose pasar por familiares directos (hijos o nietos), utilizando técnicas de *voice spoofing* para recrear voces conocidas [10]. La aparición de la inteligencia artificial ha intensificado dramáticamente esta amenaza: herramientas de clonación de voz basadas en IA pueden generar imitaciones convincentes de familiares con apenas unos segundos de audio extraído de redes sociales o mensajes de voz [11]. Estas tecnologías de síntesis de voz, accesibles incluso para usuarios sin conocimientos técnicos avanzados, aumentan significativamente la credibilidad del engaño al reproducir no solo la voz sino también patrones de habla, entonación y expresiones características de la persona suplantada [4].

Las narrativas más comunes incluyen situaciones de emergencia como accidentes, robos o problemas legales que requieren transferencias de dinero inmediatas, incluso en ocasiones manteniendo la comunicación telefónica activa hasta que se concreta la entrega del dinero [10].

Una variante emergente involucra la suplantación de servicios públicos (Obras Sanitarias del Estado OSE, Administración Nacional de Usinas y Trasmisiones Eléctricas UTE, Administración Nacional de Telecomunicaciones Antel), donde los estafadores informan sobre supuestos problemas con facturación o servicios que requieren "actualización de datos" para evitar cortes o multas. Esta modalidad aprovecha la confianza ciudadana en las instituciones públicas y la preocupación por mantener servicios esenciales [12][13].

Estafas de Inversión, Premios y Ofertas Fraudulentas

Los estafadores ofrecen oportunidades de inversión extraordinarias, préstamos preferenciales o productos financieros con condiciones irrealmente favorables. Utilizan técnicas de presión temporal ("oferta válida solo hoy") y referencias a supuestas normativas del Banco Central del Uruguay para generar urgencia y aparente legitimidad [14]. Esta modalidad se ha incrementado significativamente como parte del crecimiento

global de estafas de inversión, que en otros países representan las mayores pérdidas reportadas en fraudes telefónicos [15].

Estas modalidades frecuentemente incluyen solicitudes de "pagos por adelantado" para "liberar fondos" o "activar cuentas", así como pedidos de información bancaria completa para "procesar la operación". Los montos ofrecidos suelen ser sustanciales para maximizar el atractivo de la oferta [14].

Suplantación de Entidades de Salud y Servicios Sociales

Una modalidad en crecimiento involucra la suplantación de instituciones de salud, Banco de Previsión Social BPS o seguros médicos. Los estafadores contactan especialmente a personas mayores informando sobre "beneficios no reclamados", "actualizaciones de datos obligatorias" o "reembolsos pendientes" que requieren confirmación de información personal y bancaria. Esta técnica aprovecha la vulnerabilidad de adultos mayores que frecuentemente interactúan con sistemas de salud y seguridad social [16].

Medios de Comunicación Utilizados

Las estafas telefónicas uruguayas no se limitan a llamadas tradicionales, sino que han diversificado sus canales según la evolución tecnológica global:

Las llamadas telefónicas directas continúan siendo el medio principal, aprovechando la confianza que genera el contacto directo y la capacidad de ejercer presión psicológica inmediata [16]. Este método mantiene su efectividad debido a que permite la manipulación en tiempo real de las emociones de la víctima [17].

Por otra parte, los estafadores utilizan WhatsApp para llamar o enviar mensajes de texto (*smishing*, término que combina "SMS" y "phishing"). Estos mensajes pueden incluir enlaces fraudulentos, códigos QR maliciosos o solicitudes directas de transferencias de dinero [18].

El *smishing* también se utiliza para enviar alertas falsas de entidades bancarias, servicios públicos o empresas reconocidas, solicitando llamar a números fraudulentos o acceder a

sitios web maliciosos [9]. Esta técnica permite a los estafadores modificar el número emisor del mensaje (SMS *spoofing*) para que aparente provenir de fuentes confiables.

Por último, los delincuentes utilizan perfiles falsos en Facebook, Instagram y otras plataformas para establecer confianza antes de proceder con el engaño telefónico [9], aprovechando información personal disponible públicamente para personalizar sus aproximaciones y aumentar la credibilidad de sus comunicaciones [14].

2.3 Impacto Social y Económico

El impacto de las estafas telefónicas en Uruguay trasciende las pérdidas económicas directas, generando consecuencias profundas incluso a nivel social y en la confianza general de la población. El análisis multidimensional revela la magnitud real de esta problemática y sus efectos de largo plazo en la sociedad uruguaya.

Pérdidas del Sistema Bancario

Las cifras oficiales del Banco República del Uruguay (BROU) revelan la magnitud económica del problema. Entre 2019 y 2023, la institución recibió 2.653 denuncias por estafas y fraudes cibernéticos, logrando prevenir o recuperar dinero en 936 casos (35,2% del total) por un monto de USD 5.499.510. Sin embargo, en los 1.717 casos restantes donde no fue posible impedir la estafa, las pérdidas ascendieron a USD 1.448.464 [19].

La cifra total de transacciones digitales fraudulentas desde 2020 alcanza los USD 2,8 millones, afectando aproximadamente 600 clientes del BROU, con un promedio de USD 4.667 por cliente victimizado [7]. Estas cifras representan únicamente los casos denunciados en una institución bancaria, por lo que el impacto real del sistema financiero uruguayo es considerablemente mayor [19].

Pérdidas Estimadas del Sector Privado

Estudios del Centro de Estudios de la Realidad Económica y Social (CERES) indican que, en los registros del Ministerio del Interior de 2024, aproximadamente un tercio de los casos de estafa cuantifican montos denunciados que ascienden a más de 27 millones

de pesos uruguayos y cerca de USD 230.000 [4]. Sin embargo, estas cifras representan únicamente los casos denunciados y cuantificados.

El alto subregistro identificado por el Instituto Nacional de Estadística, donde solo el 11,7% de las estafas reales son denunciadas formalmente y otro 16,2% son reportadas informalmente [5], sugiere que las pérdidas económicas reales del sistema son significativamente superiores a las cifras oficiales disponibles [4].

Costos Operativos para Entidades Financieras

Las instituciones bancarias han debido incrementar sustancialmente sus inversiones en ciberseguridad y sistemas de detección de fraudes [20]. El BROU implementó sistemas de monitoreo que le permitieron interceptar el 60% de los intentos de fraude en 2023, logrando evitar que se concreten las estafas en esos casos. Esta mejora en la capacidad de detección requiere inversiones tecnológicas continuas y capacitación especializada del personal [20].

Los costos administrativos asociados al procesamiento de denuncias, investigaciones internas y atención a víctimas representan una carga operativa adicional para todas las entidades del sistema financiero. Cada caso de fraude requiere procesos de investigación interna, coordinación con autoridades e implementación de medidas correctivas [19].

Vulnerabilidad Demográfica

El análisis de las víctimas revela patrones demográficos específicos que intensifican el impacto social. Según datos del Ministerio del Interior disponibles en ANEXO 1, entre enero de 2019 y julio de 2024 se registraron 7.872 denuncias por estafas telefónicas, de las cuales 5.297 fueron dirigidas a mujeres (67%). Esta sobrerrepresentación femenina se explica por factores culturales, mayor longevidad y, frecuentemente, menor familiaridad con tecnologías digitales.

El perfil etario muestra una concentración en personas mayores de 70 años, quienes representan el grupo más victimizado debido a factores como menor conocimiento tecnológico, mayor confianza en instituciones tradicionales y situaciones de aislamiento social que facilitan la manipulación psicológica.

Impacto social y emocional

Más allá de las pérdidas económicas, las víctimas enfrentan consecuencias psicológicas severas que a menudo son subestimadas. Las investigaciones internacionales y nacionales documentan que las víctimas experimentan sentimientos profundos de culpa, vergüenza, traición y pérdida de autoestima [21]. Un alto porcentaje desarrolla desconfianza hacia las instituciones financieras y evita utilizar servicios donde ocurrió el fraude [22].

Las consecuencias emocionales incluyen depresión, ansiedad severa, problemas de sueño y, en casos extremos, pensamientos suicidas [22]. Esta sintomatología puede persistir durante meses o años, impactando la calidad de vida y la capacidad de trabajo de las personas afectadas [23]. El aislamiento social es común, ya que las víctimas evitan compartir sus experiencias por miedo al juicio social [22].

El testimonio de víctimas entrevistadas refleja el impacto emocional: "Me siento como una tonta por haber caído en eso. Ahora desconfío de todo el mundo, hasta de las llamadas de mis propios hijos" - expresa Irma una jubilada de 82 años que estuvo a punto de perder sus ahorros de toda la vida en una estafa bancaria. Estos casos documentados ilustran cómo las estafas generan un impacto que va más allá de lo económico, afectando la confianza interpersonal y la autonomía de las personas mayores.

Perspectivas de Evolución

Las tendencias internacionales indican una sofisticación creciente de las estafas telefónicas mediante el uso de inteligencia artificial. Los ciberdelincuentes han comenzado a emplear técnicas como la generación de vídeos fraudulentos con IA y la suplantación de identidad mediante tecnologías de síntesis de voz, aunque estas modalidades aún no son comunes, pero indican la dirección que están tomando las nuevas tácticas.

En Uruguay, el aumento se intensificó notoriamente a partir de 2016, coincidiendo con la masificación de tecnologías de comunicación, y la pandemia de COVID-19 actuó

como catalizador adicional, acelerando la digitalización de servicios y aumentando globalmente las oportunidades para actividades cibernéticas maliciosas [5].

La implementación de soluciones tecnológicas preventivas representa una estrategia para contener estas tendencias, especialmente considerando que las estafas se consideran actualmente delitos predatorios, cometidos en su mayoría desde el anonimato y con carácter transnacional, con una participación creciente del crimen organizado [5]

2.4 La experiencia de las víctimas: ¿Qué pueden hacer y qué pueden recuperar?

Cuando una persona es víctima de una estafa telefónica en Uruguay, se enfrenta a un proceso complejo con opciones limitadas de recuperación y un impacto que trasciende lo económico. El análisis de la situación actual revela que las víctimas viven una experiencia caracterizada por la incertidumbre, la frustración y consecuencias psicológicas significativas.

Opciones disponibles para las víctimas

Denuncia formal**:** La principal herramienta disponible para las víctimas es la presentación de una denuncia ante las autoridades competentes. Pueden dirigirse a la seccional policial más cercana, al Departamento de Delitos Financieros de Interpol (Maldonado 1109, 2° piso) o a la Unidad de Cibercrimen del Ministerio del Interior (Maldonado 1109, 1° piso). También existe la posibilidad de realizar denuncias en línea a través del sistema del Ministerio del Interior, aunque estas requieren confirmación presencial posterior.

Reclamo bancario**:** Las víctimas deben iniciar reclamos inmediatos ante sus entidades bancarias, proceso que resulta muy importante para las posibilidades de recuperación. Según datos del BROU, la detección temprana mejora significativamente las chances de bloquear transferencias fraudulentas [24].

Realidad de la recuperación económica

Las estadísticas oficiales revelan un panorama desalentador en términos de recuperación. Según datos del BROU para el período 2019-2023, de 2.653 denuncias recibidas por estafas, solo se logró recuperar el dinero en 936 casos, representando únicamente el 35,2% del total [24]. El monto recuperado ascendió a USD 5.499.510, mientras que en los 1.717 casos donde no fue posible impedir la estafa, las pérdidas sumaron USD 1.448.464 [24].

Esta baja tasa de recuperación se explica por la naturaleza misma de las estafas telefónicas: una vez que la víctima transfiere voluntariamente el dinero tras ser engañada, los fondos en general son movidos con rapidez a través de "mulas" (intermediarios que retiran el dinero y lo envían al exterior) [5], dificultando enormemente su rastreo y recuperación.

Por otra parte, el fenómeno de la falta de denuncias o subregistro amplifica la problemática. Según la Encuesta Continua de Hogares del INE, solo el 11,7% de las estafas son efectivamente denunciadas, mientras que un 16,2% adicional es reportado informalmente [5]. Esto significa que aproximadamente el 72% de las estafas nunca llegan al conocimiento oficial, perpetuando el ciclo de impunidad y limitando las posibilidades de justicia para las víctimas.

Respuesta institucional limitada

El sistema de respuesta penal muestra limitaciones estructurales significativas. Según datos del AECA, en 2024 se registraron apenas 1,5 imputaciones por cada 100 denuncias de estafa [5], evidenciando que la gran mayoría de casos quedan impunes. Esta baja tasa de esclarecimiento refuerza la sensación de desprotección de las víctimas y reduce los incentivos para denunciar.

La experiencia típica de una víctima de estafa telefónica en Uruguay se caracteriza, por tanto, por opciones limitadas de acción, bajas probabilidades de recuperación económica, impacto psicológico significativo y escasas posibilidades de obtener justicia a través del sistema penal.

3. Product Discovery

3.1 Design Thinking en el desarrollo del producto

Para el desarrollo de SafeCall se adoptó *Design Thinking* como metodología principal, priorizando un enfoque centrado en el usuario que favoreciera el desarrollo con diferentes prototipos y validación rápida. Esta elección se fundamentó en la naturaleza compleja y multidimensional del problema de las estafas telefónicas, que requería una comprensión profunda de las necesidades, comportamientos y contextos de uso de las personas vulnerables a estos delitos.

Design Thinking resulta particularmente apropiado para abordar problemas complejos que involucran factores humanos, tecnológicos y sociales [25]. En el caso de las estafas telefónicas, la solución no solo requiere de un fuerte componente tecnológico, sino también una comprensión más profunda de los usuarios objetivo, especialmente adultos mayores que pueden tener limitaciones tecnológicas y patrones de comportamiento específicos ante situaciones de estrés.

La metodología permite combinar métodos cuantitativos (encuestas, estadísticas oficiales, investigación de escritorio) con enfoques cualitativos (entrevistas con usuarios, observación), permitiendo comprender el problema desde múltiples perspectivas. Esta combinación resulta esencial cuando se desarrollan soluciones para poblaciones vulnerables, donde los datos numéricos deben complementarse con *insights* sobre experiencias, emociones y contextos de uso reales.

El proceso de *Design Thinking* aplicado según el "Design thinking bootleg" de Stanford d.school [26] se basó en cinco fases principales:

Empatía:

Comprensión profunda de los usuarios afectados por estafas telefónicas mediante investigación secundaria (análisis de escritorio, estadísticas oficiales, artículos periodísticos) e investigación primaria que incluyó una encuesta cuantitativa y entrevistas con usuarios, víctimas y expertos (INTERPOL, Administración Nacional de Telecomunicaciones ANTEL, ISBEL y Unidad Reguladora y de Control de Datos

Personales URCDP). Durante las entrevistas se aplicó la técnica de "Compartir y Capturar", donde cada miembro del equipo compartió las historias y descubrimientos sobre el trabajo de campo, capturando la experiencia del usuario desde diferentes perspectivas.

Definición:

Síntesis de los descubrimientos y notas mediante la técnica de "Saturar y Agrupar", mediante la cual se saturó el espacio de trabajo (tablero de Miró [27]) con notas y observaciones del trabajo de campo, para luego agruparlas por temas e identificar patrones recurrentes. Este proceso permitió definir dos perfiles de usuario específicos (Perfil 1: usuarios vulnerables y Perfil 2: usuarios tecnológicamente competentes) mediante la creación de *User Personas*.[28]

Ideación:

Generación de tres conceptos alternativos de solución mediante sesiones de *brainstorming* o lluvia de ideas (realizado en rondas de 3 minutos cada una donde cada miembro del equipo generaba diferentes ideas sin juzgarlas), seguido de evaluación comparativa utilizando la Matriz de Pugh con criterios técnicos y de negocio [29]. Por cada uno de los conceptos se realizaron pruebas técnicas de factibilidad y de análisis de competencia de soluciones comerciales existentes (Truecaller, Hiya, RoboKiller, etc.) para identificar brechas de mercado (este análisis se encuentra disponible en el ANEXO 1). A partir del análisis y considerando los mejores aspectos de los otros anteriores, se generó un concepto final a trabajar.

Prototipación:

Desarrollo de un prototipo de baja fidelidad utilizando Proto.io [30], permitiendo simular flujos de usuario sin requerir desarrollo de código. El prototipo buscaba obtener *feedback* temprano, facilitando modificaciones rápidas de funcionalidades y validación de conceptos.

Pruebas:

Validación mediante múltiples técnicas complementarias: Mago de Oz (se simulaba el comportamiento del sistema mediante intervención manual),[31] *Think Aloud* (usuarios verbalizan pensamientos mientras navegan) [32], pruebas de usabilidad basadas en tareas (usuarios ejecutan escenarios específicos) [33] y observación (registro de gestos, expresiones y patrones de navegación) [34]. El refinamiento de la propuesta se basó en *feedback* cualitativo y cuantitativo obtenido durante estas sesiones.

3.2 Investigación Secundaria

La fase de investigación secundaria constituyó el punto de partida para comprender la magnitud, características y tendencias de las estafas telefónicas en Uruguay, estableciendo el contexto necesario para la investigación primaria posterior.

Se realizaron múltiples instancias de análisis de distintas fuentes de información para construir una base de conocimiento sólida sobre la problemática. Esta revisión incluyó artículos periodísticos de medios nacionales, informes de organismos oficiales (como el Ministerio del Interior) y documentación técnica sobre tecnologías de detección de fraudes.

La búsqueda se centró en identificar patrones de comportamiento de estafadores, modalidades de fraude más frecuentes, perfiles de víctimas más vulnerables y soluciones tecnológicas existentes a nivel internacional.

Particularmente, se realizó una solicitud formal de acceso a la información pública al Ministerio del Interior bajo el Expediente N° 2024-4-1-0004676, la cual fue tramitada a través de la Asesoría Letrada del Señor Ministro del Interior junto con el Observatorio Nacional sobre Violencia y Criminalidad Uruguay el 15 de agosto de 2024.

La solicitud específica requirió datos sobre:

- Cantidad de denuncias de estafas telefónicas o intentos de estafas en los últimos 5 años
- Monto total denunciado o de pérdidas económicas
- Cantidad de denuncias discriminadas por género
- Edades de los afectados o denunciantes
- Modalidades específicas denunciadas
- Operatoria utilizada para solicitar el dinero.

Esta gestión permitió obtener datos de primera mano sobre la evolución del fenómeno en el país, complementando la información pública disponible con estadísticas desagregadas por modalidad, rango etario y distribución geográfica. (Detalle del pedido formal de información en el ANEXO 1)

Los datos revelaron que entre el 1° de enero de 2019 al 31 de julio de 2024 se presentaron 7.872 denuncias por estafas telefónicas, información que se utilizó para dimensionar la muestra objetivo de las encuestas y establecer la representatividad estadística requerida.

La respuesta oficial confirmó que el 67% de las denuncias fueron dirigidas a mujeres (5.297 de 7.872 casos), patrón que orientó la segmentación demográfica de la investigación primaria y validó el enfoque en adultos mayores como segmento objetivo.

Además de ello, se analizaron fuentes periodísticas y estudios disponibles que documentaban el crecimiento del problema. Las fuentes consultadas incluían reportes de medios nacionales sobre casos específicos, declaraciones oficiales del BROU sobre impacto en sus clientes, y análisis preliminares que anticipaban la problemática creciente.

Esta investigación secundaria resultó fundamental para establecer la magnitud y relevancia del problema en Uruguay. Las cifras oficiales de denuncias obtenidas del Ministerio del Interior y los reportes del BROU confirmaron que las estafas telefónicas representan una amenaza creciente y significativa para la población, validando la necesidad de desarrollar soluciones tecnológicas para combatirlas.

Sin embargo, la información disponible presentaba limitaciones importantes para el diseño de una solución específica. Los datos oficiales no incluían:

- Desglose por modalidades específicas de estafa más frecuentes
- Análisis de comportamientos y patrones de respuesta de las víctimas durante el ataque
- Medidas de protección actualmente utilizadas por la población
- Percepción de riesgo y disposición a adoptar soluciones tecnológicas

Dicha información, seria recabada directamente de los usuarios potenciales a través de la investigación primaria llevada adelante.

3.3 Investigación Primaria

La investigación primaria se diseñó para obtener tanto información cuantitativa como cualitativa sobre la problemática, complementando los datos secundarios con *insights* directos de usuarios potenciales y expertos del sector.

Encuesta

Se implementó una encuesta dirigida a personas de todas las edades con conocimientos variados de tecnología, buscando obtener una visión cuantitativa del problema y su distribución en diferentes segmentos demográficos.

El instrumento fue diseñado para responder interrogantes sobre aspectos no documentados en fuentes secundarias, enfocándose en ocho dimensiones principales: prevalencia y frecuencia de intentos de estafa, perfil demográfico de las víctimas, modalidades más comunes en Uruguay, señales de alerta identificadas por los usuarios, comportamientos ante llamadas sospechosas, métodos de protección actualmente utilizados, características valoradas en una potencial solución y barreras percibidas para su adopción. El detalle de estas dimensiones y su justificación se presenta en el ANEXO 1.

Resultados Cuantitativos Principales

La encuesta obtuvo más de 100 respuestas (cuyo detalle se puede visualizar en ANEXO 1), un volumen considerable que permitió identificar patrones significativos en el comportamiento de los usuarios.

Mediante ella, se abordaron las siguientes dimensiones del problema: prevalencia y frecuencia de intentos de estafa, perfil demográfico de las víctimas, modalidades más comunes, señales de alerta identificadas por los usuarios, comportamientos ante llamadas desconocidas, métodos de protección actuales, demanda y características valoradas en una solución, y potenciales barreras de adopción de nuevas soluciones.

Los resultados revelaron que las estafas telefónicas constituyen un problema transversal que afecta a todos los rangos etarios de manera relativamente uniforme, contradiciendo la percepción inicial de que se trataba de un problema exclusivo de adultos mayores.

El análisis de la muestra por rangos etarios evidenció la transversalidad del problema:

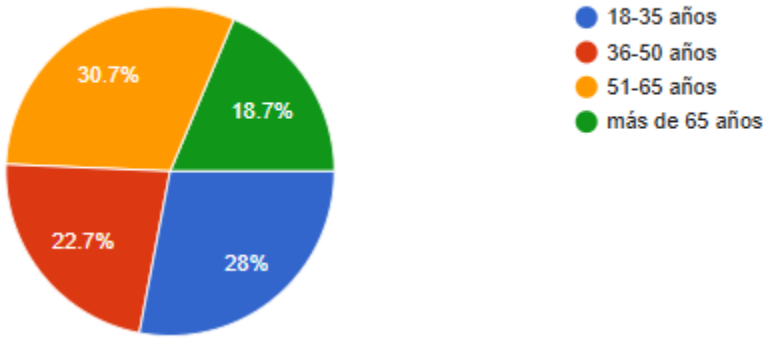


Ilustración 2 Distribución por edad de víctimas de intentos de estafa telefónica según encuesta realizada

Esta distribución confirma que el problema afecta a todos los rangos de edad de manera relativamente equitativa, con una ligera concentración en el segmento de 51 a 65 años.

Respecto a la prevalencia del problema, el 65 % de los encuestados reportó haber recibido llamadas que percibían como intentos de estafa, mientras que más del 50 % recibe estas llamadas con frecuencia.

Los participantes identificaron las modalidades más comunes como premios o sorteos falsos (33%), suplantación de entidades bancarias o financieras (23%), falsos técnicos de servicios públicos o privados (18,6%), y la modalidad del "cuento del tío" modernizado con falsos familiares (13%).

En cuanto a los patrones de sospecha y detección, los usuarios identifican como principales señales de alerta los números extranjeros o desconocidos, solicitudes de información personal como datos bancarios, presión temporal para tomar decisiones inmediatas, y suplantación de bancos, entidades estatales o falsos familiares.

El análisis de frecuencia de contacto reveló patrones preocupantes donde 3 de cada 10 usuarios contestan siempre llamadas de números desconocidos, otros 2 de cada 10 solo contestan si esperan una llamada, y los 5 restantes nunca contestan llamadas de números desconocidos.

Medidas de Protección Actuales y Demanda de Soluciones

El estudio reveló una brecha significativa en la adopción de medidas preventivas, donde el 43% no utiliza ninguna medida de protección, el 47% utiliza bloqueo de números, solo el 3% utiliza aplicaciones de identificación, y el 7% mantiene listas de números no deseados.

Los resultados evidenciaron una fuerte demanda por soluciones tecnológicas. Las funcionalidades más valoradas incluyen: identificación automática de llamadas sospechosas (70%), base de datos o registro de números reportados como sospechosos (51%), bloqueo automático de números sospechosos (43%), reporte directo a autoridades (40%), y notificaciones inteligentes (31%).

Sin embargo, los participantes expresaron inquietudes específicas sobre soluciones tecnológicas. El costo resultó ser un factor determinante, ya que 8 de cada 10 participantes indicaron que no pagarían por una solución, estableciendo la gratuidad como requisito fundamental para la adopción. También expresaron preocupación por falsos positivos que bloqueen llamadas legítimas por error, reticencia a compartir datos sobre llamadas con terceros, e inquietud sobre la compatibilidad con diferentes modelos de teléfonos.

Estos datos evidenciaron tanto la prevalencia del problema como la existencia de una brecha significativa en la adopción de medidas preventivas, validando la necesidad de desarrollar soluciones accesibles, gratuitas y efectivas. La alta demanda por funcionalidades específicas proporcionó una guía clara para la priorización de características en el desarrollo del Prototipo Funcional.

Entrevistas Cualitativas

Se realizaron 8 entrevistas en profundidad con usuarios que habían experimentado intentos de estafa telefónica, aplicando técnicas específicas de *Design Thinking* descritas en la sección 3.1. para maximizar la extracción y organización de *insights*. El proceso siguió una metodología estructurada que incluyó tanto entrevistas con usuarios objetivo y víctimas como entrevistas con expertos del sector. (Detalles de las entrevistas realizadas en ANEXO 1)

Resultados de las técnicas de Empatía

Como resultado del análisis del proceso de Empatía, se agruparon las notas y observaciones en seis temas principales que emergieron de manera consistente a través de todos los casos estudiados: Modalidad (tipos de estafas experimentadas), Medidas de Protección Actuales (estrategias defensivas implementadas por los usuarios), Preferencias sobre posible Solución (expectativas y funcionalidades deseadas), Restricciones (límites de privacidad y funcionalidad), Confiabilidad (factores que generan confianza en la solución), y Manejo de Tecnologías (niveles de competencia tecnológica de los usuarios). Esta categorización permitió identificar patrones transversales y diferencias significativas entre los grupos de usuarios entrevistados. (Detalle de tablero de Miro en ANEXO 1)

Modalidad

En cuanto a modalidades, se identificaron dos grupos principales. El Grupo 1 corresponde a víctimas directas que experimentaron el "cuento del tío", estafas bancarias con pérdidas significativas, estafas en Facebook Marketplace [35] y transferencias falsas, así como casos donde la presencia de familiares jóvenes interrumpió el intento. El Grupo 2 incluye víctimas indirectas que conocen casos

cercanos de estafas por WhatsApp [36] y suplantación de entidades públicas, pero no han sido víctimas directas, aunque están conscientes del riesgo creciente.

Los patrones identificados muestran que los estafadores atacan principalmente cuando detectan adultos mayores solos, utilizan múltiples canales como llamadas telefónicas, WhatsApp y mensajes SMS, emplean información personal previamente obtenida para generar confianza, y aprovechan momentos específicos como viernes a la tarde durante el cierre bancario.

Medidas de Protección Actuales

Respecto a las medidas de protección actuales, se detectó que la gran mayoría de los entrevistados no cuenta con medidas de protección significativas y muestran una pérdida de confianza generalizada en llamadas telefónicas.

Las limitaciones identificadas revelan que las medidas actuales son reactivas en lugar de preventivas, existe una gran dependencia de conocimiento tecnológico para aplicar ciertas medidas que muchos usuarios no poseen, y se presenta la paradoja de que quienes más necesitan protección son quienes menos herramientas técnicas utilizan.

Preferencias sobre Posible Solución

Los entrevistados expresaron un consenso general sobre la necesidad universal de soluciones que detecten automáticamente intentos de estafa, preferencia por soluciones que no requieran intervención manual durante la llamada, expectativa de notificación inmediata a contactos de confianza o familiares, y deseo de identificación de números reportados como peligrosos o sospechosos.

En particular, los adultos mayores entrevistados manifestaron la necesidad de contar con una solución con una interfaz extremadamente simple e intuitiva, que opere tanto en casa como en espacios públicos y que la notificación automática a su red familiar no requiriera configuración compleja.

Restricciones

En términos de restricciones, los usuarios manifestaron negativa total al acceso a la galería personal de imágenes y videos en sus dispositivos, prohibición de acceso a contraseñas almacenadas, exigencia de políticas de privacidad que indiquen que datos personales se van a utilizar y con qué motivo, limitación del acceso a información no relacionada con llamadas, control sobre qué acciones puede realizar la aplicación, y uso del micrófono limitado exclusivamente a momentos de llamada.

Confiabilidad

Los factores de confiabilidad identificados incluyen respaldo de entidades telefónicas reconocidas como ANTEL, validación por organismos regulatorios como la Unidad Reguladora de Servicios de Comunicaciones URSEC, asociación con instituciones de seguridad o empresas internacionales establecidas, y transparencia en procedencia y funcionamiento de la solución.

En particular, los adultos mayores entrevistados manifestaron que su principal fuente de confianza vendría dada por la recomendación de personas de confianza como factor decisivo para adoptar y utilizar una solución, además de su simplicidad de uso.

En cuanto a los adultos jóvenes o de edad media, su confianza viene dada por la transparencia completa sobre el almacenamiento y uso de sus datos personales y por el respaldo institucional de la solución en entidades conocidas y establecidas.

Manejo de Tecnologías

El análisis del manejo tecnológico reveló una brecha significativa entre los diferentes grupos de usuarios entrevistados. El Grupo 1, caracterizado por conocimiento básico y limitado, muestra un uso muy rudimentario de tecnología con manejo básico del celular limitado a funciones esenciales. Aunque poseen smartphones de diversa gama o antigüedad, sus conocimientos son limitados y requieren soluciones que funcionen de manera completamente transparente, necesitando que la tecnología opere automáticamente sin requerir aprendizaje complejo.

Por el contrario, el Grupo 2 demuestra alto conocimiento y dominio de tecnologías, con uso profesional y personal avanzado de herramientas digitales. Este grupo muestra preferencia por aplicaciones configurables pero de uso simple, y posee la capacidad de comprender y evaluar aspectos técnicos de las soluciones propuestas.

La brecha tecnológica identificada presenta una paradoja: los usuarios que más necesitan protección son precisamente quienes menos dominan las herramientas disponibles. Se observó que la vulnerabilidad a estafas es indirectamente proporcional a la capacidad de usar soluciones tecnológicas complejas, lo que establece que la solución debe adaptarse al nivel tecnológico más básico para ser efectiva con el grupo más vulnerable.

Perfiles de Usuarios Identificados

Los perfiles de usuario son representaciones de grupos con características, necesidades y comportamientos similares, derivadas del análisis de datos cualitativos recopilados durante las entrevistas. Estos perfiles permiten sintetizar patrones recurrentes y facilitan la toma de decisiones de diseño centradas en el usuario.

La derivación de perfiles se realizó mediante las técnicas de síntesis de la fase de Definición descritas en el capítulo 3.1, identificando patrones en dolores, necesidades, nivel tecnológico y actitudes hacia las estafas, agrupando a los usuarios según similitudes en estos aspectos (Por análisis detallado de derivación dirigirse al ANEXO 1).

Basándose en el análisis de los resultados, se identificaron dos perfiles principales:

Perfil 1: Usuarios Adultos mayores Vulnerables

Corresponde a usuarios vulnerables, principalmente adultos mayores con conocimientos variados de tecnología y que han sido víctimas directas o tienen experiencias cercanas de estafas telefónicas.

Su principal dolor radica en el miedo de sufrir una estafa, perder dinero y sentirse avergonzados de ello, además del temor a que su información sea utilizada con fines maliciosos. Experimentan pérdida de confianza personal, aislamiento autoimpuesto al

evitar contestar llamadas, vulnerabilidad tecnológica para distinguir llamadas legítimas, ansiedad constante sobre posibles estafas, y pérdida de autonomía al necesitar depender de familiares para validar situaciones.

Los *insights* clave de este perfil revelan que desconfían del uso de su información personal por parte de las aplicaciones y no quieren sumar un gasto extra a su rutina diaria por un servicio que puede no ser utilizado con frecuencia. Además, necesitan que sea alguien en quien confían quien instale la solución y requieren que la familia esté presente para ayudarlos. Muestran una clara preferencia por detección automática sin necesidad de acciones manuales.

Sus necesidades específicas se centran en tres aspectos fundamentales: protección automática expresada como "que el aparato se dé cuenta solo de la estafa y avise a la familia" según Raúl, una red de apoyo familiar donde "mi familia puede saber que estoy en peligro cuando tengo la emergencia" también según Raúl, y simplicidad extrema resumida por Sonia como "muy fácil de usar".

Perfil 2: Usuarios Tecnológicamente Competentes

Incluye usuarios tecnológicamente competentes, principalmente adultos jóvenes o de edad media, con conocimientos avanzados de tecnología y sin gran preocupación sobre ser estafados debido a su confianza en su habilidad para detectar estas situaciones.

Su principal dolor es recibir muchas llamadas de números desconocidos y no querer ser molestados continuamente. Experimentan interrupciones laborales por llamadas durante el trabajo, falsa confianza que los lleva a subestimar el riesgo, y falta de uso de herramientas preventivas disponibles.

Los *insights* clave de este perfil muestran que, al igual que el Perfil 1, desconfían del uso de su información personal por parte de las aplicaciones y no quieren sumar un gasto extra a su rutina diaria por un servicio que puede no ser utilizado con frecuencia. Sin embargo, se diferencian en que valoran especialmente la transparencia en el manejo de datos personales y prefieren tener control sobre las funcionalidades de la solución.

Sus necesidades específicas se enfocan en cinco aspectos principales: control granular para entender y controlar cómo funciona la solución, información preventiva sobre modalidades actuales de estafa, sistemas de base de datos colaborativa que mejoren con el uso comunitario, verificación bidireccional mediante métodos de autenticación como 2FA (*Two-factor Authentication*) para entidades legítimas, y filtrado inteligente para reducir interrupciones por llamadas no deseadas.

Entrevistas con Expertos

Además de las entrevistas con usuarios, se realizaron 4 entrevistas especializadas para validar tanto el problema como las aproximaciones técnicas y legales de la solución propuesta, obteniendo perspectivas de expertos en seguridad, telecomunicaciones y regulación de datos (evidencia de las entrevistas en el ANEXO 1).

Confirmación del Problema con INTERPOL

Se llevó a cabo una entrevista presencial con Pablo Tafernaberry, Jefe del departamento de investigación de delitos financieros de INTERPOL. Esta entrevista confirmó que la problemática es real y creciente, reforzando las hipótesis iniciales del proyecto.

Tafernaberry explicó que los delincuentes actualizan constantemente sus métodos de operación, adaptándose a las medidas de protección implementadas por usuarios y entidades. Confirmó que el problema se ha extendido significativamente a WhatsApp y otras redes sociales, pero destacó que atacar el problema desde las llamadas telefónicas constituye un punto de partida estratégico válido.

Un punto importante de esta entrevista fue la confirmación de que existe un subregistro (faltante de denuncias) masivo de casos debido a la vergüenza de las víctimas y el desconocimiento sobre cómo proceder tras un intento de estafa. Esta información validó la importancia de desarrollar herramientas que no solo protejan, sino que también faciliten el reporte y seguimiento de casos.

Análisis de Viabilidad Técnica con Telecomunicaciones

La validación técnica se realizó mediante dos entrevistas especializadas que proporcionan perspectivas complementarias sobre la viabilidad técnica de la solución.

En particular, se entrevistó a José Joskowicz, experto en telecomunicaciones de ISBEL y docente de la Universidad de la República (UDELAR). Joskowicz confirmó la factibilidad técnica de sistemas de análisis de audio en tiempo real, aunque indicó que, para ello resultaría necesario contar con herramientas de *Speech Analytics* o también de infraestructura especializada como, por ejemplo, basada en Asterisk [3] que permitan obtener el contenido de la llamada en el momento.

Además, confirmó la viabilidad de integración con operadoras telefónicas y explicó que los marcos regulatorios existentes permiten el procesamiento de llamadas para fines de detección de fraudes, siempre que se cumplan los protocolos de privacidad correspondientes.

Complementariamente, se realizó una entrevista con Gonzalo Escuder, Gerente de operaciones móviles de ANTEL. Esta entrevista resultó en la apertura a colaboraciones futuras por parte de la operadora estatal, confirmación operativa de la problemática desde la perspectiva de la principal operadora del país, y validación de la necesidad de soluciones especializadas para abordar el creciente volumen de estafas telefónicas.

Marco Legal y Regulatorio con URCDP

Además de las anteriores, se llevó a cabo una entrevista con la Unidad Reguladora de Control de Datos Personales (URCDP), con la participación de Lylian Massarino y Flavia Baladan del equipo técnico. Esta entrevista estableció el marco legal completo que debe cumplir SafeCall para operar en Uruguay.

En cuanto al marco regulatorio para inteligencia artificial, se estableció que para el procesamiento de IA es necesario anonimizar las llamadas sin identificar a la persona que realiza la llamada, y que es imperativo disociar correctamente los datos para evitar generar sesgos en los modelos de *machine learning* y sobre todo cuidar la privacidad de los usuarios.

Las restricciones de transferencia de datos a nivel internacional resultaron ser particularmente restrictivas. Se prohíbe la transferencia de datos a países que no tengan niveles adecuados de protección de datos personales, y específicamente se confirmó que Estados Unidos no está incluido en la lista de países con protección adecuada. En caso

de requerir alojar información en países sin protección adecuada, se debe solicitar autorización previa a URCDP, aunque se recomienda fuertemente alojar la base de datos en Uruguay.

Los requerimientos de seguridad establecen la obligación de fijar los datos en un área segura con especial atención a la redundancia, inscribir obligatoriamente la base de datos en URCDP según la Ley 18.331 Art. 13 [37], y asegurar que si se elige a Amazon Web Services AWS u otros proveedore en la nube, que estos cumplan con las normativas locales.

Para la puesta en producción, se requiere documentación legal obligatoria que incluye la elaboración de Términos y Condiciones y Políticas de Privacidad siguiendo las directrices del sitio web de la URCDP [38], la inclusión de derechos de información especificando qué implican y quiénes intervienen en el procesamiento, y la implementación de un *pop-up* informativo con aspectos básicos que el usuario debe conocer.

El proceso de registro exige registrar obligatoriamente la base de datos ante URCDP, documentar todos los tratamientos de datos personales, y establecer procedimientos claros para el ejercicio de derechos de los titulares de datos.

Las consideraciones específicas para SafeCall, en caso de llegar a desarrollarse como un producto final, establecen que la naturaleza de los datos de llamadas requiere especial atención al marco legal, el procesamiento automatizado para detección de fraudes debe balancear efectividad con protección de privacidad, y la colaboración con operadoras telefónicas debe contemplar aspectos regulatorios específicos que fueron detallados durante la entrevista.

Conclusiones y Resultados del proceso de Empatía

El proceso de entrevistas validó la existencia de dos perfiles claramente diferenciados, con necesidades específicas pero complementarias. El Perfil 1 de Adultos mayores vulnerables representa el segmento con mayor dolor expresado y necesidad urgente de protección, mientras que el Perfil 2 de Adultos jóvenes tecnológicamente competentes,

constituye un grupo que puede beneficiarse de herramientas más sofisticadas de información y control.

En la siguiente Tabla se muestra una comparación entre ambos perfiles identificados.

	Perfil 1	**Perfil 2**
Descripción	Usuario Vulnerable	Usuario Tecnológicamente Competente
Demografía	Adultos mayores con conocimientos tecnológicos variados (principalmente básicos)	Adultos jóvenes/mediana edad (20-50 años) con conocimientos tecnológicos avanzados
Experiencia con estafas	Víctimas directas o con experiencias cercanas de estafas telefónicas	Sin gran preocupación por ser estafados debido a su confianza en detectar estas situaciones
Dolor principal	Miedo de sufrir una estafa, perder dinero y sentirse avergonzados; pérdida de confianza personal y aislamiento autoimpuesto	Interrupciones laborales constantes por llamadas de números desconocidos; falsa confianza que subestima el riesgo
Necesidades clave	• Protección automática sin acciones manuales • Red de apoyo familiar ("mi familia puede saber que estoy en peligro") • Simplicidad extrema ("muy fácil de usar")	• Control sobre el funcionamiento de la solución • Información preventiva sobre modalidades actuales de estafa • Base de datos colaborativa de números sospechosos • Verificación bidireccional

		(2FA)
Comportamiento	Evitan contestar llamadas desconocidas; necesitan depender de familiares para validar situaciones	Confían en su habilidad para detectar estafas, pero no usan herramientas preventivas disponibles
***Insights* comunes**	• Desconfianza del uso de información personal por aplicaciones • No quieren sumar gasto extra por un servicio de uso poco frecuente • Necesitan que alguien de confianza instale la solución	• Desconfianza del uso de información personal por aplicaciones • No quieren sumar gasto extra por un servicio de uso poco frecuente
***Insights* diferenciadores**	• Requieren que la familia esté presente para ayudarlos • Prefieren soluciones completamente automáticas para no sentirse culpables ni hacer pasar un mal rato a su familia	• Valoran especialmente la transparencia en el manejo de datos personales • Prefieren tener control sobre las funcionalidades de la solución • Requieren confianza sobre almacenamiento y uso de su información

Tabla 1 - Comparación de Perfiles identificados

Con base en estos resultados, se definió enfocar el desarrollo inicial del Prototipo Funcional en el Perfil 1 (Adultos mayores vulnerables). Esta decisión se fundamenta en tres factores clave: primero, representan el grupo más desprotegido ante las estafas telefónicas; segundo, expresaron el mayor nivel de dolor y necesidad urgente de protección; y tercero, una solución tecnológica apropiada tendría el mayor impacto

potencial en este segmento. Las funcionalidades avanzadas orientadas al Perfil 2 se contemplaron para futuras versiones de la solución.

Los *insights* obtenidos proporcionaron la base directa para la definición de requerimientos funcionales, criterios de usabilidad y consideraciones de privacidad que guiaron el desarrollo de SafeCall, validando tanto la prevalencia del problema como la existencia de una brecha significativa en la adopción de medidas preventivas, lo que justifica el desarrollo de soluciones accesibles, gratuitas y efectivas.

3.4 Ideación de Soluciones

A partir de los perfiles y la información obtenidos durante la fase de *Product Discovery*, el equipo procedió a la generación de alternativas de solución mediante un proceso de ideación. Esta etapa se caracterizó por la exploración de múltiples enfoques tecnológicos que pudieran abordar eficazmente el problema de las estafas telefónicas desde diferentes perspectivas. Como parte del proceso, se realizaron múltiples sesiones cortas de *brainstorming* donde, en base a una pregunta especifica, cada miembro del grupo proponía ideas sin cuestionarlas. Una vez finalizada esta fase, se agruparon las ideas en base a los posibles tipos de solución y también contemplando potenciales características de las mismas a incorporar (para ver el detalle del proceso de ideación dirigirse al ANEXO 1).

3.4.1 Generación de Conceptos

El proceso de ideación resultó en tres conceptos principales, cada uno con un enfoque tecnológico diferente para abordar la detección y prevención de estafas telefónicas.

Concepto 1: Aplicación Móvil con Inteligencia Artificial

El Concepto 1 propone una aplicación móvil que, mediante la escucha de la llamada permite identificar la intención del fraude con uso de *machine learning* e inteligencia artificial, notificando al usuario y personas interesadas.

Este concepto aprovecha las capacidades nativas de los dispositivos móviles para capturar, analizar y procesar las llamadas telefónicas en tiempo real utilizando técnicas de procesamiento de lenguaje natural para detectar patrones característicos de fraude.

Concepto 2: Sistema de Lista Blanca con Autenticación de Dos Factores

El Concepto 2 plantea una aplicación de identificación y seguridad en llamadas que solo permite el ingreso de llamados desde destinos seguros mediante una lista blanca o autenticación de dos factores.

Esta alternativa se enfoca en un enfoque preventivo, creando un ecosistema de confianza donde únicamente las llamadas de origen verificado pueden establecer comunicación con el usuario, implementando mecanismos de autenticación avanzados.

Concepto 3: Dispositivo Externo de Captura de Audio

El Concepto 3 considera un dispositivo de conexión externa al teléfono celular, mediante *bluetooth* o *plug-and-play*, que permite la grabación del audio de la llamada en tiempo real y mediante *machine learning* identifica la intención de fraude.

Este concepto plantea una solución híbrida *hardware-software* que supera las limitaciones de acceso al audio impuestas por los sistemas operativos móviles modernos.

3.4.2 Selección de Conceptos

Para evaluar la factibilidad técnica de cada concepto, se desarrollaron prototipos técnicos mediante codificación de aplicaciones demo y adquisición y prueba de dispositivos de detección de audio.

Concepto 1: Aplicación Móvil con IA

Durante la validación técnica del Concepto 1, se identificaron limitaciones importantes relacionadas con las políticas de privacidad de Android. Las pruebas realizadas demostraron que Android 11 y versiones superiores no permiten acceso al audio de las llamadas telefónicas estándar debido a restricciones de privacidad introducidas en estas versiones.[39]

Las investigaciones técnicas incluyeron pruebas con emulador Android 11 y la exploración de tecnologías como “Sinch” [40] para crear un sistema de llamadas alternativo, desarrollo de interfaces para acceso a grabaciones existentes y

configuración de servicios de transcripción en tiempo real. Sinch es una plataforma de comunicaciones en la nube que proporciona APIs para integrar funcionalidades de voz, vídeo y mensajería en aplicaciones, permitiendo crear sistemas de llamadas independientes de las restricciones del sistema operativo [41].

Se determinó que la implementación sería muy compleja y poco segura, requiriendo enfoques alternativos como compatibilidad con dispositivos antiguos u obsoletos o funcionalidades nativas limitadas.

La implementación puramente de software ofrece ventajas en términos de reducción de costos de *hardware*, integración nativa con funcionalidades del dispositivo, capacidad de análisis en tiempo real y escalabilidad mediante tiendas de aplicaciones. Sin embargo, las restricciones técnicas identificadas representan obstáculos significativos para la viabilidad del concepto.

Concepto 2: Sistema de Lista Blanca

El análisis revela que este enfoque presenta fortalezas significativas en términos de seguridad y control. Los usuarios mantienen autoridad completa sobre quién puede contactarlos, mientras el sistema se optimiza automáticamente según los patrones de uso del teléfono. Todo el procesamiento ocurre localmente sin dependencia de bases de datos externas, garantizando privacidad por diseño.

Las limitaciones incluyen el potencial bloqueo de llamadas legítimas de números no registrados como servicios de entrega o emergencias, el requerimiento de interacción inicial del usuario para validar nuevos números y menor efectividad en entornos con alta frecuencia de llamadas de números no registrados.

La factibilidad técnica se respalda en la capacidad de los *smartphones* modernos para ejecutar procesamiento de llamadas y gestión de contactos. Aplicaciones similares como “Truecaller”, una aplicación que identifica números desconocidos y bloquea llamadas y mensajes de spam mediante una lista comunitaria [42], demuestran la viabilidad técnica de sistemas de gestión de llamadas, validando la implementación de funcionalidades de lista blanca/ lista negra y autenticación de dos factores.

Concepto 3: Dispositivo Externo

Como parte de la validación, se realizó una investigación de dos dispositivos externos de grabación distintos. Se evaluó el RecorderGear PR200, un dispositivo Bluetooth para grabación de llamadas con limitaciones en acceso en tiempo real [43], y se desarrolló una prueba de concepto utilizando Bluetooth Externo Urgreen 5A para transmitir audio a dispositivo USB-C.[44]

El enfoque de hardware dedicado ofrece ventajas como la superación de limitaciones de software, independencia de políticas de sistema operativo y procesamiento potencialmente más robusto. Sin embargo, presenta desafíos significativos incluyendo complejidad de desarrollo y manufactura, costos de producción mayores, dependencia de accesorios externos y experiencia de usuario más compleja.

3.4.3 Matriz de Evaluación de Pugh

Para realizar una evaluación objetiva de los tres conceptos se implementó la Matriz de Pugh. Esta técnica permite comparar sistemáticamente diferentes alternativas contra un conjunto de criterios ponderados, facilitando la selección de la opción más viable mediante una técnica de puntuación (a mayor puntuación, más se adecua el concepto a las necesidades de los usuarios y del negocio) [29]

Los criterios de evaluación seleccionados incluyeron detección del fraude, velocidad de detección en tiempo real, facilidad de uso, seguridad, factibilidad, costo de desarrollo, capacidad de denuncia de números y efectividad de notificación de fraude.

Para ver los detalles de la comparativa mediante la Matriz, dirigirse a ANEXO 1.

Los resultados de la evaluación mostraron que el Concepto 1 destacaba en velocidad de detección y facilidad de uso, pero presentaba desafíos en factibilidad y costo de desarrollo. El Concepto 2, con igual puntuación que el anterior, sobresalía en seguridad y costo de desarrollo, aunque presentaba limitaciones en velocidad de detección en tiempo real. El Concepto 3 obtuvo menor cantidad de puntos, manteniendo capacidades de detección, pero enfrentando dificultades en facilidad de uso y costo de desarrollo.

3.4.4 Soluciones Comerciales Existentes

El análisis de soluciones comerciales existentes revela que para el Concepto 1 existen aplicaciones como “Truecaller” para identificación de llamadas *spam* mediante base de datos comunitaria [42], “Hiya” para protección contra llamadas no deseadas con tecnología de IA [45] y “RoboKiller” para filtrado automático utilizando *machine learning* [46]. Estas aplicaciones han demostrado la viabilidad comercial de sistemas de detección automatizada, con “Truecaller” reportando más de 350 millones de usuarios activos mensuales y capacidad de bloquear más de 40 mil millones de llamadas spam anualmente [42].

Para el Concepto 2 se identificaron aplicaciones nativas con funcionalidades de lista blanca y con denuncias comunitarias como “Call Blocker” [47] y sistemas de bloqueo basado en listas personalizadas como “Mr. Number” [48]. Estas soluciones validan la efectividad del enfoque preventivo mediante control de acceso.

El Concepto 3 cuenta con dispositivos especializados para grabación como “Call Recorder” por Bluetooth (ya mencionado en el apartado anterior), entre otros, y soluciones empresariales de interceptación profesional, aunque estas últimas resultan costosas para usuarios individuales. La investigación reveló limitaciones significativas en dispositivos comerciales disponibles, particularmente en capacidades de procesamiento en tiempo real y accesibilidad de costos.

3.4.5 Concepto Final

Tras el análisis comparativo mediante la Matriz de Pugh y considerando los resultados técnicos de validación, se desarrolló un Concepto Final híbrido que combinó los elementos más viables de los conceptos evaluados. El concepto final se define como: “una aplicación que permite, tras cortar la llamada, identificar la intención del fraude, notificar al usuario y personas interesadas, y facilitar la denuncia del número sospechoso a una lista negra.”

El concepto incorpora análisis post-llamada eliminando el requisito de procesamiento en tiempo real, lo que soluciona las limitaciones técnicas identificadas en el Concepto 1. Implementa procesamiento offline aprovechando grabaciones existentes o

transcripciones para análisis posterior, superando restricciones de acceso al audio en tiempo real.

El sistema incluye notificaciones automáticas a usuarios y contactos de confianza tras detección de fraude y una base de datos colaborativa mediante sistema de denuncia que alimenta una lista negra comunitaria. Este enfoque híbrido combina detección por inteligencia artificial con elementos preventivos de lista blanca y negra.

Las ventajas del concepto final incluyen viabilidad técnica al superar limitaciones de acceso al audio en tiempo real, factibilidad de desarrollo mediante reducción de complejidad técnica, escalabilidad con capacidad de mejora continua del modelo, impacto social contribuyendo a protección colectiva y compatibilidad con dispositivos Android modernos sin requerir *hardware* adicional.

Este concepto final representa la síntesis de las observaciones y resultados de los aprendizajes obtenidos durante la fase de evaluación, priorizando la viabilidad técnica sin comprometer la efectividad en la detección de fraudes telefónicos. La decisión de proceder con análisis post-llamada, aunque reduce la inmediatez de la respuesta, garantiza la factibilidad del proyecto dentro de las limitaciones técnicas y de recursos identificadas.

3.5 Pruebas del Concepto

Una vez definido el concepto final híbrido en la sección anterior, se procedió a su validación mediante el desarrollo de un prototipo de baja fidelidad y entrevistas con usuarios potenciales, como fue mencionado en la sección 3.1. Esta fase fue importante para refinar las funcionalidades propuestas y establecer las bases para la definición de requerimientos funcionales y no funcionales del sistema.

El prototipo, disponible en el ANEXO 1, se diseñó considerando los *insights* obtenidos durante las entrevistas del *discovery*, particularmente las necesidades del Perfil 1 de usuarios vulnerables que requerían simplicidad extrema y funcionalidad automática. Se priorizó una interfaz minimalista con elementos visuales claros y navegación intuitiva.

Funcionalidades Validadas en el Prototipo

El prototipo inicial incluyó las siguientes funcionalidades centrales identificadas durante la fase de ideación:

Sistema de Alertas Automáticas: La funcionalidad principal se centró en mostrar notificaciones emergentes que alertaran sobre posibles intentos de estafa detectados. El prototipo simulaba la aparición de una alerta con el mensaje "Se ha detectado un posible intento de estafa en su última llamada" con un botón "VER DETALLES" para acceder a información adicional.

Configuración de Alertas: Se implementó una pantalla de configuración que permitía al usuario especificar las acciones deseadas tras la detección de una llamada sospechosa. Esta incluía dos opciones principales mediante *switches* activables: "Notificar a Usuarios designados" para enviar alertas a contactos de confianza, y "Guardar número como sospechoso" para alimentar una base de datos de números problemáticos.

Gestión de Contactos de Emergencia: El prototipo incorporó la funcionalidad "Agregar contactos para Notificaciones" que permitía al usuario designar familiares o personas de confianza que recibirían alertas automáticas en caso de detectarse un intento de estafa.

Interfaz de Usuario Simplificada: El diseño priorizó elementos visuales grandes, colores contrastantes y texto descriptivo claro para facilitar el uso por parte de adultos mayores con posibles limitaciones visuales o tecnológicas.

Funcionalidades Descartadas Durante la Validación

El proceso de validación con usuarios reveló que ciertas funcionalidades inicialmente consideradas no agregaban valor o generaban fricción innecesaria:

Sistema de Inicio de sesión/Registro: Una de las decisiones más significativas fue la eliminación del sistema de autenticación tradicional. Las entrevistas con usuarios del Perfil 1 revelaron que cualquier barrera inicial de acceso reduciría significativamente la adopción. Varios de los adultos mayores entrevistados, expresaron que en general evitan usar aplicaciones ya que se olvidan de las contraseñas y contar con otras aplicaciones para guardar y memorizar las mismas les implica aprender a usar aún más aplicaciones.

Esta retroalimentación condujo a la implementación de un acceso directo sin credenciales, priorizando la facilidad de uso sobre funcionalidades avanzadas de personalización.

Funcionalidades de Bloqueo Automático: Durante las entrevistas, varios de los entrevistados, como Paula y Amanda expresaron claramente que no deseaban que la aplicación "tome decisiones" por ellos. Amanda específicamente indicó que prefería "que simplemente avise, que no decida por mí y cuelgue la llamada". Esta retroalimentación llevó a eliminar cualquier funcionalidad que cortara o bloqueara llamadas automáticamente, manteniendo al usuario en control total de sus comunicaciones.

Técnicas de Validación Aplicadas

Se emplearon técnicas exploratorias para validar conceptos generales y recopilar ideas de funcionalidades. Se presentó a los potenciales usuarios un prototipo interactivo y se les solicitó realizar tareas concretas para comprender sus expectativas y obtener sugerencias de mejora. Asimismo, se utilizó la técnica de Mago de Oz, simulando funcionalidades del sistema de manera controlada para observar cómo los usuarios interactuarían con el prototipo en situaciones realistas. Complementariamente, se realizaron entrevistas contextuales para entender cómo los usuarios imaginarían utilizar cada funcionalidad en su vida cotidiana y validar la relevancia práctica de las propuestas, como también descartar aquellas que no les aportaran valor.

Resumen de resultados de Pruebas con usuarios

Las entrevistas de validación produjeron *insights* que refinaron el concepto final:

En primer lugar, los usuarios validaron la necesidad de la detección automática de estafas telefónicas, confirmando que el 100 % de los entrevistados había experimentado o conocía casos cercanos de intentos de estafa. La propuesta de valor del concepto final se alineó efectivamente con las necesidades expresadas.

También se confirmó la preferencia de los usuarios por sistemas que alertarán sin tomar decisiones automáticas. Ningún entrevistado deseaba que la aplicación cortara llamadas o bloqueara números sin su intervención explícita, validando la decisión de implementar un sistema de alertas no intrusivo.

Todos los entrevistados expresaron la necesidad de transparencia completa sobre el manejo de datos personales y grabaciones de llamadas. Esta retroalimentación llevó a priorizar la documentación clara de políticas de privacidad y el desarrollo de funcionalidades de control de datos por parte del usuario.

Por último, las entrevistas confirmaron que la simplicidad de uso resultaba más importante que la sofisticación técnica. Los usuarios preferían funcionalidades automáticas confiables sobre opciones de configuración complejas.

La validación del concepto funcionó como un puntapié inicial para el desarrollo del prototipo funcional, proporcionando información valiosa sobre las necesidades y expectativas de los usuarios. A partir de ella, fue posible identificar áreas de mejora en la interfaz, la forma de comunicar alertas, así como confirmar algunas decisiones de diseño iniciales.

4. Definición del Prototipo Funcional

El proceso de definición de requerimientos se caracterizó por una superposición entre *discovery* y *delivery*, donde la validación de funcionalidades se fue dando a través de tres *releases* sucesivos. Esto permitió refinar los requerimientos basándose tanto en los *insights* del *discovery* como en resultados obtenidos durante el desarrollo y pruebas con usuarios. (por imagen conceptual de la superposición dirigirse a ANEXO 2)

La investigación primaria proporcionó datos cuantitativos y cualitativos que se transformaron en requerimientos funcionales iniciales, mientras que las pruebas con usuarios tras cada *release* permitieron identificar requerimientos nuevos que enriquecieron la propuesta de valor de la aplicación.

4.1 Requerimientos Funcionales

Como criterio de priorización, se tomó el valor o beneficio que genera para el usuario dicha funcionalidad, de forma de poder aportar el mayor valor posible en el menor tiempo. Para ello, se designaron las siguientes categorías:

- Crítica: Indispensable para el funcionamiento mínimo del producto.
- Alta: Funcionalidad muy importante, pero no indispensable.
- Media: Funcionalidad deseable que mejora la experiencia, pero no esencial.
- Baja: Funcionalidad de menor prioridad o que se puede posponer.

En la siguiente tabla se detallan los diferentes requerimientos con su descripción, justificación, prioridad y los hallazgos de los cuales derivaron que pueden ser consultados en la Tabla de Resultados en el ANEXO 1

ID	Descripción	Justificación	Prioridad	Hallazgos
RF01	El sistema debe detectar automáticamente llamadas fraudulentas mediante análisis de audio	Responde a la alta incidencia de llamadas sospechosas y la demanda de identificación automática	CRÍTICA	H01, H02
RF02	El sistema debe generar alertas automáticas al usuario cuando detecte una llamada sospechosa de estafa	Implementa la funcionalidad de identificación valorada por el 70 % de usuarios	CRITICA	H02
RF03	El sistema debe enviar notificaciones por SMS a contactos de seguridad, incluyendo el número sospechoso y señal de alerta	Atiende la necesidad de alertar a familiares, especialmente para adultos mayores	ALTA	H03, H06
RF04	El sistema debe permitir al usuario agregar/eliminar contactos de seguridad designados	Facilita la gestión personalizada de contactos de confianza solicitada en validaciones	ALTA	H06
RF05	El sistema debe mostrar un historial de llamadas sospechosas	Proporciona seguimiento de eventos y alertas, ayudando al usuario a recordarlas	MEDIA	H05, H09

RF06	El sistema debe permitir al usuario ver detalles de cada llamada sospechosa	Complementa el historial con información detallada para un posible uso posterior	MEDIA	H05, H09
RF07	El sistema debe permitir al usuario bloquear/desbloquear un número de teléfono para evitar recibir llamadas telefónicas del mismo	Implementa la capacidad de bloqueo solicitada por usuarios en entrevistas	MEDIA	H07
RF08	El sistema debe permitir al usuario denunciar manualmente un número como sospechoso para luego ser considerado por el sistema como tal en futuras llamadas	Integra recomendaciones de expertos sobre lista negra colaborativa y facilita denuncias	MEDIA	H04, H08
RF09	El sistema debe permitir al usuario poder reproducir y compartir grabaciones de las llamadas sospechosas	Necesidad identificada durante pruebas con usuarios entre Releases	MEDIA	Emergente - Release 1
RF10	El sistema debe permitir la Configuración de Preferencias para cada usuario	Requerimiento de personalización surgido de feedback de usuarios durante pruebas	MEDIA	Emergente - Release 1
RF11	El sistema debe permitir la Variación de Tipografía/Tamaño de	Necesidad de accesibilidad identificada durante pruebas con adultos	MEDIA	Emergente - Release 1

	letras	mayores (Perfil 1)		
RF12	El sistema debe permitir el Bloqueo automático de llamadas de números desconocidos (“Modo Guardián”)	Funcionalidad avanzada identificada durante pruebas con usuarios entre Releases	MEDIA	Emergente - Release 2

Tabla 2 - Requerimientos Funcionales

4.2 Requerimientos No Funcionales

Los requerimientos no funcionales de SafeCall se derivan de la información obtenida durante la etapa de *discovery* y se alinean con los atributos de calidad priorizados para el prototipo: performance, usabilidad y deployabilidad. Estos atributos fueron seleccionados por su impacto directo en la percepción de valor, en la experiencia de los usuarios y en la viabilidad técnica de su implementación. Para ver los requerimientos en formato tabla dirigirse a ANEXO 2

Performance

La performance constituye el atributo más importante, ya que el valor principal de SafeCall reside en su capacidad para detectar y alertar al usuario con rapidez ante una llamada potencialmente fraudulenta.

Se estableció como requerimiento que el análisis post-llamado se complete en un tiempo igual o inferior a 30 segundos, umbral que permite actuar de manera preventiva frente a intentos de estafa en los que el estafador suele continuar la interacción después de finalizada la llamada.

Además, dado que la aplicación se apoya en modelos de inteligencia artificial, se definió como meta una precisión mínima del 80 %, equilibrio que permite demostrar valor y efectividad ante los usuarios y posibles interesados, sin comprometer la viabilidad técnica. [49]. Este nivel de desempeño asegura resultados confiables,

manteniendo tiempos de respuesta adecuados y la posibilidad de mejorar el modelo en futuras iteraciones.

Usabilidad

En segundo lugar, la usabilidad se estableció como un atributo fundamental, considerando que el público objetivo de SafeCall está compuesto principalmente por adultos mayores.

Este grupo presenta limitaciones tecnológicas y potencialmente físicas que condicionan la experiencia de uso, lo que hace imprescindible una interfaz accesible y fácil de comprender.

Asimismo, las notificaciones deben emplear colores contrastantes y tipografías legibles, de modo que sean claramente visibles incluso en situaciones de estrés o vulnerabilidad. Estos criterios buscan maximizar la accesibilidad y reducir la carga cognitiva, garantizando que la tecnología actúe como soporte y no como barrera para el usuario final.

Deployabilidad

Por último, la deployabilidad tiene valor en el desarrollo del prototipo de SafeCall, dado que permite mantener el prototipo operativo de manera confiable durante las pruebas con usuarios. Para facilitar esto, el sistema fue desarrollado utilizando contenedores Docker [50], lo que asegura que el prototipo pueda ejecutarse de forma consistente en distintos entornos (desarrollo, *testing* y producción). Esta decisión tecnológica habilita la integración y la entrega continuas (CI/CD, por sus siglas en inglés) [51], reduciendo riesgos al actualizar el prototipo y simplificando la incorporación de cambios o mejoras sin afectar la experiencia de los usuarios que lo están evaluando.

Requerimientos No Funcionales a contemplar para una solución en Producción.

Además de los atributos de calidad priorizados, existen otros identificados que resultan relevantes en caso de querer elevar el prototipo a una solución de producción. Entre ellos, la seguridad y la privacidad adquieren especial importancia debido al manejo de datos personales y a la naturaleza sensible de las llamadas analizadas por el sistema.

Asimismo, la disponibilidad sería un atributo relevante en una versión definitiva, ya que la efectividad de SafeCall depende de que el servicio se encuentre siempre operativo para brindar protección al usuario en tiempo real.

Por último, la confiabilidad también se vuelve un aspecto central, puesto que los usuarios deben poder confiar en la exactitud del análisis y en el resguardo de su información.

Si bien estos atributos no fueron abordados en esta etapa, su consideración será fundamental en la evolución del sistema hacia una solución plenamente productiva.

5. Conceptos de Inteligencia Artificial

5.1 Definición y Alcance de la Inteligencia Artificial

La Inteligencia Artificial (IA) puede definirse de manera general como el conjunto de técnicas y métodos que buscan dotar a los sistemas informáticos de la capacidad de ejecutar tareas que, en condiciones normales, requieren de la inteligencia humana [52]. Estas tareas incluyen el reconocimiento del lenguaje, la toma de decisiones, la resolución de problemas, el aprendizaje a partir de datos y la adaptación a nuevos contextos.

En términos prácticos, la IA se fundamenta en algoritmos que procesan grandes volúmenes de información, detectan patrones ocultos y generan predicciones o clasificaciones que resultan útiles para resolver problemas específicos. Este enfoque ha transformado numerosos campos, desde la medicina hasta la ingeniería, pasando por la educación, las finanzas y las telecomunicaciones.

5.2 Procesamiento de Lenguaje Natural

Uno de los subcampos más relevantes para este proyecto es el Procesamiento de Lenguaje Natural, disciplina de la IA que estudia cómo los sistemas pueden entender, interpretar y generar lenguaje humano. El PLN combina técnicas lingüísticas y matemáticas con modelos de aprendizaje automático, permitiendo analizar texto escrito o transcripciones de audio y extraer de ellas información significativa.

Retos y Complejidad del Lenguaje Humano

El desafío central del PLN radica en la ambigüedad, la riqueza semántica y la variabilidad del lenguaje humano [53]. Consideremos los siguientes aspectos:

- Ambigüedad léxica: Una palabra puede tener múltiples significados según el contexto
- Variabilidad sintáctica: Una misma idea puede expresarse de múltiples formas
- Implícitos culturales: Referencias que requieren conocimiento contextual específico

- Ironía y sarcasmo: Expresiones donde el significado literal difiere del intencional

Estos desafíos requieren modelos complejos para alcanzar un entendimiento robusto del lenguaje humano.

Un Cambio de Paradigma en el PLN con Transformers

Un avance relevante en PLN fue la introducción de la arquitectura Transformer en 2017 [54]. Los Transformers utilizan un mecanismo de atención que permite al modelo enfocarse en diferentes partes del texto simultáneamente, capturando relaciones complejas entre palabras independientemente de su distancia en la secuencia.

La arquitectura Transformer supera las limitaciones de las redes neuronales recurrentes (RNN, por su sigla en inglés) tradicionales, que procesaban el texto secuencialmente y tenían dificultades para mantener información a largo plazo. Los Transformers procesan todo el texto en paralelo, mejorando significativamente la eficiencia y capacidad de modelado

5.3 Modelos Pre-entrenados y Fine-Tuning

El propósito de esta sección es exponer las bases del aprendizaje transferido y su relevancia para SafeCall. Incluir estos conceptos permite contextualizar la elección de modelos pre-entrenados y explicar de qué manera esta estrategia mejora la eficiencia, la precisión y la adaptabilidad del sistema.

Por qué Usar Modelos Pre-entrenados

Los modelos pre-entrenados constituyen un avance en el procesamiento de lenguaje natural moderno. Ejemplos como BERT, DistilBERT, RoBERTa o GPT han sido entrenados previamente con grandes volúmenes de texto en múltiples idiomas y dominios, desarrollando una comprensión general del lenguaje. A partir de ellos, el *transfer learning* [55] permite adaptar estas arquitecturas a problemas específicos mediante un proceso denominado *fine-tuning*, en el cual sus parámetros son ajustados con un conjunto de datos más pequeño pero especializado. Esta estrategia presenta

varias ventajas: en primer lugar, mejora la eficiencia computacional al reducir de manera significativa el tiempo y los recursos necesarios para entrenar modelos especializados; en segundo lugar, incrementa el rendimiento gracias a que los modelos pre-entrenados ya capturan conocimiento lingüístico general que puede transferirse a la tarea objetivo; y, finalmente, disminuye los requerimientos de datos, posibilitando obtener resultados competitivos incluso con conjuntos de entrenamiento más reducidos.

Aplicación del Fine-Tuning en SafeCall

En el contexto del proyecto SafeCall, el ajuste fino se aplicó para especializar los modelos en la detección de patrones característicos de mensajes fraudulentos en transcripciones telefónicas. Si bien los modelos pre-entrenados ya captaron conocimiento general del lenguaje, no están optimizados para identificar estafas específicas. Al realizar un ajuste fino, se logra que los modelos adapten su comprensión general a las particularidades del dominio, aumentando significativamente la precisión de la clasificación y evitando la necesidad de entrenar un modelo desde cero, lo que reduce tanto el tiempo como los recursos computacionales requeridos.

BERT y las Evoluciones que Impulsaron el PLN

BERT (*Bidirectional Encoder Representations from Transformers*) introdujo el concepto de entrenamiento bidireccional, donde el modelo considera tanto el contexto anterior como posterior a cada palabra [56]. Esto contrasta con modelos previos que procesaban el texto unidireccionalmente.

DistilBERT es una versión optimizada de BERT que mantiene el 95 % de su rendimiento utilizando solo el 60 % de sus parámetros [57] Esta eficiencia la convierte en una opción atractiva para aplicaciones con restricciones computacionales.

RoBERTa (*Robustly Optimized* BERT *Pre Training Approach*) mejora BERT mediante optimizaciones en el proceso de entrenamiento, eliminando la tarea de predicción de la siguiente oración y utilizando secuencias más largas [58]

XLM-RoBERTa extiende RoBERTa al contexto multilingüe, entrenando con datos en 100 idiomas diferentes, lo que le confiere capacidades superiores para textos en español y otros idiomas no ingleses [59]

5.4 Detección de Estafas Telefónicas en SafeCall

Definición del Problema y Patrones de Estafa

La aplicación de inteligencia artificial en este trabajo se centra en la detección discriminativa de estafas telefónicas a partir de las transcripciones textuales de las conversaciones. El objetivo es entrenar un sistema que, al recibir un mensaje, sea capaz de clasificarlo como sospechoso (*scam*) o legítimo (neutral). Este enfoque se considera factible porque el lenguaje utilizado en las estafas suele presentar patrones recurrentes que pueden ser identificados por un modelo entrenado.

Los patrones típicos incluyen:

- Urgencia temporal: Frases como "debe actuar inmediatamente" o "tiempo limitado"
- Solicitudes de información sensible: Peticiones de datos bancarios, contraseñas o códigos
- Autoridad falsa: Suplantación de entidades bancarias, gubernamentales o de servicios
- Ofertas irreales: Promesas de ganancias excesivas o beneficios improbables
- Presión emocional: Apelación a miedos o emociones para reducir el pensamiento crítico

Características y Preparación del Dataset

Para el entrenamiento de los modelos se utilizó un *dataset* especializado en conversaciones de estafa, compuesto por 650 registros distribuidos en 65 diálogos, cada uno con hasta 17 pasos o intervenciones individuales dentro del intercambio. Cada paso corresponde a una línea de diálogo emitida por uno de los participantes, ya sea el posible estafador o la víctima. Este conjunto incluye, para cada intervención, información como el identificador de conversación, el número de paso, el texto, el contexto, la etiqueta asignada, las características estilísticas y eventuales anotaciones que marcan estrategias o señales relevantes.

La siguiente tabla presenta dos ejemplos extraídos del corpus, que reflejan la diferencia entre un mensaje neutral y uno sospechoso (scam):

Mensaje	**Ejemplo**	**Etiqueta**	**Estrategias**
Intercambio estándar	"Buenos días, le habla el asistente personal de [Su Nombre]. ¿En qué puedo ayudarle hoy?"	Neutral	Saludo formal y apertura de conversación típica.
Posible estafa	"Sí, hola. Le habla el Agente Thompson de la Agencia Federal de Impuestos. Hemos identificado una discrepancia significativa en las declaraciones de su empresa que requiere atención inmediata para evitar acciones legales y sanciones."	Scam	Uso de autoridad, urgencia y títulos oficiales como estrategias de persuasión.

Tabla 3 - Ejemplo de valores en dataset

El *dataset* constituye el insumo principal, ya que la calidad y diversidad de los datos determinan directamente el desempeño de los modelos. Para garantizar su utilidad fue necesario normalizar, es decir, unificar formatos, corregir inconsistencias y estandarizar la información, de modo que los algoritmos pudieran procesar los registros de manera consistente y confiable.

Cabe señalar que el *corpus* utilizado fue obtenido de la plataforma Kaggle[60], una comunidad abierta para el intercambio de *datasets* y recursos de aprendizaje automático. El conjunto empleado se encuentra disponible públicamente bajo una licencia de uso libre para investigación, lo que permite su utilización con fines académicos y experimentales. En el ANEXO 3.1 se documenta su procedencia, condiciones de licencia y estructura original.

5.5 Repositorio de Modelos Hugging Face

Hugging Face se ha establecido como la plataforma líder para el desarrollo y distribución de modelos de aprendizaje automático, especialmente en el ámbito del procesamiento de lenguaje natural [61]. Esta plataforma ofrece un ecosistema completo que incluye:

- *Hub* de modelos: Repositorio con más de 200,000 modelos pre-entrenados disponibles públicamente
- Biblioteca Transformers: *Framework* unificado para cargar, utilizar y realizar el ajuste fino de modelos pre-entrenados.
- *Datasets*: Colección de *datasets* estándar para entrenamiento y evaluación.
- Espacios (*Spaces*): Plataforma para crear y compartir demos interactivas.

Ventajas de Hugging Face para SafeCall

El uso de Hugging Face en el proyecto SafeCall proporcionó múltiples beneficios que se exponen a continuación.

Acceso simplificado a modelos SOTA (*State of the Art* - estado del arte).

La plataforma permite acceder a modelos avanzados como BERT, DistilBERT, RoBERTa y XLM-RoBERTa mediante una API unificada. En este contexto, el estado del arte significa que estos modelos han alcanzado resultados sobresalientes en evaluaciones de desempeño ampliamente utilizadas. Al tratarse de modelos pre-entrenados, se evita la necesidad de implementar y entrenar cada arquitectura desde cero, lo que reduce significativamente tanto el tiempo como el costo de desarrollo, acortando la brecha hacia la etapa de ajuste fino en la tarea específica [62].

Compatibilidad y estandarización

Los modelos se distribuyen bajo un formato unificado que integra configuración, pesos y tokenizador. Esta homogeneidad simplifica la experimentación y permite comparar de forma sistemática distintas arquitecturas, lo cual resulta de relevancia para una evaluación rigurosa dentro de este proyecto.

Eficiencia en el desarrollo

La biblioteca Transformers proporciona utilidades preconfiguradas para tokenización, entrenamiento y evaluación, acelerando significativamente el ciclo de desarrollo.

Reproducibilidad

Los modelos disponibles en Hugging Face incluyen no solo los pesos entrenados, sino también la configuración completa (hiperparámetros, arquitectura y tokenizador) y mecanismos de verificación como *checksums* que aseguran que los archivos descargados no hayan sido alterados. Esto permite que cualquier investigador o desarrollador pueda volver a ejecutar los mismos experimentos y obtener resultados consistentes, lo cual se utiliza para la validación científica y para comparar de manera justa el desempeño entre distintos modelos.

La abstracción proporcionada por Hugging Face permitió concentrarse en los aspectos específicos del problema de detección de estafas, como la optimización de hiperparámetros y el análisis de resultados, en lugar de implementar infraestructura básica.

5.6 Metodología de Ajuste Fino Utilizada

Etapas del Proceso

El ajuste fino es un proceso de transferencia de aprendizaje donde un modelo pre-entrenado se adapta a una tarea específica mediante el ajuste de sus parámetros con un dataset especializado De forma general, este proceso involucra la selección del modelo base, la preparación de los datos, la configuración de hiperparámetros, el entrenamiento supervisado y la evaluación en datos no vistos.

En el contexto del proyecto SafeCall, estos pasos se implementaron con particularidades propias: se eligieron arquitecturas pre-entrenadas de la biblioteca Transformers (como BERT y DistilBERT) según criterios de rendimiento y eficiencia, se aplicó tokenización al *dataset* de conversaciones de estafa previamente normalizado, y se ajustaron hiperparámetros para optimizar la detección de mensajes fraudulentos frente a mensajes legítimos[63].

- Selección del modelo base: Elección entre arquitecturas pre-entrenadas según criterios de rendimiento y eficiencia.
- Preparación de datos: Tokenización y formateo del *dataset* según los requerimientos del modelo.
- Configuración de hiperparámetros: Ajuste de variables que controlan el proceso de entrenamiento.
- Entrenamiento supervisado: Proceso iterativo donde el modelo ajusta sus pesos para minimizar errores de clasificación.
- Evaluación y validación: Medición del rendimiento en datos no vistos durante el entrenamiento.

Hiperparámetros en el Entrenamiento

Los hiperparámetros constituyen variables que determinan el comportamiento del entrenamiento de un modelo y requieren un ajuste cuidadoso para optimizar su rendimiento. Entre los más relevantes se encuentran las épocas de entrenamiento (*epochs*), que indican cuántas veces el modelo recorre por completo el conjunto de

datos; un número adecuado permite mejorar la precisión, aunque un exceso puede provocar sobreajuste y pérdida de capacidad de generalización. Otro hiperparámetro es el *batch size*, que define cuántos ejemplos se procesan de manera simultánea; tamaños pequeños suelen favorecer la generalización al introducir mayor variabilidad en el entrenamiento, mientras que tamaños grandes proporcionan mayor estabilidad, aunque a costa de más recursos. La tasa de aprendizaje o *learning rate* en inglés, es probablemente el hiperparámetros más relevante, ya que regula la magnitud de los ajustes que realiza el modelo en cada iteración: valores demasiado altos pueden generar inestabilidad, mientras que valores muy bajos ralentizan el proceso. Finalmente, el *max length* establece cuántos tokens puede procesar el modelo en una sola entrada, lo que implica un equilibrio entre el contexto disponible y el costo computacional; secuencias más cortas permiten mayor eficiencia, pero reducen la capacidad de capturar información en textos extensos.

Experimentos

Los experimentos los denotamos como "expXX", donde "XX" es un identificador propio para identificarlo con sus respectivas configuraciones. Luego se definió un experimento *baseline* (exp01) con valores estándar, y se crearon 19 variaciones explorando diferentes combinaciones.

La siguiente tabla muestra las configuraciones más representativas

Experimento	Objetivo	Configuración Distintiva
exp01	Establecer un punto de referencia (*baseline*) para comparar el desempeño de las configuraciones posteriores.	4 épocas, *batch size* 8, *learning rate* 2e-5, 128 tokens.
exp02	Evaluar cómo influye entrenar el modelo	8 épocas (el doble que el *baseline*).

	durante más tiempo en su desempeño.	
exp06	Analizar si un incremento en el *learning rate* acelera el aprendizaje sin comprometer la precisión.	*Learning rate* elevado (5e-5).
exp10	Explorar una configuración "optimista" que combina un mayor número de épocas y un *batch size* superior, buscando un mejor equilibrio entre precisión y eficiencia.	8 épocas, *batch size* 16, *learning rate* 3e-5, 256 tokens.
exp15	Probar una configuración intermedia que mantenga un balance entre velocidad de entrenamiento y capacidad de generalización.	6 épocas, *batch size* 12, *learning rate* 3e-5, 128 tokens.
exp20	Someter el modelo a un escenario extremo de entrenamiento prolongado y alta carga de datos para observar su comportamiento frente al sobreajuste.	16 épocas, *batch size* 32, *learning rate* 1e-5, 256 tokens.

Para mayor detalle, puede encontrarse la tabla extendida en el ANEXO 3.2

Rendimiento de DistilBERT con ajuste fino

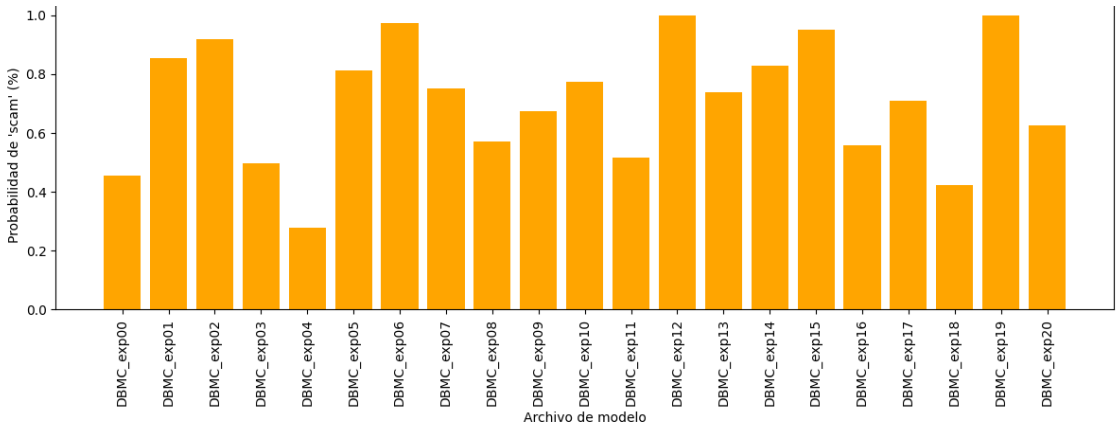


Ilustración 3-Análisis de probabilidad de estafa por modelo (DistilBERT fine-tuned)

La grafica anterior muestra como los distintos experimentos tienen distintas inferencias. Confirmando que no todas las configuraciones son óptimas para la mejora del modelo base.

Métricas de Evaluación

Para evaluar cada configuración se utilizaron cuatro métricas complementarias:

F1-Score: Balance entre precisión y *recall*. Especialmente útil cuando hay desbalance de clases (más llamadas normales que estafas). Un F1-Score de 95% significa que el modelo detecta bien las estafas sin generar demasiadas falsas alarmas.

Precision: De todas las llamadas que el modelo marca como estafa, ¿cuántas realmente lo son? Alta precisión = pocas falsas alarmas = usuario confía en las alertas.

Recall: De todas las estafas reales, ¿cuántas detecta el modelo? Alto *recall* = pocas estafas se escapan = mayor protección.

Loss (pérdida): Medida del error del modelo. Menor valor = mejor ajuste. Funciona como termómetro del aprendizaje.

Sobreajuste y Subajuste en el Entrenamiento

El sobreajuste y el subajuste (*overfitting* y *underfitting* por su traducción en ingles correspondientemente) son fenómenos en el entrenamiento de modelos de aprendizaje automático, ya que afectan directamente su capacidad de generalización. El sobreajuste ocurre cuando un modelo aprende demasiado los datos de entrenamiento, memorizando patrones específicos y ruido, lo que resulta en alto desempeño sobre esos datos, pero pobre rendimiento en datos no vistos. Por el contrario, el subajuste se produce cuando un modelo es demasiado simple para capturar la complejidad de los datos, mostrando bajo desempeño tanto en entrenamiento como en validación.

En SafeCall, estos fenómenos se detectaron comparando la pérdida de entrenamiento (*train_loss,* por su nombre en inglés) con la pérdida de validación (*eval_loss*, por su nombre en inglés). Una diferencia significativa indica que el modelo podría estar memorizando los ejemplos en lugar de aprender patrones generalizables. Durante los experimentos con DistilBERT, se observaron distintos comportamientos: algunos modelos sobre ajustaban bajo configuraciones extremas (tamaños de lote pequeños, secuencias cortas o muchas épocas), mientras que otros lograron un balance adecuado entre entrenamiento y validación.

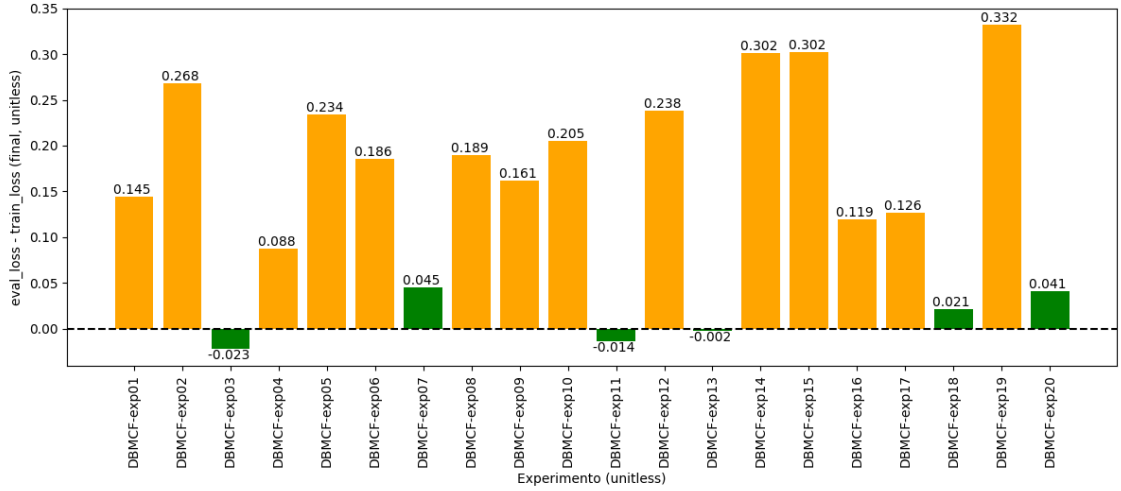


Ilustración 4 - Sobre-ajuste y Sub-ajuste por experimento

En la gráfica anterior se pueden ver en naranja los experimentos que presentan sobreajuste, mientras que en verde los más balaceados, mientras que en rojo los que presentan subajuste que por las características del *dataset* no se da esta situación.

Configuración Óptima

Tomando como referencia la Ilustración 3 (inferencias), Ilustración 4 (análisis de sobreajuste y subajuste), y con apoyo en las métricas *F1-Score, Precision, Recall* y *Loss*, se seleccionó el experimento Exp15 como la configuración más balanceada.

El Exp15 se destacó dentro de los modelos basados en DistilBERT, alcanzando un *F1-Score* del 96%, con una precisión superior al 90% y un *recall* bien equilibrado. Esta configuración empleó valores intermedios de hiperparámetros *batch size* de 12, seis épocas de entrenamiento y un *learning rate* de 3e-5, lo que permitió optimizar el rendimiento sin incurrir en sobreajuste.

Cabe destacar que otros experimentos, como el Exp12, presentaron resultados muy similares al Exp15, ya que sus configuraciones difieren mínimamente en los hiperparámetros utilizados. No obstante, al comparar el conjunto de los 20 experimentos realizados, se evidencia cómo incluso pequeñas variaciones en estos parámetros pueden influir significativamente en el desempeño del modelo.

Por el detalle de cada uno de los experimentos y los resultados de las métricas obtenidas dirigirse al ANEXO 3.3.

Comparación de DistilBERT con ajuste fino con otros modelos.

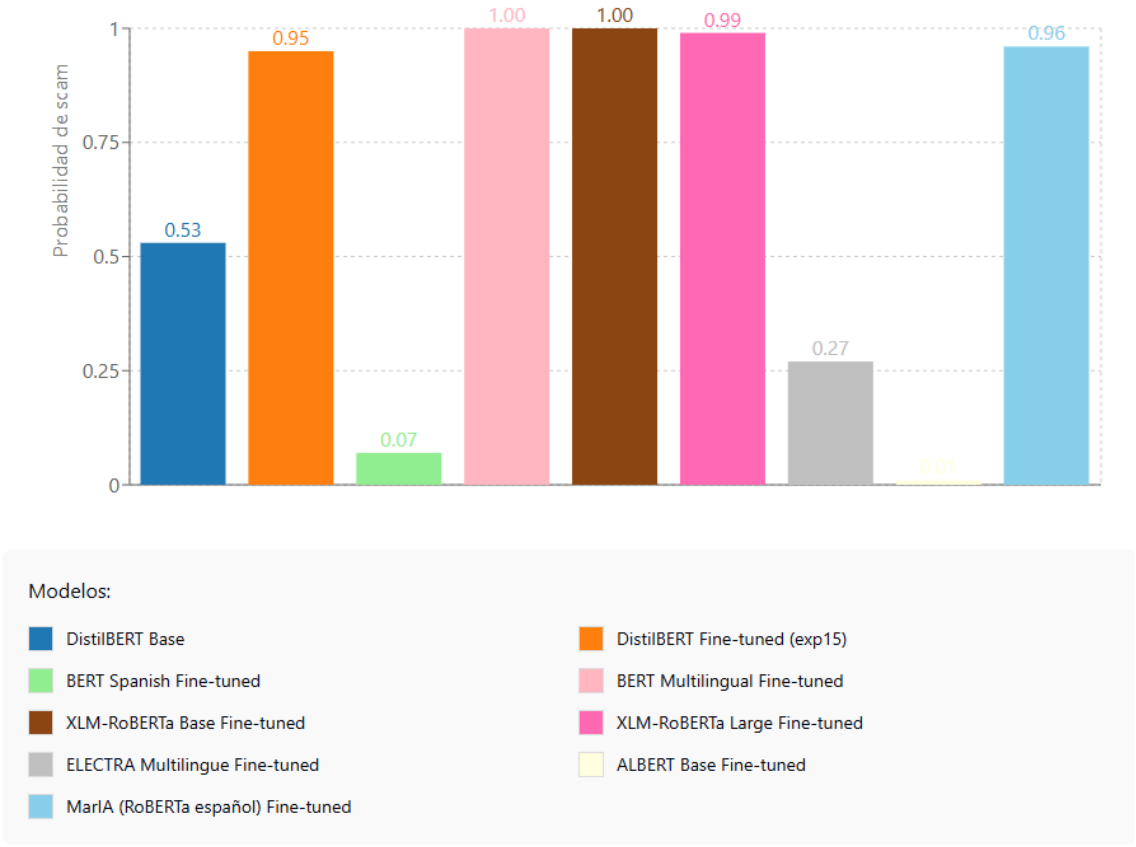


Ilustración 5 - Probabilidad de estafa por modelo ajustado finamente

La grafica anterior muestra que el modelo DistilBERT, tras un ajuste fino, alcanza niveles de desempeño comparables a los de XLM-RoBERTa también ajustado. Esto demuestra que DistilBERT, con un costo computacional significativamente menor, puede lograr resultados equivalentes a los de modelos más grandes como RoBERTa.

Confianza en los modelos

La siguiente grafica muestra que no todos los modelos identifican correctamente una estafa cuando se les presenta un caso específico. Mientras que algunos modelos no la detectan, DistilBERT y XLM-RoBERTa ajustados finamente sí lo hacen, lo que indica que son capaces de detectarla con confianza.

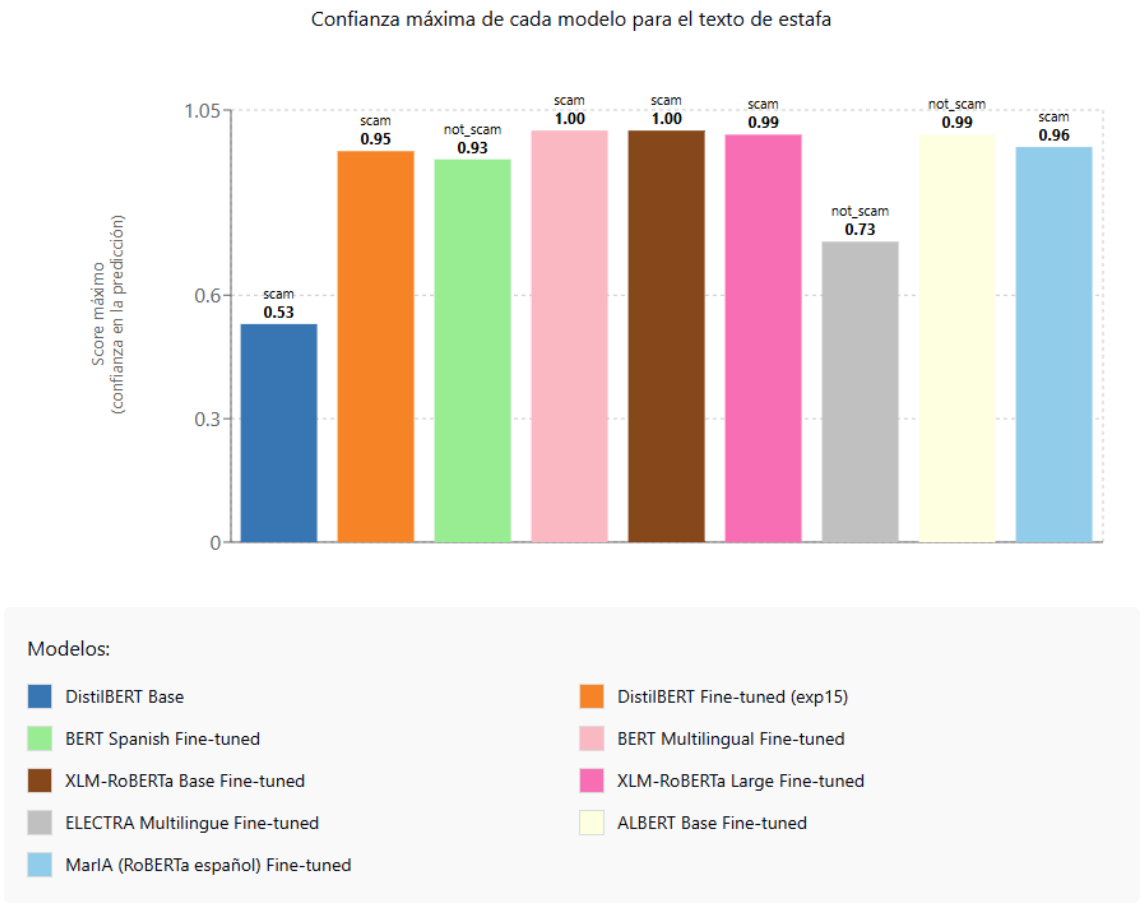


Ilustración 6 - Confianza en el resultado por modelo ajustado finamente

Decisiones según los resultados de Entrenamientos

En el ecosistema de modelos Transformer existen variantes reducidas como DistilBERT, TinyBERT [64] o MobileBERT [65], que buscan disminuir el consumo computacional y facilitar el despliegue en dispositivos con recursos limitados. Dentro de estas opciones, DistilBERT fue seleccionado como base inicial por su buen balance entre precisión y eficiencia

Durante la etapa inicial del desarrollo, se optó por DistilBERT como base debido a su balance entre precisión y eficiencia computacional, especialmente considerando la posibilidad de incorporarlo en dispositivos Android con recursos limitados [66].

Limitaciones en Textos Largos

El análisis reveló una limitación de DistilBERT, presentando una degradación significativa del rendimiento en textos extensos. Para textos superiores a 100 palabras, el modelo mostró pérdida de sensibilidad ante patrones de estafa, con *scores* de predicción que caían abruptamente incluso en casos con alta carga semántica fraudulenta.

XLM-RoBERTa como Solución Avanzada

Para superar las limitaciones de DistilBERT en textos largos, se evaluó XLM-RoBERTa, que demostró capacidades superiores:

- Consistencia en textos largos: Mantiene rendimiento sólido independientemente de la longitud del texto
- Capacidad multilingüe: Mejor adaptación a patrones del español y contexto latinoamericano
- Robustez semántica: Mayor capacidad para detectar estafas camufladas con lenguaje formal.

5.7 Evolución de la Estrategia de Inferencia

A partir de este momento llamaremos al modelo XLM-RoBERTa con ajuste fino simplemente “Roberta”.

En esta sección se describe la estrategia adaptativa diseñada para optimizar el desempeño de los modelos durante la inferencia, buscando un equilibrio entre precisión y eficiencia computacional. Esta estrategia se aplicó al análisis de textos de diversas longitudes, como los mensajes que pueden contener intentos de estafa y se validó mediante un conjunto de guiones representativos de distintos escenarios de fraude reales. Los guiones completos de estafa se encuentran en el ANEXO 3.4

La arquitectura escalonada de la estrategia clasifica los textos según su extensión y aplica el modelo más adecuado a cada caso. Para textos cortos (hasta 100 palabras) se empleó DistilBERT, que ofrece una buena precisión con bajo costo computacional, ideal para respuestas rápidas y alta concurrencia. En textos de longitud media (entre 100 y 250 palabras) se utilizó Roberta, capaz de realizar un análisis más profundo manteniendo la eficiencia y una latencia aceptable. Finalmente, los textos extensos (más de 250 palabras) se procesaron mediante una combinación de segmentación y análisis dual, dividiendo el texto en fragmentos óptimos de entre 80 y 100 palabras, respetando los límites de oración y con una ligera superposición para conservar el contexto.

En la fase de agregación de resultados, se consideró el puntaje máximo de estafa entre todos los fragmentos, de modo que un solo segmento sospechoso bastara para activar la alerta. Este enfoque no solo mejora la detección, sino que también permite identificar qué fragmento específico contiene la señal de fraude, facilitando su análisis posterior.

Tal como se muestra en el análisis del modelo hibrido disponible en el ANEXO 3.5, el rendimiento de DistilBERT comenzó a degradarse en textos largos, omitiendo señales relevantes que sí eran captadas por Roberta. La segmentación permitió recuperar detecciones perdidas en varios casos, demostrando su efectividad en escenarios complejos.

Posteriormente, durante las pruebas con el prototipo funcional, esta diferencia se acentuó: la precisión de DistilBERT disminuyó aún más en contextos reales y con datos ruidosos, lo que motivó la adopción definitiva de Roberta para todas las longitudes de texto.

5.8 Técnica de Juez de IA

Para garantizar la confiabilidad de las predicciones generadas por el modelo principal de SafeCall, se implementó una técnica de juez, consistente en utilizar un modelo de IA externo como verificador independiente. Este “juez” no reemplaza el modelo principal ni altera sus resultados; su función es proporcionar una segunda opinión que confirme la detección de posibles estafas, especialmente en casos ambiguos o difíciles de evaluar.

En este caso, se empleó GPT-3.5 Turbo[67] como modelo juez, dado que se trata de un *Large Language Model* (LLM, por su sigla en inglés) de gran escala, ampliamente entrenado y validado, capaz de ofrecer una evaluación objetiva y fundamentada de los resultados del modelo principal. Su uso permitió verificar la coherencia y precisión de las inferencias obtenidas, reforzando la confianza general en el sistema.

El costo de cada verificación es aproximadamente de un centavo de dólar (U$S 0.01), lo que representa un gasto mínimo en comparación con el valor que aporta como mecanismo adicional de validación. En términos de costo-beneficio, la implementación resulta ampliamente justificada.

Se reconoce que el uso de un modelo externo implica riesgos de privacidad, ya que los datos de las conversaciones se envían sin anonimización. No obstante, considerando que estamos en una etapa de desarrollo de un prototipo, se priorizó la obtención de una segunda opinión confiable para validar y ajustar el modelo principal. Esta estrategia permite fortalecer la precisión y consistencia de las predicciones en un contexto controlado, mientras se acumula evidencia sobre su desempeño.

De este modo, el juez mantiene su rol como complementario y de apoyo, asegurando la fiabilidad del prototipo sin convertirse en elemento central del proceso de decisión.

6. Arquitectura

Las decisiones arquitectónicas de SafeCall se tomaron en el marco del desarrollo de un Prototipo Funcional, con el objetivo de validar aspectos tanto técnicos como de usuario. En particular, se buscó comprobar que el sistema pudiera detectar llamadas sospechosas en menos de 30 segundos y con una precisión superior al 80%, al mismo tiempo que se evaluaban los flujos de interacción y la experiencia del usuario. Esto implicó priorizar soluciones sencillas y efectivas que permitieran validar rápidamente el concepto, manteniendo un equilibrio entre simplicidad de implementación y bajo costo operativo, sin descuidar los atributos de calidad definidos en los requerimientos.

6.1. Arquitectura del Sistema

6.1.1 Visión General

SafeCall implementa una arquitectura cliente-servidor compuesta por dos componentes principales: una aplicación móvil Android y un *backend* con API desplegado en la nube. La aplicación móvil, instalada en el dispositivo del usuario, se encarga de la experiencia de uso, el monitoreo de llamadas y la integración con servicios de Firebase [68], como App Distribution (para distribución de *builds* a *testers*), Analytics y Events (para monitoreo de uso y eventos de la aplicación) y Firestore (para almacenamiento y sincronización de datos). Por su parte, el *backend* concentra las tareas más intensivas de cómputo, incluyendo la transcripción automática de llamadas, el análisis de patrones mediante modelos de *machine learning* y la gestión de datos.

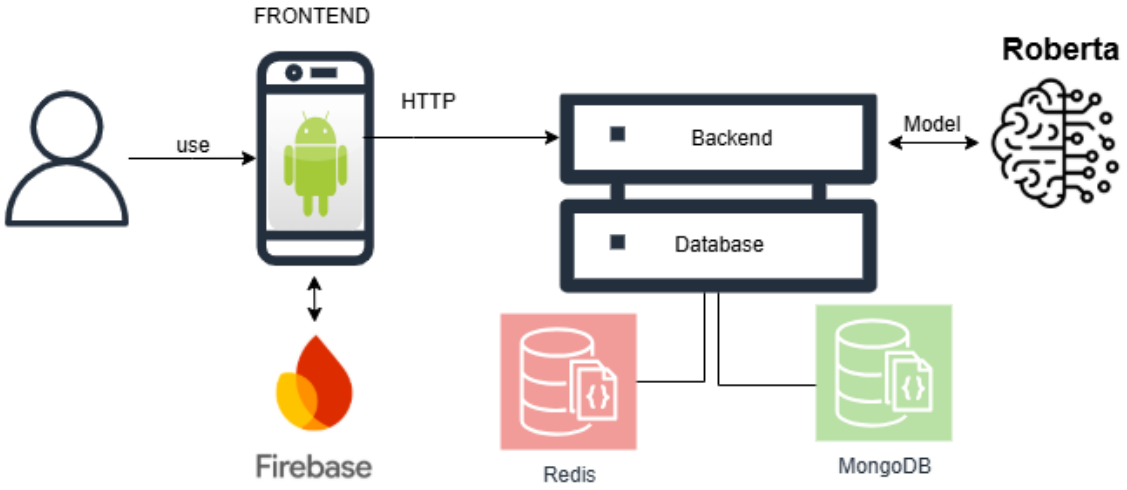


Ilustración 7- Diagrama de alto nivel de la solución

El flujo principal de la aplicación se comporta de la siguiente manera: cuando entra una llamada, el móvil revisa localmente si debe bloquearla automáticamente según las configuraciones del usuario y las listas de números sospechosos. Si la llamada sigue activa, se graba y al finalizar se envía el audio al *backend*. Allí se transcribe el contenido, se analiza con los modelos de detección y, si se identifican indicios de fraude y se devuelve una alerta al dispositivo. El usuario y también sus familiares (en caso de corresponder) reciben la notificación, lo que les permite estar informados de la situación.

Este proceso resume el ciclo de interacción entre el móvil y el *backend*: un cliente ligero enfocado en usabilidad y respuesta rápida, y un servidor robusto que asume el trabajo pesado de procesamiento.

6.1.2 Atributos de Calidad

El diseño arquitectónico de SafeCall se orienta principalmente a cumplir con tres atributos de calidad priorizados. En primer lugar, la *performance*, ya que el sistema debía ser capaz de detectar posibles estafas de manera rápida y precisa, garantizando tiempos de respuesta adecuados para la protección del usuario. En segundo lugar, la usabilidad, un aspecto esencial dado que la aplicación está dirigida también a adultos mayores, por lo que se buscó una experiencia sencilla, clara y accesible. Finalmente, la

deployabilidad, alcanzada mediante el uso de contenedores Docker, lo que simplifica y agiliza el despliegue en distintos entornos y asegura la portabilidad de la solución.

6.2. Diseño del Backend

6.2.1. Arquitectura del Backend

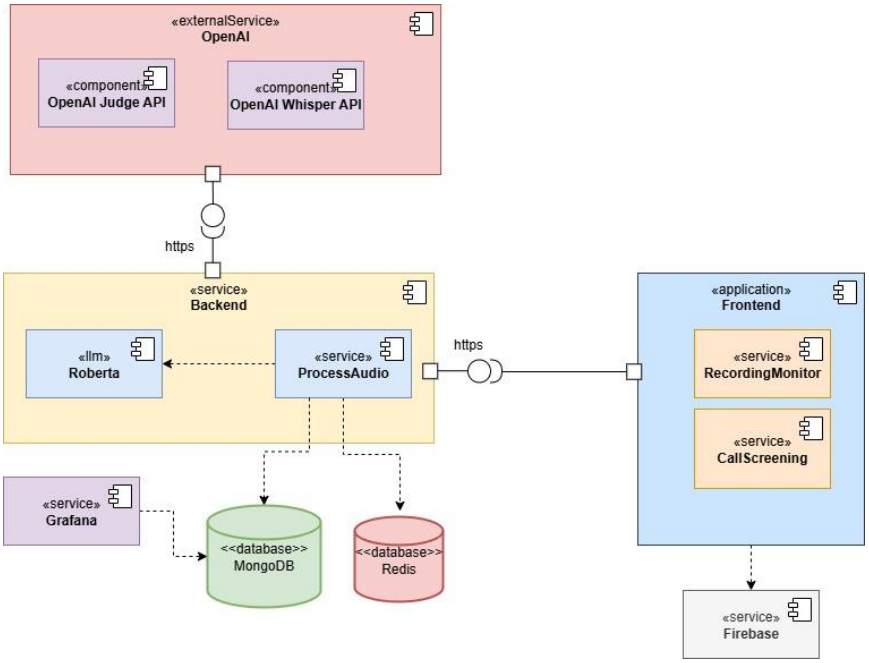


Ilustración 8 Diagrama Principal de Componentes

El *backend* de SafeCall implementa una arquitectura de API de inferencia diseñada para procesar audios de llamadas y detectar intentos de estafa mediante modelos de *machine learning*. Esta estructura separa la lógica de procesamiento pesado del *frontend* móvil, permitiendo ofrecer predicciones de manera rápida y confiable. La API se organiza en módulos especializados con responsabilidades claramente definidas: un módulo de preprocesamiento normaliza y valida los archivos de audio según su duración y características acústicas, un módulo de transcripción convierte el audio a texto mediante Whisper [69], un módulo de inferencia aplica el modelo Roberta para detectar patrones de estafa, y un módulo de verificación utiliza OpenAI GPT-3.5 Turbo como validador externo para fines académicos y de medición de calidad.

Los resultados y métricas se almacenan en MongoDB [70], mientras que un módulo de monitoreo registra tiempos de procesamiento y errores, integrándose con herramientas

como Grafana [71] para *dashboards* interactivos. Además, el *backend* ofrece servicios complementarios para la gestión de números sospechosos, consulta analítica y verificación del estado general de la aplicación, y su arquitectura modular permite actualizar cualquier componente o modelo de inferencia de manera independiente sin afectar el resto del sistema. Los detalles técnicos completos y documentación de los *endpoints* se encuentran en el ANEXO 6.

6.2.2. Flujo en el Backend

Cuando la aplicación móvil envía un audio al *backend*, este sigue un flujo de procesamiento secuencial.

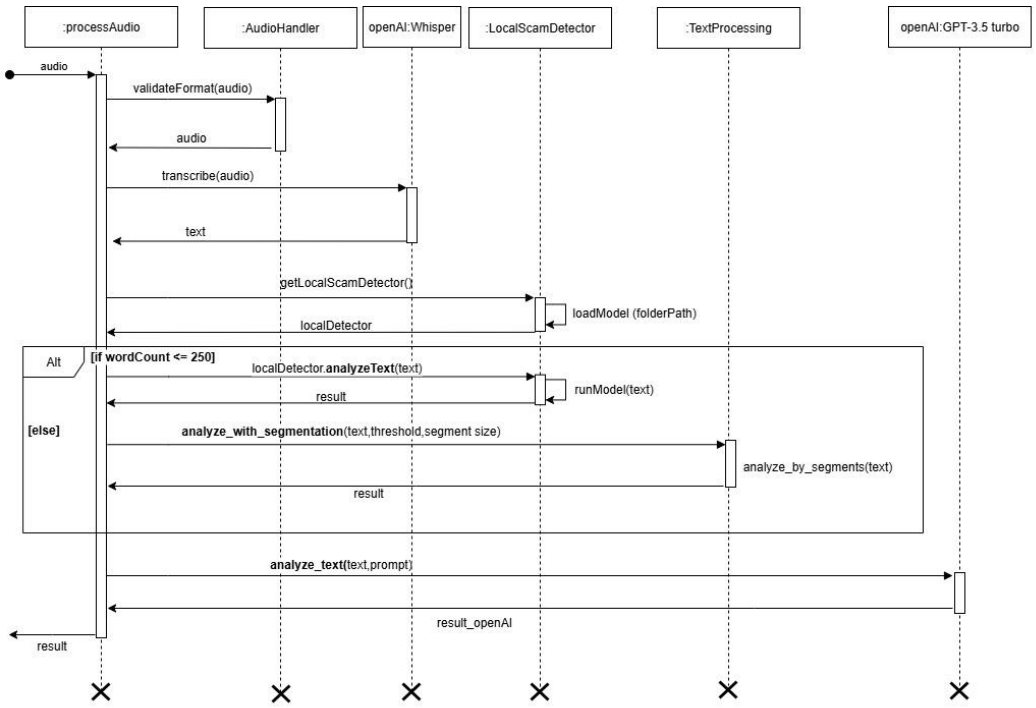


Ilustración 9 - Diagrama de flujo de procesamiento de audio en el backend

El proceso inicia cuando el *endpoint* de la API recibe el archivo de audio desde la aplicación móvil. Un módulo de procesamiento normaliza y valida el formato del audio, verificando su integridad y características técnicas. Si el archivo no cumple con los requisitos mínimos, se devuelve un error inmediato al cliente.

Una vez validado, el audio se envía al módulo de transcripción que convierte la voz en texto mediante Whisper. Tal como se describió en el Capítulo 5, el texto transcrito se

segmenta automáticamente según la cantidad de palabras, lo que facilita su posterior análisis en los módulos de procesamiento y clasificación. Esta segmentación permite identificar con mayor precisión los fragmentos específicos que contienen patrones de estafa, especialmente en llamadas largas o con múltiples temas de conversación.

La transcripción segmentada se envía al módulo de inferencia, que aplica nuestro modelo Roberta, especializado en la detección de llamadas fraudulentas. El modelo analiza cada fragmento de forma independiente, generando un score de probabilidad de estafa para cada uno. La respuesta final considera el score máximo entre todos los segmentos, de manera que un solo fragmento sospechoso puede activar la alerta, generando la predicción correspondiente.

Una vez obtenida la respuesta de nuestro modelo Roberta, el audio también es procesado mediante el Modelo Juez de OpenAI GPT-3.5 Turbo. Este componente actúa como verificador independiente, permitiendo comparar la precisión de nuestro sistema con fines académicos y de evaluación de calidad. Aunque el Juez puede retornar una respuesta diferente a la predicción de nuestro modelo, la respuesta que se devuelve al usuario siempre es la de Roberta, ya que el Juez se utiliza únicamente para medir el desempeño del sistema.

Los resultados completos del análisis del audio, las transcripciones, los scores de cada segmento, las predicciones de ambos modelos y los tiempos de procesamiento se registran en MongoDB para su posterior uso en métricas académicas y análisis histórico. Esta información permite tanto la evaluación continua del sistema como la generación de estadísticas de rendimiento.

Finalmente, el *backend* devuelve la predicción a la cliente obtenida por el modelo Roberta y registra métricas de rendimiento y errores para monitoreo, asegurando consistencia y escalabilidad en el procesamiento de llamadas. Durante todo este proceso, el sistema mantiene trazabilidad completa, registrando cada etapa del procesamiento para facilitar diagnóstico de problemas y optimización del sistema.

Para mayor detalle sobre el procesamiento del audio en el *backend*, véase el diagrama de flujo completo en el ANEXO 4.

6.2.3. Decisiones de Arquitectura del Backend

En el *backend* se definió una arquitectura modular, compuesta por etapas de preprocesamiento, inferencia y monitoreo, lo que facilita la mantenibilidad y permite reemplazar componentes de forma independiente sin afectar el resto del sistema.

A nivel de almacenamiento se adoptó una estrategia dual, combinando Redis [72] como base de acceso rápido y temporal para información operativa (como los números sospechosos) y MongoDB como base de datos persistente para resultados de análisis y registros de métricas.

El despliegue se realizó mediante contenedores Docker ejecutados sobre instancias EC2 de AWS [73], lo que garantiza portabilidad, consistencia y capacidad de *rollback* entre versiones. Para mantener direcciones fijas, se configuró una Elastic IP [74].

El Juez de IA se implementa como servicio desacoplado, lo que permite actualizarlo sin afectar el flujo principal de inferencia.

Para monitoreo y observabilidad, se incorporó Grafana, proporcionando *dashboards* en tiempo real que apoyan el diagnóstico, la optimización y el mantenimiento del sistema. Todas estas decisiones se documentaron formalmente mediante Registros de Decisión de Arquitectura (ADR), disponibles en el ANEXO 4.

6.3. Diseño de la Aplicación Móvil

6.3.1. Integración con el Backend

La aplicación móvil se comunica con el *backend* mediante protocolos HTTPS, empleando certificados auto firmados generados para el entorno de desarrollo y despliegue inicial. Esta elección permitió establecer canales seguros de comunicación sin depender de autoridades de certificación externas, manteniendo el control sobre las claves y reduciendo costos.

Los servicios principales incluyen el monitoreo de grabaciones, que notifica al *backend* sobre cada nuevo archivo generado, y el filtrado de llamadas, que verifica números entrantes en listas locales y en línea. Esta integración asegura consistencia entre datos

locales y remotos, permitiendo decisiones de alerta confiables y comunicación segura de extremo a extremo.

6.3.2. Arquitectura Frontend

La aplicación móvil sigue un patrón arquitectónico MVVM (Model–View–ViewModel), que permite separar responsabilidades, facilitar la mantenibilidad y favorecer la escalabilidad del sistema.

Ilustración 10 - Diagrama de capas de modelo MVVM

Para ver la vista de componentes detallada de la aplicación dirigirse a ANEXO 4.

En la capa *Model* se concentra la lógica de negocio y la comunicación con el *backend*. Para ello se utilizó Retrofit y OkHttp [75], bibliotecas que simplifican y optimizan la gestión de solicitudes HTTP, junto con kotlinx-serialization y Gson para transformar datos en formato JSON [76] de manera confiable. Además, se integraron servicios de Firebase mencionados en la sección 6.1.1.

La capa ViewModel expone el estado de la aplicación a la interfaz de usuario. Se uso StateFlow, herramienta que permite manejar y observar cambios en el estado de forma reactiva, asegurando que la interfaz responda de inmediato a las acciones del usuario sin necesidad de recargar manualmente la información.

En la capa de presentación (*View*) se organizan las pantallas mediante Navigation Compose, lo que facilita definir y mantener el flujo de navegación entre vistas (inicio, detección de llamadas, historial y configuración). La interfaz incorpora mecanismos de alerta y notificación que mantienen al usuario informado sobre llamadas sospechosas, cambios en el estado del servicio y eventos relevantes detectados por el sistema.

En cuanto a la configuración y despliegue, la aplicación soporta diferentes entornos (desarrollo, *release* y preproducción), con parámetros específicos para *backend* y firma de la aplicación. El proceso de construcción se realiza con Gradle, integrando ProGuard en modo *release* para optimizar y proteger el código.

6.3.3. Flujo en la Aplicación Móvil

En la aplicación móvil se distinguen dos flujos principales de funcionamiento, ambos orientados a proteger al usuario frente a llamadas potencialmente fraudulentas.

En el siguiente diagrama de alto nivel se puede visualizar el flujo descrito anteriormente.

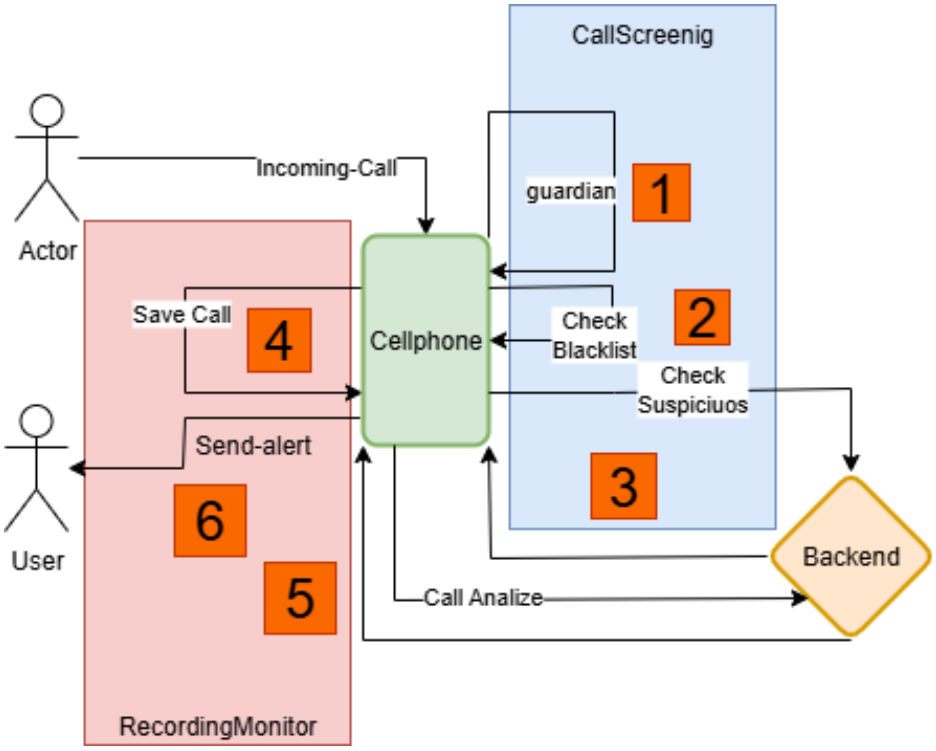


Ilustración 11 - Diagrama Alto Nivel Flujo Frontend.

El primer flujo se activa en el momento en que se registra una llamada entrante, a través de un servicio residente en el dispositivo (llamado CallScreening). Antes de que el usuario conteste, la aplicación realiza una serie de verificaciones: en primer lugar, comprueba si el "modo guardián" está habilitado (paso 1 en el diagrama), bloqueando automáticamente todo número que no se encuentre en la agenda de contactos. Si la llamada supera ese filtro, se valida si el número no está incluido en la lista negra local ("Blacklist"), donde se almacenan los números bloqueados por el propio usuario (paso 2). En caso de no encontrarse en ella, el sistema consulta al *backend* para determinar si el número está reportado como sospechoso (paso 3). Si la respuesta es positiva, se alerta de forma sonora al usuario sobre el posible riesgo, dándole la posibilidad de decidir cómo proceder antes de responder.

El segundo flujo comienza una vez finalizada la llamada. Un servicio independiente se encarga de monitorear las grabaciones almacenadas en el dispositivo dentro de la carpeta "Recordings" (paso 4). Al detectar una nueva grabación, esta se envía automáticamente al *backend* (paso 5), donde se aplica el análisis de audio (descrito en 6.2.2). Si el *backend* determina que la llamada tiene características de posible estafa, el

servicio de monitoreo (llamado RecordingMonitor) en el *frontend* no solo notifica al usuario, sino que también envía un SMS a los contactos de confianza seleccionados por este, alertándolos sobre el intento de fraude (paso 6).

De esta forma, el *frontend* funciona como una primera línea de defensa inmediata, mientras que el *backend* opera como un módulo de verificación posterior más exhaustivo, complementando la protección del usuario.

Los detalles de los flujos mediante diagramas de secuencia, se encuentra en el ANEXO 4

6.3.4. Interfaz de usuario

La interfaz de usuario de SafeCall ha sido diseñada con un enfoque minimalista y accesible, priorizando la usabilidad como atributo de calidad central. Se buscó garantizar la facilidad de uso y la comprensión inmediata, especialmente para adultos mayores.

Ilustración 12 - Pantalla Bienvenida

La pantalla de inicio permite acceder a la aplicación sin necesidad de introducir credenciales, reduciendo barreras tecnológicas y facilitando un uso ágil desde el primer momento.

Ilustración 13 - Menú Principal

Una vez dentro, el menú principal centraliza la navegación, ofreciendo acceso directo a todas las funcionalidades y mostrando de manera permanente un indicador del estado del servicio mediante un código de colores tipo semáforo: verde para funcionamiento normal, amarillo para rendimiento reducido y rojo para interrupciones.

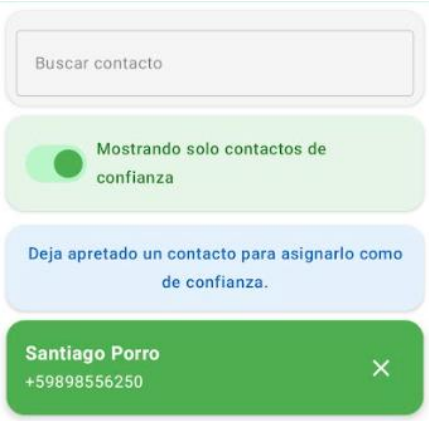


Ilustración 14 - Zoom a pantalla de Gestión de Contactos

La gestión de contactos se presenta de manera clara y ordenada, destacando aquellos de confianza y permitiendo seleccionar fácilmente a quien notificar en caso de detección de llamadas sospechosas.

Ilustración 15 – Zoom a objeto en lista de Grabaciones

La sección de grabaciones centraliza los archivos generados por la aplicación, ofreciendo reproducción, pausa, compartición a través de servicios externos y eliminación de contenidos, lo que garantiza un control seguro sobre la información almacenada.

Ilustración 16 – Zoom a pantalla de Historial de Llamadas

De manera complementaria, el historial de llamadas presenta registros diferenciados por color según la acción tomada (contestadas o perdidas) e incluye información detallada de número, fecha, hora y duración, facilitando un seguimiento completo de la actividad telefónica.

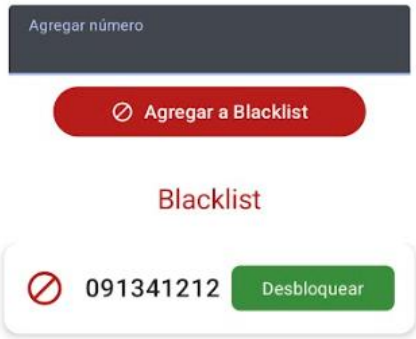


Ilustración 17 – Zoom a pantalla de “Blacklist” (Números bloqueados)

La gestión de números no deseados se realiza mediante la “Blacklist”, que permite agregar o eliminar números, asegurando un control directo sobre la seguridad y personalización de la experiencia de comunica

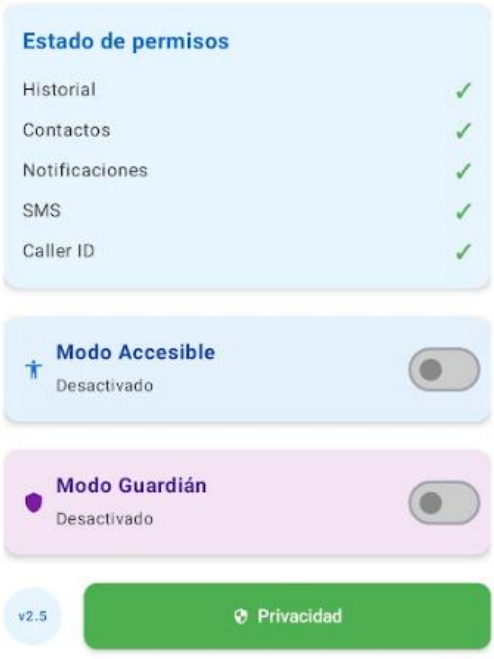


Ilustración 18 – Zoom a pantalla de Configuración

Por último, la sección de configuración permite revisar y ajustar permisos otorgados (llamadas, contactos, notificaciones, envío de SMS), activar un modo accesible que

incrementa tipografía y contraste, y un modo guardián de llamadas. Además, se incluye información sobre la versión de la aplicación, historial de actualizaciones, política de privacidad y opciones relacionadas con la identificación de llamadas y detección de comunicaciones no deseadas.

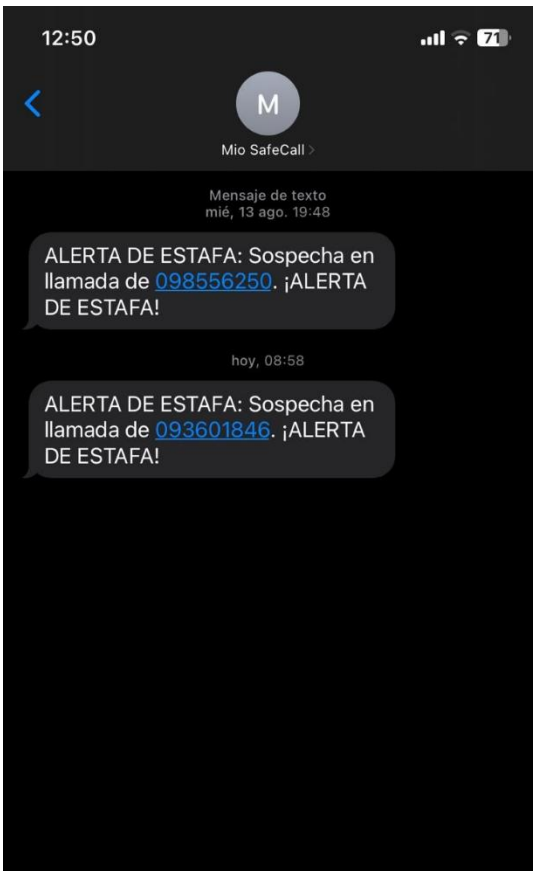


Ilustración 19 – Ejemplo de Notificaciones recibidas por un Contacto (SMS)

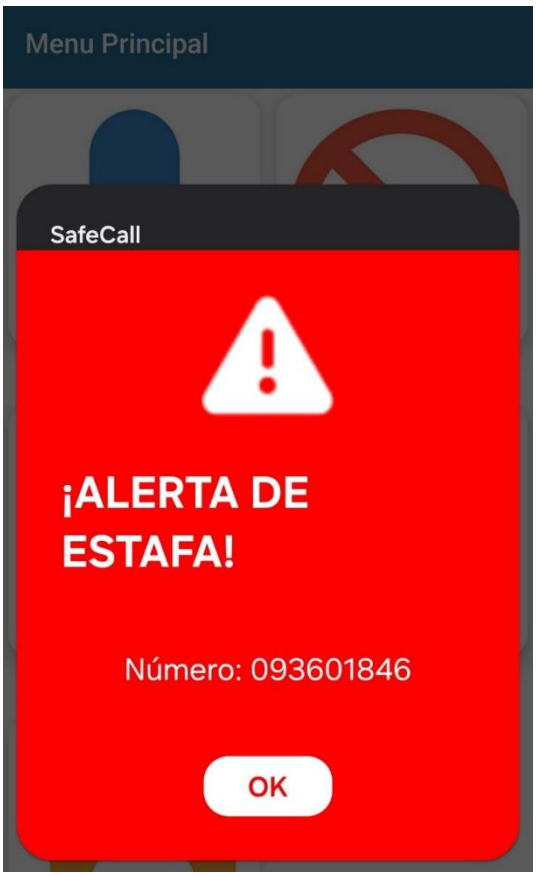


Ilustración 20 – Ejemplo de mensaje de Alerta de estafa

Esta sección se complementa con un sistema de alertas y notificaciones, que informa al usuario sobre posibles riesgos, utilizando mensajes emergentes, códigos de color y señales sonoras, de forma que las decisiones se puedan tomar de manera inmediata sin necesidad de navegar por la aplicación.

6.4 Análisis de Atributos de Calidad claves

Performance

Uno de los atributos de calidad priorizados en la arquitectura fue la *performance*, valioso para garantizar una respuesta lo más rápida posible en el análisis y clasificación de llamadas.

Desde el *frontend*, el servicio CallScreening se activa automáticamente al detectarse una llamada entrante, por lo que el sistema debe realizar la verificación del número en cuestión, antes incluso del primer timbrado del teléfono. Para lograr esa respuesta casi instantánea, se incorporó Redis al *backend* como base de datos en memoria de tipo clave-valor, utilizada como mecanismo de *caching* de alta velocidad para almacenar y consultar números sospechosos. Esta elección permitió resolver las verificaciones en pocos segundos, emitiendo una alerta inmediata al usuario cuando el número coincidía con registros previamente catalogados como potencialmente fraudulentos.

Además de ello. el *backend* implementa validación temprana del formato e integridad del audio al recibirlo, devolviendo errores al instante cuando no cumple los requisitos, sin iniciar el procesamiento posterior.

El despliegue del sistema se realizó mediante contenedores Docker ejecutados sobre instancias EC2 de AWS. La instancia t2.large (2 vCPUs, 8 GB RAM) fue seleccionada tras evaluar alternativas como t2.medium (4 GB RAM, insuficiente para contenedores Docker + modelo ML) y t2.xlarge (16 GB RAM, sobredimensionada por el alto costo). La configuración elegida permite ejecutar simultáneamente los contenedores del *backend* FastAPI, Redis, MongoDB y el modelo Roberta. Inicialmente se consideró correr localmente la transcripción de audio, pero al requerir GPU y por sus altos costos fue necesario usar el servicio de OpenAI. Si bien esta decisión introduce latencia de red, ejecutar Whisper localmente en CPU hubiera resultado en tiempos de procesamiento considerablemente mayores, por lo que delegar la transcripción a OpenAI representa una mejora de performance bajo las restricciones presupuestarias del proyecto.

Con esta configuración se logró cumplir el objetivo de mantener la respuesta por debajo de los 30 segundos, alcanzando un balance adecuado entre costo y *performance*. Para visualizar más detalles técnicos dirigirse al ANEXO 4.

Usabilidad

Otro atributo de calidad priorizado dentro de la arquitectura fue sin dudas la usabilidad.

Para ello, el diseño de la interfaz se basó en las heurísticas de Nielsen [77] y en patrones de experiencia de usuario (UX, por su sigla en inglés) [78], buscando una navegación clara, coherente y accesible. Se priorizó la simplicidad, la visibilidad del estado del sistema y la facilidad de uso, asegurando que las funcionalidades más importantes pudieran ser utilizadas sin dificultades por el público objetivo, especialmente adultos mayores.

La aplicación incorpora mecanismos que facilitan la interacción y reducen la carga cognitiva, como el reconocimiento sobre el recuerdo en *login* y reproducción de grabaciones, un diseño minimalista que resalta lo esencial, notificaciones claras que mantienen al usuario informado y diálogos de confirmación que previenen errores en acciones críticas. Asimismo, se garantiza consistencia con los estándares de Android, flexibilidad de uso mediante ajustes personalizables y control total sobre elementos como la “blacklist”, ofreciendo así una mejor experiencia.

La adopción de un sistema de navegación unificado mediante un menú inferior en todas las pantallas garantizó consistencia y redujo la carga cognitiva, mientras que la arquitectura modular permitió incorporar ayudas contextuales, etiquetas descriptivas y un tutorial de *onboarding* sencillo, sin afectar la estructura principal de la aplicación.

El detalle de Heurísticas aplicadas por cada una de las pantallas se encuentra en el ANEXO 4

En cuanto a lo tecnológico, el uso de Jetpack Compose posibilitó el desarrollo de interfaces declarativas y reactivas, donde los cambios en los datos se reflejan automáticamente en pantalla, mejorando la percepción de inmediatez y coherencia visual [79].

La gestión de estado se implementó mediante StateFlow, garantizando una sincronización eficiente entre la capa de datos y la visualización. Esta decisión redujo el riesgo de inconsistencias y contribuyó a mantener la interfaz actualizada en tiempo real, incluso ante fallos de red o actualizaciones del *backend*.

El uso de temas dinámicos (o *themes*, por su traducción en inglés) permitió ajustar el contraste, la tipografía y otros elementos visuales, facilitando la lectura en diferentes condiciones de visibilidad y permitiendo la incorporación de un modo de alta visibilidad para usuarios con baja visión. Esta funcionalidad se nombró como “Modo Accesible”.

En conjunto, estas decisiones no solo respondieron a requerimientos funcionales, sino que también sostuvieron el atributo de calidad de usabilidad, asegurando una experiencia accesible, predecible y confiable para el usuario final.

7. Calidad

La estrategia de calidad implementada se fundamentó en un enfoque centrado en el usuario, considerando que la calidad se define en función de su experiencia y satisfacción. En este sentido, las decisiones adoptadas priorizaron la evaluación del modelo Roberta en inferencia, analizando su capacidad para detectar correctamente intentos de estafa, su confiabilidad ante distintos contextos de entrada y la velocidad de respuesta, aspectos fundamentales para garantizar una experiencia fluida y segura para el usuario; la validación de la experiencia de usuario, mediante pruebas de usabilidad orientadas a determinar la facilidad con que los usuarios podían realizar las tareas propuestas y comprender los resultados presentados por el sistema; y la aplicación de prácticas de ingeniería de *software* ágiles, que promovieron la mejora continua, la integración temprana de retroalimentación y la adaptación rápida ante cambios en los requerimientos o nuevos descubrimientos durante el desarrollo. La selección de estos métodos se justificó en función de los objetivos académicos del proyecto y las características técnicas de la solución.

7.1 Evaluación de Modelos de Inferencia

Esta evaluación permitió seleccionar la combinación de modelos más adecuada a lo largo de los distintos *releases*. Uno de los factores a considerar para ello, fue la degradación observada en el modelo DistilBERT. (Las gráficas se encuentran disponibles en el ANEXO 3.5).

La precisión en la detección de estafas telefónicas es crucial, ya que determina directamente el nivel de protección que el sistema ofrece a los usuarios. Para evaluar el desempeño del modelo, se utilizan *endpoints* que facilitan la visualización de las respuestas obtenidas, así como *dashboards* en Grafana, donde se generan *Key Performance Indicators* (KPIs,) que resumen métricas relevantes, como precisión, *recall* y tasa de error.

La evaluación del modelo se basa en métricas de clasificación binaria derivadas de la matriz de confusión [80], la cual clasifica las predicciones en cuatro posibles escenarios:

- Verdaderos Positivos (TP) - estafas correctamente identificadas.
- Falsos Positivos (FP) - llamadas legítimas marcadas erróneamente.
- Verdaderos Negativos (TN) - llamadas legítimas correctamente clasificadas.
- Falsos Negativos (FN) - estafas no detectadas.

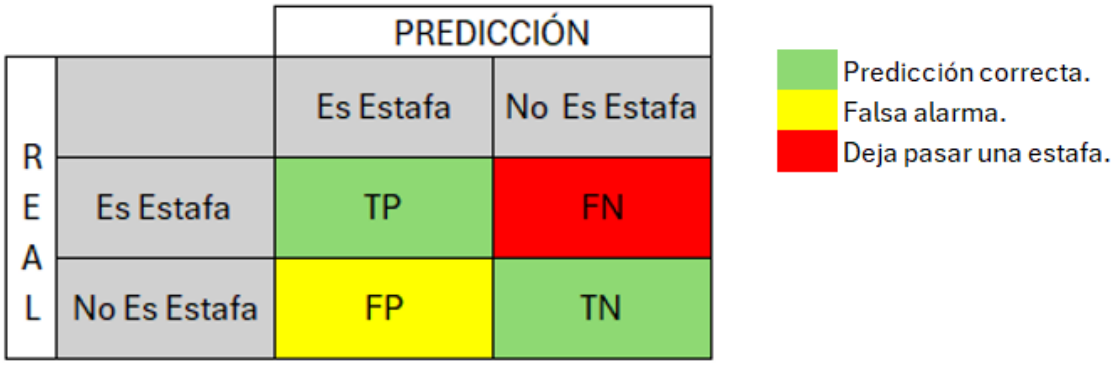


Ilustración 21 - Representación de la matriz de confusión para clasificación binaria

Pruebas de Precisión y Recall en Inferencia

Las pruebas de ajuste fino del modelo, diseñadas para garantizar su calidad y desempeño, se documentan en detalle en el capítulo 5 (IA).

La evaluación del sistema se realizó mediante un servicio interno que permite verificar el resultado de cada análisis de un texto de estafa transcrito frente a un valor predefinido. Para cada caso, se indicaba manualmente si correspondía a una estafa (*true*) o no (*false*), lo que permitió calcular métricas de confusión como precisión, *recall* y *F1-score*. El ejemplo de una *request* puede encontrarse en el ANEXO 5.

Este enfoque permite ver el comportamiento del modelo para cada caso evaluado. Además, en el servicio principal de detección el sistema combina la predicción del modelo local Roberta con una validación adicional usando OpenAI GPT-3.5-turbo (Juez-ia), lo que permite detectar discrepancias y evaluar la robustez de las predicciones. La concordancia entre ambos modelos alcanzó un 82,3%, según las métricas registradas durante las pruebas. (Captura con evidencia del valor en ANEXO 5)

Las fórmulas aplicadas para cada métrica son las siguientes: [81]

Nombre	Fórmula	Propósito
Accuracy	(TP + TN) / Total	Proporción de predicciones correctas.
Precisión	TP / (TP + FP)	Tasa de verdaderos positivos sobre positivos predichos.
Recall	TP / (TP + FN)	Proporción de estafas realmente detectadas.
F1-Score	2 × (Precision × *Recall*) / (Precision + *Recall*)	Media armónica que balancea ambos aspectos.

Tabla 4 – Fórmulas de Métricas

Evaluación Release 2 (Modelo hibrido DistilBERT + Roberta)

Tomando un conjunto de pruebas de 34 transcripciones de conversaciones de estafas y otras que no lo eran, se obtuvieron los siguientes resultados:

- Precisión: 100% (0 falsos positivos, los indicados como estafa lo eran)
- *Recall*: 35.7% (estafas detectadas)
- *Accuracy*: 47.1% (clasificaciones correctas)
- Tiempo de procesamiento: 3.9s promedio

Métrica	**Resultado**
Accuracy	47.1%
Precisión	100.0%

Estafas Detectadas	10/28 = 35.7%
Total Evaluaciones	34
Confianza Promedio Modelos	84.8%

Tabla 5 - Resultado evaluación release 2

De los resultados obtenidos, fue posible hacer la distinción entre Modelos como se muestra en la siguiente tabla.

Métrica	**DistilBERT**	**Roberta**
Accuracy	33.3%	70.0%
Precisión	100.0%	100.0%
Estafas Detectadas	2/14 = 14.3%	5/8 = 62.5%
Total Evaluaciones	18	10

Tabla 6 - Distinción de modelos

DistilBERT demostró ser inadecuado para producción al perder la gran mayoría de las estafas reales (12 de las 14 estafas de prueba no fueron detectadas), fallando en su propósito de detectar correctamente los fraudes. Aunque era más rápido, esta ventaja es irrelevante cuando el sistema no detecta las amenazas. Con esta evidencia, se decidió migrar a Roberta debido a su capacidad superior para identificar estafas, reduciendo drásticamente los casos perdidos.

Evaluación Release 3 (Optimización Roberta)

Tras optimizaciones del modelo y pipeline de procesamiento:

- Tiempo de procesamiento: 12s promedio (reducción del 48% desde Release 1 considerando audios de mayor duración)

- Percepción de usuario: 100% (5/5 usuarios) consideraron tiempo aceptable

Evaluación Final - Modelo Roberta en Producción

Métrica	**Resultado**
Accuracy	85.3%
Precisión Roberta	85.3%
Estafas Detectadas	67/71 = 94.4%
Total Evaluaciones	95
Confianza Promedio Roberta	98.3%

Tabla 7 - Evaluación Modelo Roberta

El modelo Roberta (XLM-RoBERTa ajustado) alcanzó un *accuracy* de 85,3% en la evaluación con 95 diálogos reales transcritos. El *recall* de 94,4% (67 de 71 estafas detectadas) es especialmente relevante desde la perspectiva de confiabilidad, ya que reduce la probabilidad de falsos negativos.

Estos resultados pueden visualizarse en el ANEXO 5.

7.2 Pruebas

Pruebas de usabilidad

La validación del prototipo de SafeCall se llevó a cabo mediante múltiples técnicas complementarias de evaluación de usabilidad, combinando enfoques cualitativos y cuantitativos. Principalmente se realizaron pruebas basadas en tareas [82], en las que los usuarios ejecutaron tres escenarios representativos (agregar un contacto de confianza, denunciar un número sospechoso y recibir una llamada con análisis del sistema). El refinamiento de la propuesta se basó en el *feedback* cualitativo y cuantitativo obtenido durante estas sesiones.

A su vez, se empleó el método *Think Aloud*, donde los participantes comentaron sus pensamientos y decisiones mientras interactuaban con la interfaz, permitiendo identificar confusiones y errores de navegabilidad. Al finalizar cada tarea, este nivel de errores era medido en una escala de: Sin errores, con errores mínimos, con muchos errores, no se completó.

Además de ello, al finalizar con cada tarea solicitada, se les solicitó indicar un nivel de satisfacción y experiencia con el comportamiento y usabilidad de la aplicación en una escala del 1 al 5.

Las pruebas fueron realizadas con el siguiente conjunto de usuarios:

Usuario	**Edad**	**Profesión**	**Nivel tecnológico**	**Comentario**
U1	80	Jubilada	Bajo	Usa WhatsApp, llama, no instala aplicaciones nuevas.
U2	47	Psicóloga	Medio	Usa apps laborales, no sabe configurar teléfonos.
U3	19	Estudiante	Bajo-Medio	Usa celular frecuentemente, instala aplicaciones, pero posee poco conocimiento tecnológico/manejo de herramientas
U4	62	Sanitario	Bajo	Usa celular solo para llamadas y mensajes
U5	58	Escribana	Bajo-Medio	Usa celular y Whatsapp para llamadas y mensajes, no instala aplicaciones por su cuenta

Tabla 8 - Pruebas con usuarios

En la siguiente Tabla, se muestran las tareas solicitadas por cada uno de los releases.

	Tarea 1	**Tarea 2**	**Tarea 3**	**Tarea 4**
Release 1	Simular una estafa, recibir alerta y ver detalle	Ver historial de llamadas	Agregar número a Blacklist (bloqueados) y recibir llamada del número	N/A
Release 2	Agregar un contacto de confianza	Simular una estafa y recibir notificación como contacto de confianza	Bloquear y denunciar un número	Escuchar y compartir una grabación
Release 3	Agregar un contacto de confianza	Activar “Modo Guardian” y recibir una llamada desconocida	Eliminar Grabación	Salir de la aplicación

Tabla 9 - Tareas solicitadas por release

Métrica	**Fórmula**	**Descripción**
Tasa de completitud	(Tareas completadas / Total de tareas) × 100	Porcentaje de tareas que los usuarios lograron completar exitosamente
Satisfacción promedio	Σ(Satisfacción individual) / Total de tareas	Promedio de las puntuaciones de satisfacción (escala 1-5)

Tasa de errores	(Tareas con errores / Total de tareas) × 100	Porcentaje de tareas completadas con errores (mínimos o muchos errores)

Tabla 10 - Métricas para pruebas

Los resultados generales, pueden verse en la tabla a continuación.

	Release 1	**Release 2**	**Release 3**
Usuarios evaluados	5	5	5
Escenarios probados	3	4	4
Tareas totales evaluadas	15	20	20
Tasa de completitud	86.7% (13/15)	95% (19/20)	100% (20/20)
Satisfacción promedio	3.7/5	4.2/5	4.7/5
Tasa de errores	80% (12/15)	55% (11/20)	30% (6/20)

Tabla 11 - Resultado de pruebas con usuarios

El detalle completa de las pruebas se encuentra en el ANEXO 5.2

Los resultados de las pruebas de usabilidad muestran una mejora consistente entre *releases*, evidenciando que las modificaciones realizadas sobre la interfaz y la experiencia de usuario tuvieron un impacto positivo significativo.

El aumento en la tasa de completitud indica que las barreras iniciales de navegación y comprensión fueron reducidas considerablemente. Este incremento se correlaciona directamente con las modificaciones realizadas en la interfaz entre *releases*, documentadas en la sección 8.5.1 (Resultados por Release), donde se detallan las mejoras implementadas en respuesta al *feedback* de usuarios.

Además de ello, el aumento en la satisfacción promedio, refleja no solo la capacidad de los usuarios para completar tareas, sino también una percepción más positiva de la

experiencia general. Este aumento está vinculado a la aplicación y mejora continua de las heurísticas de Nielsen (descritas en la sección 6.3.4), particularmente en aspectos de prevención de errores, visibilidad del estado del sistema y consistencia de la interfaz.

Pruebas de Portabilidad

Se validó compatibilidad en 6 dispositivos para verificar funcionalidad de grabación de llamadas:

Dispositivo	**SO**	**Resultado**	**Notas**
Samsung Galaxy A55	Android 15	✓	Grabación nativa funcional
Samsung Galaxy A55	Android 12	✓	Grabación nativa funcional
Samsung Galaxy A15	Android 14	✓	Grabación nativa funcional
Samsung Galaxy J7	Android 8	✗	Sin función grabación nativa
Xiaomi Redmi 10	Android 11	✗	Sin API grabación disponible
Huawei P30 Lite	Android 11	✗	Grabación nativa no funcional
Emulador Pixel 5	Android 13	✓	Grabación simulada funcional
Emulador Nexus 6P	Android 9	✗	Limitación de emulación en versiones antiguas
Emulador Pixel 6	Android 14	✓	Grabación simulada funcional
Emulador Pixel 4 XL	Android 12	✓	Grabación simulada funcional

Tabla 12 - Resultado pruebas portabilidad por dispositivos

La tasa de compatibilidad en dispositivos físicos fue aproximadamente 50% (3/6 dispositivos) para grabación nativa. Las pruebas confirmaron que esta funcionalidad

requiere Android 11 o superior y, en la práctica, funciona de manera confiable principalmente en dispositivos Samsung.

Estas pruebas, realizadas también en emuladores, permitieron identificar limitaciones del sistema y validar el alcance del Prototipo, definiendo los requerimientos mínimos y documentando posibles escenarios de incompatibilidad futura.

Pruebas de Interoperabilidad

Se realizaron validaciones con los tres principales proveedores de telefonía en Uruguay: ANTEL, Claro y Movistar.

Proveedor	**Resultado**	**Observaciones**
ANTEL	✓ Exitoso	Verifica que la aplicación funciona con líneas ANTEL y que la detección de estafas no depende del proveedor.
Claro	✓ Exitoso	Confirma compatibilidad con líneas Claro y asegura la correcta grabación y alertas, incluso con números privados en algunos casos.
Movistar	✓ Exitoso	Garantiza que los usuarios de Movistar reciban la misma funcionalidad, asegurando interoperabilidad completa.

Tabla 13 - Pruebas de Interoperabilidad

Tasa de compatibilidad: 100 % (3/3 proveedores)

Se realizaron llamadas simuladas de estafa cubriendo todas las combinaciones posibles entre proveedores (por ejemplo, estafador con línea ANTEL llamando a víctima con línea Claro y alertando a contacto en Movistar, y así sucesivamente). En cada escenario se verificó que la aplicación grabara correctamente la llamada, detectara patrones de estafa y enviara las alertas a los contactos de seguridad.

Estas pruebas son fundamentales para garantizar que la solución funcione de manera consistente en entornos reales, donde los usuarios pueden tener diferentes proveedores. También permiten identificar casos límite, como llamadas con número privado, que podrían afectar la detección o el registro de la llamada.

7.3 Calidad de Código

Para asegurar un código legible, mantenible y consistente a lo largo del desarrollo de SafeCall, se aplicaron principios de *Clean Code* [83], siguiendo buenas prácticas de nombrado y simplicidad en la lógica. Durante el desarrollo se utilizaron herramientas de asistencia y análisis automático para reforzar la calidad. Por ejemplo, GitHub Copilot [84] ayudó a revisar y sugerir correcciones en operaciones como *merges*, asegurando consistencia y reduciendo errores humanos.

El análisis de SonarQube [85] permitió monitorear la calidad del código de manera continua durante el desarrollo del prototipo. La herramienta identificó tres problemas de severidad alta asociados a una vulnerabilidad en la validación de certificados SSL/TLS en las conexiones HTTPS, clasificando la vulnerabilidad como crítica para la seguridad y privacidad. Si bien el objetivo del proyecto no se trata de una solución en producción, estas vulnerabilidades deberán corregirse antes de un despliegue productivo. El detalle de la vulnerabilidad se encuentra en el ANEXO 5.

La separación en múltiples repositorios permitió aislar responsabilidades por componente (*backend*, *frontend* y modelos de *machine learning*), facilitando la gestión independiente de cada módulo y la posibilidad de escalar o actualizar componentes sin afectar al resto del sistema. Cada repositorio estaba documentado y acompañado de diagramas de apoyo, lo que facilitó la comprensión y el mantenimiento del sistema.

Durante el desarrollo se aplicaron de manera transversal varias directrices del modelo 12 Factor App [86], asegurando una arquitectura flexible y fácil de mantener. Cada aplicación de SafeCall mantuvo su código principal (*codebase*) en un repositorio independiente, garantizando aislamiento de responsabilidades y trazabilidad. Las dependencias se declararon explícitamente mediante Gradle, el archivo requirements.txt de Python y configuraciones de Docker, asegurando entornos reproducibles. La

configuración del sistema se externalizó en archivos y *pipelines* de CI/CD, permitiendo modificar entornos sin alterar el código. Los *backing services*, como MongoDB en AWS y Redis, se conectaron como recursos intercambiables. El flujo de compilación, *release* y ejecución (*build, release, run*) permitió generar APKs de Android en Kotlin según el entorno (desarrollo, pruebas o producción) y desplegar versiones específicas por *release*. Los registros y eventos (*logs*) se gestionaron mediante Firebase Analytics [87], registrando información de ejecución, errores y uso de la aplicación, proporcionando trazabilidad y soporte para diagnóstico.

7.4 Gestión de la Configuración

El flujo de trabajo del proyecto se estructuró en múltiples repositorios de soporte, destacándose tres principales: el *frontend*, el *backend* y el proyecto de *machine learning*. Esta separación permitió mantener la independencia y la mantenibilidad de cada componente, facilitando el trabajo en paralelo. Para el control de versiones se siguió la estrategia GitFlow [88], con ramas dedicadas a producción (*main*), desarrollo (*develop*), nuevas funcionalidades (*feature*/nombre-de-funcionalidad) y corrección de errores (*bug*/nombre-de-bug), asegurando un flujo de trabajo ordenado y consistente. El *pipeline* de integración y despliegue continuos (CI/CD) realizado mediante GitHub Actions [89], automatiza la compilación, ejecución de pruebas, análisis de calidad mediante *linting* [90] y despliegue en un ambiente de *staging*. El uso de Docker garantiza consistencia entre entornos, evitando diferencias entre el código probado localmente y el desplegado en producción. Gracias a esta automatización, cada *merge request* puede estar disponible en producción en pocos minutos, reduciendo riesgos y acelerando los tiempos de entrega. Se pueden observar ejemplos en el ANEXO 5.4

8. Gestión del Proyecto

8.1. Introducción

La gestión del proyecto constituyó un aspecto central para coordinar el trabajo en equipo, garantizar el cumplimiento de los plazos académicos y orientar los esfuerzos hacia la entrega de un prototipo funcional. Dado que el proyecto presentaba un alto grado de incertidumbre inicial y la necesidad de adaptarse a cambios frecuentes en los requerimientos, se decidió adoptar un enfoque ágil.

En una primera etapa, durante el *Product Discovery*, se utilizó la metodología Kanban [91], que permitió gestionar tareas exploratorias con flexibilidad, en un contexto de incertidumbre temporal propio de la fase de investigación. Más adelante, y previo al inicio de la fase de desarrollo, el equipo migró a Scrum [92], ajustando la duración de los Sprints en función de la experiencia adquirida y de las necesidades de retroalimentación. Este capítulo describe la distribución de roles, la metodología adoptada, el manejo de historias de usuario, la planificación y el seguimiento del trabajo, la gestión de riesgos y los principales resultados obtenidos.

8.2. Distribución de Roles y Responsabilidades

La asignación de roles se realizó de manera evolutiva, considerando tanto las habilidades individuales como la afinidad de cada miembro con las distintas tareas. Esta dinámica se ajusta a los principios de autogestión de los equipos ágiles, en los que se prioriza la complementariedad de capacidades y la colaboración. Santiago Porro asumió el rol de *Product Owner*, aportando conocimiento sobre el dominio del mercado y gestión de equipos. Andrés Montaño se desempeñó como líder en inteligencia artificial y *machine learning*, además del desarrollo *frontend*. Por su parte, Suren Keushkerian tomó la responsabilidad de *Project Manager* y del desarrollo *backend*. Aunque estos roles se mantuvieron a lo largo del proyecto, existió flexibilidad para que cada miembro colaborara en otras áreas cuando fue necesario.

8.3. Metodologías de Gestión

En la siguiente línea de tiempo se muestran las etapas por las que paso el proyecto con los hitos más importantes en cada una de ellas.

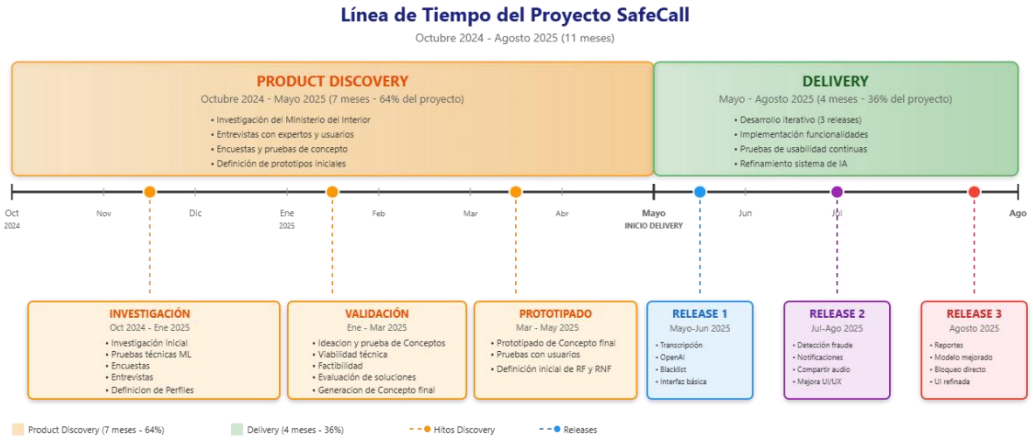


Ilustración 22 - Diagrama de hitos a lo largo del proyecto SafeCall

Durante la fase inicial de descubrimiento (*Product Discovery*) se aplicó Kanban, una metodología que permitió organizar y visualizar tareas exploratorias como pruebas técnicas de distintos modelos de *machine learning*, encuestas, entrevistas a usuarios y expertos, investigación de tecnologías y pruebas de concepto de posibles soluciones.

8.3.1 Kanban

Como parte de la metodología de desarrollo, la primera fase del proyecto se llevó a cabo bajo un enfoque Kanban, priorizando la flexibilidad y la adaptabilidad frente a la rigidez de procesos más estructurados. En esta etapa, se trabajó mediante historias de usuario definidas utilizando el formato propuesto por Cohn: “COMO [rol], QUIERO [funcionalidad], PARA [beneficio]” [93], que, si bien no contaban con criterios de aceptación exhaustivos, tenían como propósito principal agilizar la exploración, la investigación y el aprendizaje del equipo en un contexto experimental.

Investigación

Como se puede apreciar en el diagrama, las tareas desarrolladas durante esta fase estuvieron orientadas principalmente a la evaluación de la factibilidad técnica de la

solución propuesta. Entre ellas se incluyeron actividades como la investigación sobre los mecanismos de acceso a las llamadas telefónicas, el análisis de limitaciones de *hardware* y *software* (particularmente en distintas versiones de Android), y la exploración de modelos de lenguaje, disponibles en la plataforma Hugging Face. Dentro de este último eje, se indagó en la selección, uso y *fine-tuning* de modelos preentrenados, así como en la identificación y adaptación de *datasets* adecuados al contexto específico del proyecto.

Dado que el equipo no contaba con experiencia previa en *machine learning* ni en el trabajo con modelos de lenguaje, esta fase requirió una curva de aprendizaje inicial significativa, acompañada por la creación de diversos prototipos de validación técnica. Estos prototipos tuvieron como finalidad comprobar la viabilidad del acceso y procesamiento de llamadas, además de evaluar el rendimiento de distintos modelos de inferencia.

Validación y Prototipación

Con el objetivo de validar los supuestos técnicos y reducir la incertidumbre, se crearon prototipos experimentales que permitieron evaluar la factibilidad técnica del sistema. Algunos de estos prototipos involucraron el uso de dispositivos externos para la captura y escucha de llamadas, así como el desarrollo de aplicaciones de prueba destinadas a verificar el comportamiento del sistema bajo diferentes escenarios de *hardware* y versiones de Android. Estos experimentos resultaron fundamentales para comprender las restricciones del entorno móvil y definir los requerimientos mínimos para una futura implementación funcional.

Paralelamente, se desarrollaron tareas vinculadas a la investigación y comprensión del problema, incluyendo la realización de encuestas a usuarios, entrevistas con expertos y estudios de escritorio, en el marco de las etapas de empatía, definición, ideación, prototipación y pruebas, propias del proceso de *Product Discovery*.

Flujo de Trabajo y Control del WIP

A medida que se obtenían resultados tanto del lado técnico como del de investigación, se incorporaban nuevas tareas al *backlog*. Para gestionar este flujo, el equipo definió un límite de trabajo en progreso (WIP, por su sigla en inglés) [94] de seis puntos en total, equivalentes a seis tareas máximas en estado "*DOING*". Generalmente, cada integrante trabajaba en dos historias de usuario de manera simultánea, tomando nuevas tareas desde el *backlog* una vez que las anteriores se marcaban como completadas ("*DONE*"), en coordinación con el resto del equipo.

Resultados Obtenidos

La ausencia de restricciones temporales estrictas resultó inicialmente positiva, ya que permitió adaptarse a la naturaleza incierta y cambiante de las actividades propias de la exploración. Sin embargo, con el avance del proyecto, esta flexibilidad comenzó a generar dificultades en la planificación y en la estimación de esfuerzos, afectando la percepción de avance y extendiendo la duración de ciertas tareas más de lo esperado. Esta prolongación también se vio influida por factores adicionales, como esperar a contar con un volumen suficiente de datos de las encuestas antes de avanzar hacia fases posteriores, la complejidad para coordinar entrevistas con usuarios y expertos y la multiplicidad de actividades paralelas destinadas a reducir la incertidumbre técnica.

Al concluir el *Product Discovery* y el análisis de la factibilidad técnica de la solución, para pasar al *Delivery* de la solución, el equipo identificó la necesidad de contar con hitos más definidos, objetivos medibles y resultados tangibles por iteración, en lugar del flujo continuo característico de Kanban. Este aprendizaje motivó la adopción de la metodología Scrum, considerado más adecuado para la etapa de desarrollo, donde la planificación temporal, la entrega incremental y la evaluación continua del progreso se volvieron elementos muy importantes.

Además de ello, como resultados, el equipo logró validar los principales supuestos técnicos del proyecto. Se descartó la posibilidad de ejecutar el modelo de inferencia directamente en el dispositivo móvil, dado el alto requerimiento de recursos de procesamiento, optando en su lugar por una arquitectura con procesamiento externo. Al

mismo tiempo, se confirmó la viabilidad de detectar intentos de estafa mediante el ajuste fino de modelos de lenguaje, alcanzando resultados satisfactorios con los *datasets* seleccionados. Aunque no fue posible realizar el procesamiento en tiempo real durante la llamada, se comprobó que podía efectuarse inmediatamente después, manteniendo tiempos de respuesta adecuados. También se verificó que, a partir de Android 11, la aplicación podía configurarse como "Caller ID & Spam App", lo que habilitó el acceso a funciones clave como la identificación del número del remitente. Finalmente, de esta fase surgió un prototipo funcional que sirvió como base para la siguiente etapa de desarrollo bajo metodología Scrum, junto con decisiones técnicas relevantes como la adopción de los modelos DistilBERT y RoBERTa ajustados y la redefinición de ciertos requerimientos de producto.

8.3.2 Scrum

El desarrollo llevado adelante con la metodología de Scrum fue organizado en Sprints de dos semanas al inicio y posteriormente de una semana, lo que aumentó la frecuencia de retroalimentación y permitió agregar valor a la solución más rápidamente.

Daily Scrum

Si bien no se realizaron reuniones formales, el equipo mantuvo una comunicación continua a través de un grupo de WhatsApp, donde se realizaban intercambios breves dos o tres veces por semana. En estos espacios se comentaban los avances de cada integrante, se identificaban posibles bloqueos y se ofrecía apoyo mutuo para resolverlos.

Sprint Planning

El equipo mantenía una reunión semanal con el tutor del proyecto para presentar avances y resolver dudas, seguida de una planificación de Sprint al día siguiente. Las ceremonias de *Planning* se llevaban a cabo los viernes, luego de la reunión con el tutor del jueves. En estas instancias, el equipo revisaba las tareas completadas durante el Sprint en curso, analizaba el estado de cada una y, en base a ello, estimaba, priorizaba y seleccionaba las nuevas historias para incorporar al siguiente Sprint.

Retrospective

Se realizaron retrospectivas en determinados Sprints, registradas en Confluence [95], con el objetivo de reflexionar sobre el proceso y detectar oportunidades de mejora.

Flujo de Trabajo

En cuanto al flujo de trabajo, se utilizaron distintas herramientas de soporte: Jira para la gestión de tareas y la estimación mediante Historias de Usuario, Clockify [96] para el registro de horas invertidas, y Confluence para centralizar la documentación de retrospectivas. Al igual que en la etapa de Kanban, las tareas fueron gestionadas mediante historias de usuario redactadas en un formato estandarizado, lo que permitió reflejar claramente el contexto, la funcionalidad y la motivación de cada historia. Como ejemplo, se definió la siguiente historia: "COMO usuario registrado, QUIERO grabar automáticamente las llamadas entrantes, PARA poder conservar evidencia en caso de reclamo".

Al inicio, el equipo ya contaba con un *backlog* de historias proveniente de la etapa anterior de Kanban. A medida que avanzó el desarrollo y se realizaron pruebas con los usuarios, el backlog se amplió de forma continua con nuevas historias surgidas del *feedback* recibido, de las observaciones del tutor y de las ideas del equipo sobre posibles mejoras o funcionalidades a incorporar. Para los casos de mayor complejidad, los criterios de aceptación se formalizaron mediante el lenguaje Gherkin [97], lo que facilitó su validación. Un ejemplo de este tipo de criterio fue: "DADO que el usuario recibe una llamada, CUANDO se detecta una posible estafa, ENTONCES el sistema emite una alerta al usuario".

El equipo adoptó los *Story Points* [98] como unidad de estimación, estableciendo la equivalencia de un punto con una hora ideal de trabajo, entendida como tiempo efectivo sin interrupciones ni bloqueos técnicos. Además, se definieron criterios formales para determinar cuándo una historia estaba lista para comenzar (DoR - *Definition of Ready*) [99] y cuándo se consideraba completa (DoD - *Definition of Done*) [100]. Entre los principales criterios del DoR se incluyeron la redacción en el formato estándar, la definición de criterios de aceptación y la asignación de prioridades, mientras que el

DoD contemplaba la implementación de la funcionalidad según los criterios establecidos y la revisión de código mediante *pull requests*.

Gestión de los Sprints

La planificación y seguimiento de los Sprints se apoyó en el análisis de métricas ágiles.

La velocidad se utilizó como indicador del ritmo de entrega y permitió ajustar la planificación en función de la capacidad real del equipo.

Para el desarrollo se ejecutaron 22 Sprints (20/feb/2025 - 22/ago/2025). Los primeros cinco tuvieron una duración de 14 días; sin embargo, posteriormente se optó por acortar las iteraciones a 7 días con el objetivo de aumentar el ritmo de trabajo, mantener un seguimiento más cercano del progreso y poder corregir posibles desvíos con mayor agilidad.

Esta decisión estuvo motivada por la experiencia de la etapa anterior, en la que el equipo percibió la necesidad de establecer hitos que permitieran medir el progreso de manera más clara y mantener un mejor seguimiento del trabajo. Al pasar a Sprints tradicionales de dos semanas, se observó una mejora significativa en la organización del trabajo, la entrega de resultados y la detección temprana de desvíos. Esto impulsó al equipo a intensificar el ritmo y adoptar Sprints semanales, una dinámica que favoreció la retroalimentación constante, la planificación ajustada y la generación de valor de manera sostenida hasta el último *release* del proyecto.

Durante el transcurso de los Sprints, se ajustó la asignación de *Story Points* en función de la complejidad y el alcance de las historias planificadas, estableciéndose una *velocity* (ritmo promedio de trabajo del equipo) de aproximadamente 40 *Story Points* por Sprint.

La siguiente gráfica muestra los resultados de los últimos diez Sprints del proyecto, comparando los *Story Points* comprometidos al inicio de cada iteración (en gris) con los efectivamente completados al finalizarla (en verde).

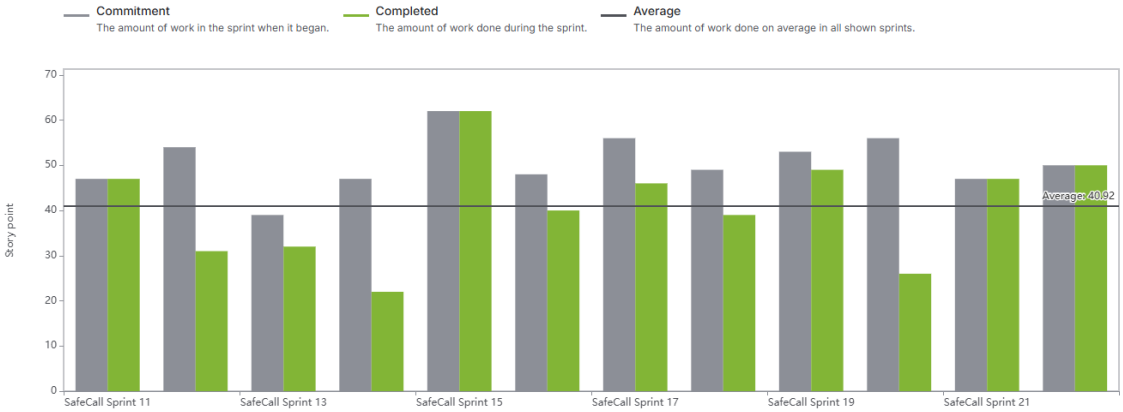


Ilustración 23 - Ilustración de Velocity de los últimos 12 Sprints

Puede observarse una tendencia general de estabilidad en la *velocity*, con algunas variaciones puntuales entre Sprints. En el Sprint 14 se registró una disminución en la cantidad de trabajo completado, principalmente debido a motivos personales de los integrantes del equipo y la carga asociada a otras materias que se estaban cursando. En contraste, el Sprint 15 mostró un pico de productividad, alcanzando un volumen de trabajo significativamente superior al promedio, resultado de una mejor organización interna y de la dedicación exclusiva durante ese período. Estas fluctuaciones reflejan los ajustes en la planificación y ejecución a medida que el equipo fue consolidando su dinámica de trabajo. Por mayor detalle de cada Sprint, dirigirse al ANEXO 6.2.

Durante cada Sprint se realizaba el seguimiento del *burndown chart*, una herramienta visual que representa el progreso del trabajo a lo largo del tiempo. Su función principal consiste en permitir al equipo monitorear si se encuentra en condiciones de completar todas las tareas comprometidas dentro del plazo establecido, comparando el trabajo efectivamente realizado con la línea de progreso ideal. Esta representación gráfica facilita la identificación temprana de desviaciones respecto al plan, como retrasos o adelantos en la ejecución de las tareas.

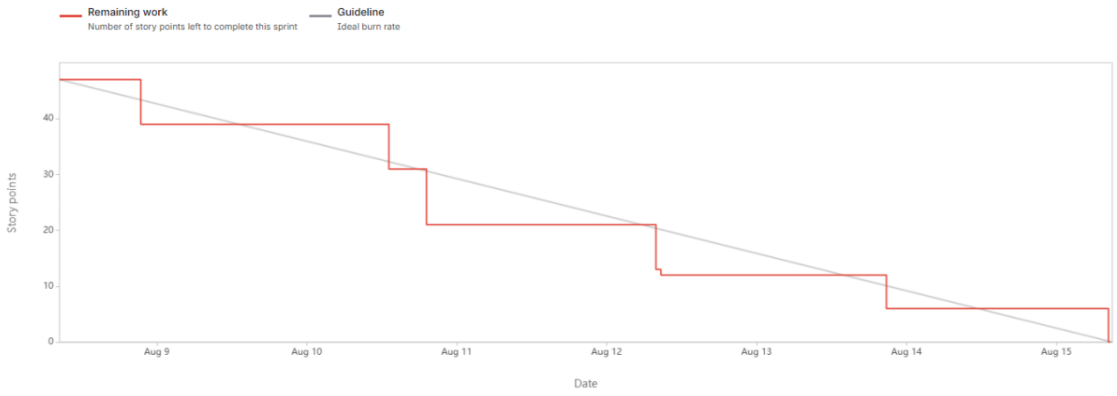


Ilustración 24 - Burndown Chart de uno de los últimos Sprints

Durante la etapa de *delivery* se realizaron cinco retrospectivas. Estas retrospectivas se desarrollaban siguiendo una dinámica breve y estructurada: cada integrante disponía de unos minutos para registrar sus observaciones en distintas columnas, clasificadas como "Lo bueno", "Lo malo", "Ideas" y "Acciones". Luego, los aportes se agrupaban por temas comunes, permitiendo identificar patrones y priorizar los aspectos más relevantes. Finalmente, el equipo acordaba acciones concretas para aplicar en el siguiente Sprint, asegurando que las mejoras propuestas se transformaran en compromisos prácticos y medibles. (ejemplos de retrospectivas en ANEXO 6).

Gestión de Releases

La estrategia de implementación se basó en *releases* planificados, donde cada entrega agregó funcionalidades específicas alineadas con los objetivos del proyecto y las necesidades identificadas tanto durante el *Product Discovery*, como en las pruebas de usuario realizadas tras cada *release*. Esta metodología permitió validar progresivamente la propuesta de valor y obtener retroalimentación temprana de usuarios reales.

Las funcionalidades que resolvían las necesidades principales del Perfil 1 (adultos mayores vulnerables) recibieron mayor prioridad, considerando que representan el grupo más desprotegido ante estafas telefónicas.

Además, se priorizaron funcionalidades que permitieran validar la arquitectura del sistema, especialmente la integración entre procesamiento de audio, transcripción y análisis de *machine learning*.

El desarrollo se organizó en tres *releases* principales con duraciones específicas:

- Release 1: 06/06/2025 - 03/07/2025 (4 semanas)
- Release 2: 04/07/2025 - 31/07/2025 (4 semanas)
- Release 3: 01/08/2025 - 29/08/2025 (4 semanas)

Aspecto	**Release 1**	**Release 2**	**Release 3**
Detección	Transcripción de Audio + Conexión a OpenAI	Detección de Fraude + Notificación de Fraude	Mejora del Modelo de Detección
Gestión de llamadas	Implementación de Blacklist + Detalle de llamadas	Reproducir y compartir grabación + Denuncia y bloqueo de número	Opción de bloqueo directo de llamadas desconocidas
Interfaz de usuario	Interfaz de usuario básica	Mejora de Interfaz de Usuario	Mejora de Interfaz de Usuario (continuación)
Notificaciones	Solo alertas locales en pantalla	Asignación y Notificación a Contactos	Sistema de notificaciones consolidado
Análisis y reportes	No implementado	No implementado	Reportes y estadísticas
Funciones adicionales	Navegación básica y pantallas core	Compartir grabaciones, denuncias, notas de versión	Funcionalidades analíticas avanzadas

Tabla 14 - Evolución de Funcionalidades por Release

Resultados por Release

Release 1: Fundamentos y Validación Técnica (06/06/2025 - 03/07/2025)

El primer *release* se centró en validar la viabilidad técnica del sistema mediante la implementación de componentes principales. Se logró establecer la comunicación entre *frontend* y *backend*, implementar transcripción de audio con OpenAI Whisper y desarrollar la interfaz básica de usuario. Se integró un sistema que analiza grabaciones de llamadas utilizando el modelo de transcripción Whisper y la API de OpenAI para detectar patrones de fraude, complementado con un servicio de detección basado en una lista de números sospechosos que alerta al usuario. La implementación de *blacklist* permitió bloquear llamadas de números específicos, mientras que las pantallas de navegación básica y visualización de detalles de llamadas establecieron la estructura de interacción principal.

Las pruebas de usabilidad realizadas con cinco usuarios (perfiles de 19 a 80 años, niveles tecnológicos bajo a medio) brindaron información importante. Los usuarios manifestaron la necesidad de agregar números desconocidos automáticamente a listas de sospechosos, implementar un sistema de denuncias que escale según frecuencia (con bloqueo automático tras 2-3 denuncias), y mejorar la visibilidad de alertas mediante letras grandes y repetición de audio. Las pruebas también identificaron la necesidad de guías post-alerta que orienten al usuario sobre acciones a tomar, opciones para reproducir y compartir grabaciones, y un buscador en la interfaz de contactos.

El principal aprendizaje técnico fue la identificación de limitaciones en la grabación de llamadas en dispositivos no-Samsung.

Release 2: Detección y Notificaciones (04/07/2025 - 31/07/2025)

El segundo *release* incorporó el objetivo central del Prototipo Funcional: detección y notificación automática ante llamadas fraudulentas. Se implementó la asignación de contactos de seguridad utilizando la agenda del teléfono con barra de búsqueda, y el sistema de notificación por SMS que envía alertas automáticas a contactos designados incluyendo el número sospechoso y señal de alerta. La funcionalidad de escuchar y compartir grabaciones de llamadas surgió como respuesta directa al *feedback* de

usuarios durante las pruebas del Release 1, junto con la opción de denunciar y bloquear números desde el detalle de llamada. Adicionalmente, se incorporaron notas de versión en la aplicación para informar sobre cambios y mejoras.

Las pruebas con cinco usuarios (19 a 80 años, niveles tecnológicos bajo a avanzado) evaluaron escenarios específicos de configuración y notificación. La configuración de contactos fue bien recibida, destacándose la utilidad de la barra de búsqueda, aunque se identificó como principal obstáculo la solicitud de permisos de acceso a la agenda que confundió a usuarios de bajo nivel tecnológico. Las notificaciones automáticas generaron alta aceptación, con usuarios valorando la tranquilidad de que familiares reciban avisos. La funcionalidad de reproducir y compartir grabaciones fue especialmente valorada por usuarios que buscan compartir pruebas con terceros, sugiriéndose un botón más visible en el listado de grabaciones y opción para editar audio antes de compartir.

Release 3: Optimización y Funcionalidades Avanzadas (01/08/2025 - 29/08/2025)

El tercer *release* se enfocó en optimizar la experiencia de usuario y agregar funcionalidades analíticas. Se implementó el sistema de reportes que permite evaluar el funcionamiento del sistema mediante métricas de efectividad, y se mejoró el tiempo de análisis a unos 12 segundos promedio. El bloqueo directo de llamadas desconocidas (“Guardian”) respondió a una necesidad identificada durante validaciones con el Perfil 2, mientras que las mejoras de interfaz se basaron en observaciones acumuladas de *releases* anteriores, priorizando simplicidad y claridad visual para adultos mayores.

Las funcionalidades analíticas avanzadas incluyeron *dashboards* de métricas que permiten visualizar tasas de detección, tiempo de respuesta del sistema y patrones de llamadas sospechosas. Estas herramientas facilitan tanto la evaluación académica del proyecto como la identificación de oportunidades de mejora en el algoritmo de detección.

8.6. Identificación de Riesgos

La identificación de riesgos del proyecto se desarrolló de forma evolutiva a lo largo del desarrollo. Al inicio, durante la fase de *Product Discovery*, se identificaron principalmente riesgos de gestión de proyecto relacionados con la coordinación del equipo, el acceso a tecnologías específicas (como la grabación de llamadas) y la viabilidad técnica de la detección de fraudes.

A medida que avanzamos en el estudio de la problemática y en las validaciones con usuarios reales, surgieron nuevos riesgos asociados a los atributos de calidad priorizados para el Prototipo Funcional: usabilidad (especialmente crítica para adultos mayores), *performance* (tiempos de procesamiento y precisión del modelo), portabilidad (limitaciones por fabricante de dispositivo) y deployabilidad (consistencia entre entornos).

Si bien no se implementó una gestión de riesgos formal, esta identificación temprana sirvió como guía para anticipar obstáculos y tomar decisiones más informadas. Dado que el equipo trabajó con objetivos concretos y un alcance claramente definido, este enfoque permitió mantener el control sobre aspectos relevantes del proyecto y reaccionar a tiempo ante eventualidades.

Riesgos Basados en Atributos de Calidad

Usabilidad

Los principales riesgos identificados se relacionan con la adecuación de la interfaz a los usuarios objetivo. Entre ellos, se destacan la complejidad visual y cognitiva para adultos mayores, la confusión en el flujo de configuración inicial, el uso de terminología técnica poco comprensible, el tamaño inadecuado de texto e íconos, la baja visibilidad de notificaciones en situaciones de estrés y el exceso de información en pantalla.

Para mitigar los riesgos de usabilidad, se aplicaron las heurísticas de Nielsen para garantizar una interfaz clara y accesible, priorizando elementos visuales de alto contraste, tipografías claras y grandes y navegación intuitiva. El diseño minimalista y las validaciones continuas con adultos mayores permitieron mejorar progresivamente la

experiencia de uso, logrando mejoras significativas en la tasa de completitud de tareas y satisfacción del usuario a lo largo de los *releases*.

Performance

Los riesgos más relevantes incluyen tiempos de análisis post-llamada elevados, precisión insuficiente del modelo de detección, uso excesivo de memoria o almacenamiento en el dispositivo, degradación del rendimiento con audios extensos, y la posibilidad de falsos positivos o falsos negativos que afecten la confianza y efectividad del sistema.

Para mitigar los riegos de *performance*, se optimizó el modelo de *machine learning* y se utilizó un *dataset* especializado en estafas telefónicas para mejorar la precisión de detección. Por otro lado, se implementó procesamiento por segmentos lo que permitió optimizar el análisis de audios de diferentes duraciones. Estas acciones permitieron cumplir con los objetivos de tiempo de análisis (≤30s) y precisión (≥80%).

Portabilidad

Los riesgos principales están vinculados a la dependencia de dispositivos y versiones específicas de Android. Se identificaron posibles incompatibilidades con otros fabricantes, limitaciones derivadas de la versión mínima del sistema operativo y la exposición ante actualizaciones constantes del entorno Android.

Los riesgos de portabilidad fueron identificados, evaluados y conscientemente aceptados tras validación técnica con múltiples dispositivos. Se documentó la limitación a dispositivos Samsung con Android 11+ como una restricción inherente a las APIs de grabación disponibles.

Deployabilidad

Los riesgos clave abarcan los tiempos prolongados de despliegue, inconsistencias entre entornos, complejidades en la gestión de dependencias y configuraciones, costos operativos elevados y la falta inicial de observabilidad del sistema en producción.

Para garantizar la deployabilidad, se utilizó Docker, asegurando consistencia entre entornos. Además, se implementó un *pipeline* CI/CD automatizado que reduce tiempos de despliegue y minimiza errores humanos.

Riesgos de Gestión de Proyecto

Los principales riesgos de gestión identificados se relacionan con la organización del equipo y la disponibilidad de recursos. Se destacaron la falta de compromiso o coordinación entre miembros, la dificultad de acceso a llamadas reales y los desafíos en la detección de fraudes durante el desarrollo.

También se registraron riesgos de carácter medio vinculados a la comunicación interna, la falta de experiencia en *machine learning*, la complejidad del entrenamiento del modelo, la dificultad para establecer plazos realistas, el registro preciso de horas trabajadas y la carga académica adicional fuera del proyecto.

Por último, se identificaron riesgos menores como las diferencias de horario entre integrantes y el posible desinterés del público objetivo hacia el prototipo desarrollado.

Los riesgos de gestión de proyecto se abordaron principalmente desde una perspectiva preventiva, identificando posibles problemas que podrían surgir durante el desarrollo.

Para mayor detalle sobre la identificación de riesgos, dirigirse al ANEXO 6

9. Conclusiones

9.1. Cumplimiento de Objetivos

SafeCall alcanzó sus objetivos principales al desarrollar un prototipo funcional capaz de detectar llamadas fraudulentas mediante el análisis del contenido de la comunicación, notificando tanto al usuario como a los contactos designados. El proyecto se centró en patrones de estafa relevantes localmente, como el "cuento del tío" y el *vishing*, logrando detecciones en un tiempo promedio inferior a 30 segundos.

Durante el desarrollo, el equipo adquirió experiencia práctica en herramientas de inteligencia artificial y procesamiento de lenguaje natural, aplicando conceptos de *machine learning* sin experiencia previa, lo que fortaleció sus competencias técnicas y académicas.

La gestión del proyecto combinó metodologías ágiles: Kanban en la fase de descubrimiento y Scrum en la fase de *delivery*, con ceremonias regulares e integración continua, lo que permitió cumplir los plazos y los hitos previstos.

9.2. Contribuciones del Proyecto

Técnicamente, el proyecto significó un crecimiento notable para el equipo, que adquirió conocimientos en Python, Kotlin, ajuste fino de modelos de IA y el uso de asistentes de programación como Copilot. Este proceso permitió salir de la zona de confort y aplicar conocimientos en nuevas situaciones. Aunque la aplicación aún no se encuentra en producción, SafeCall aborda un problema relevante y creciente: las estafas telefónicas. Su implementación efectiva podría contribuir significativamente a la protección de los usuarios y a la concientización sobre este tipo de fraudes. Atacamos un problema actual y real que las grandes empresas tienen en la mira

9.3. Lecciones Aprendidas

La principal lección aprendida se relaciona con la gestión y la resiliencia frente a los distintos desafíos presentados durante el desarrollo del proyecto. Más allá de las competencias técnicas, el equipo fortaleció habilidades en planificación, toma de

decisiones, adaptación a cambios y ajustes sobre la marcha, así como la constancia y el compromiso semanal necesarios para aportar valor de manera sostenida.

En las etapas iniciales, el uso de la metodología Kanban, sin fechas fijas ni objetivos temporales concretos, dificultó la organización del trabajo y la percepción real del avance, lo que llevó a extender algunas tareas más de lo previsto. Sin embargo, al incorporar prácticas de Scrum y establecer Sprints con plazos definidos, el equipo logró mejorar la coordinación, el ritmo de trabajo y la eficiencia en la entrega de resultados. Este aprendizaje marcó un punto de inflexión en la gestión del proyecto

El proyecto comenzó con un enfoque emprendedor, buscando desarrollar una solución innovadora con potencial impacto social. Con el tiempo, el interés y la energía del equipo se fueron concentrando gradualmente solo en el ámbito académico, permitiendo consolidar conocimientos y habilidades valiosas.

Otra lección aprendida estuvo vinculada a la gestión de riesgos. Si bien durante el proyecto se identificaron y consideraron algunos riesgos de manera informal, como por ejemplo dificultades técnicas, no se implementó una gestión de riesgos formal ni se elaboraron planes de mitigación y contingencia estructurados. A medida que el desarrollo avanzó y algunos de esos riesgos se materializaron, como las sucesivas actualizaciones de Android, que obligaron a revisar partes del código para asegurar su compatibilidad, se hizo evidente la necesidad de contar con una gestión más rigurosa que permitiera anticipar este tipo de contingencias y definir planes de acción. En proyectos con un alto nivel de incertidumbre e innovación como este, una gestión de riesgos más sistemática habría favorecido una mejor toma de decisiones, mayor previsibilidad y una respuesta más ágil ante imprevistos.

9.4 Desafíos del Proyecto

El proyecto enfrentó un problema complejo y de alta incertidumbre. Desde el inicio, el equipo se encontró con desafíos técnicos importantes, como optimizar modelos de IA para dispositivos con recursos limitados, integrarlos con distintos sistemas de telefonía y garantizar compatibilidad entre versiones de Android.

Además, el manejo de datos sensibles exigió considerar aspectos legales y de privacidad, condicionando la forma en que se podían procesar y almacenar los datos.

A pesar de estas dificultades, el equipo logró avanzar progresivamente: mediante entrevistas con expertos se profundizó en el problema, se adquirieron y aplicaron conocimientos sobre modelos de IA sin experiencia previa, y se implementaron soluciones que fueron mejorando con cada iteración. Este proceso permitió no solo superar los obstáculos iniciales, sino también obtener resultados, consolidando capacidades técnicas y prácticas valiosas para enfrentar problemas complejos en contextos reales.

9.5 Trabajo Futuro

Durante el desarrollo de la aplicación se identificaron diversas oportunidades de mejora que podrían abordarse en futuras iteraciones. Una de ellas consiste en lograr una integración más profunda con el sistema de llamadas del dispositivo, trasladar el modelo de inteligencia artificial al móvil y optimizar el consumo de memoria, batería y latencia, lo que permitiría mejorar la experiencia de uso. Asimismo, se proyecta avanzar hacia la detección de llamadas sospechosas en tiempo real, anticipando la advertencia al usuario antes de que la llamada finalice, y completar la integración del sistema de autenticación para reforzar la seguridad de la aplicación. En paralelo, ampliar la compatibilidad con distintos fabricantes y versiones de Android, e incluso explorar sistemas operativos como iOS, contribuiría a incrementar el alcance y la adopción de la herramienta.

En términos de funcionalidad, la aplicación podría incorporar notificaciones más avanzadas y versátiles, incluyendo alertas *push* enriquecidas, asistentes virtuales o señales hápticas, con el objetivo de ofrecer una experiencia más intuitiva y efectiva. También se podrían implementar mecanismos de aprendizaje continuo a partir de las detecciones realizadas, así como análisis contextual de las interacciones y patrones emocionales en la comunicación, proporcionando retroalimentación que fortalezca la conciencia del usuario sobre la comunicación y su seguridad.

De manera complementaria, SafeCall podría evolucionar hacia un asistente activo, capaz de gestionar llamadas según pautas predefinidas, responder con mensajes personalizados y coordinarse con servicios de calendario para administrar la disponibilidad, priorizar contactos y generar reportes de interacción. Estas mejoras permitirían consolidar un ecosistema de comunicación inteligente, combinando seguridad, productividad y asistencia proactiva, aumentando significativamente la utilidad y el impacto de la aplicación.

10. Referencias bibliográficas

[1] "Types of Prototypes: A Complete Overview," Product School. [Online]. Disponible:https://productschool.com/blog/product-strategy/types-prototypes. Accedido en: Oct. 11, 2025.

[2] "Desirability, Viability, and Feasibility: The Three Pillars of Design Thinking," UX Design Institute. [Online]. Disponible: https://www.uxdesigninstitute.com/blog/desirability-viability-and-feasibility. Accedido en: Oct. 11, 2025

[3] "Get Started," Asterisk. [Online]. Disponible: https://www.asterisk.org/get-started/. Accedido en: Oct. 11, 2025

[4] CERES, "Auge de estafas virtuales en Uruguay," Montevideo, Uruguay, Julio 2025. [Online]. Disponible: https://ceres.uy/index.php/estudios/ficha_estudio/226 Accedido en: Sep. 2, 2025.

[5] Ministerio del Interior, Área de Estadística y Criminología Aplicada (AECA), "Estafas en Uruguay: Evolución, Características y Desafíos Actuales," Informe Temático IT/2025/01, Montevideo, Uruguay, julio 2025. [Online]. Disponible: https://www.gub.uy/ministerio-interior/sites/ministerio-interior/files/documentos/publicaciones/AECA%202025%20Informe%20tem%C3%A1tico%20sobre%20Estafas%20IT-2025-01%20vF.pdf. Accedido en: Sep. 2, 2025.

[6] República Oriental del Uruguay, "Ley N° 20.327 - Díctanse normas para la prevención y represión de la ciberdelincuencia," IMPO, Montevideo, Uruguay, 23 Ago. 2024. [Online]. Disponible:https://www.impo.com.uy/bases/leyes-originales/20327-2024. Accedido en: Nov. 5, 2024.

[7] Ámbito, "El BROU logró recuperar el 63% de los montos perdidos por estafas," Ámbito, Uruguay, Apr. 26, 2023. [Online] Disponible: https://www.ambito.com/uruguay/el-brou-logro-recuperar-el-63-los-montos-perdidos-estafas-n5707611. Accedido en: Oct. 25, 2024.

[8] Santander, "Vishing: una llamada con mucha trampa," Santander Stories. [Online]. Disponible: https://www.santander.com/es/stories/vishing-una-llamada-con-mucha-trampa. Accedido en: Nov. 3, 2024.

[9] BBVA, "Phishing, Vishing, Smishing, ¿qué son y cómo protegerse?," BBVA, Dec. 4, 2024. [Online]. Disponible: https://www.bbva.com/es/innovacion/phishing-vishing-smishing-que-son-y-como-protegerse-de-estas-amenazas/. Accedido en: Nov. 3, 2024.

[10] Ministerio Público Fiscal de la Provincia de Salta, "Crecen las estafas con el 'Cuento del Tío': recomendaciones para no ser víctima," MPF Salta. [Online]. Disponible: https://www.fiscalespenalesalta.gob.ar/crecen-las-estafas-con-el-cuento-del-tio-recomendaciones-para-no-ser-victima/. Accedido en: Oct. 26, 2024.

[11] Portal Montevideo, "Intentaron estafar a una mujer recreando la voz de su hermano con inteligencia artificial," Montevideo.com.uy, Ago. 2024. [Online]. Disponible: https://www.montevideo.com.uy/Noticias/Intentaron-estafar-a-una-mujer-recreando-la-voz-de-su-hermano-con-inteligencia-artificial-uc908330 Accedido en: Oct. 23, 2024.

[12] "OSE Advierte Sobre Nueva Modalidad de Estafa Digital: Correos Falsos Simulan Errores de Facturación," La Colonia Portal. [Online]. Disponible: https://lacoloniaportal.com.uy/ose-advierte-sobre-nueva-modalidad-de-estafa-digital-correos-falsos-simulan-errores-de-facturacion/. Accedido en: Sept. 2, 2025.

[13] UTE, "Nueva Advertencia a los Clientes por Modalidad de Estafa Digital," Comunicados UTE. [Online]. Disponible: https://www.ute.com.uy/comunicados/nueva-advertencia-los-clientes-por-modalidad-de-estafa-digital. Accedido en: Sept. 2, 2025.

[14] Banco Santander, "Qué es el vishing: protégete de la estafa telefónica," Banco Santander. [Online]. Disponible: https://www.bancosantander.es/glosario/vishing. Accedido en: Nov. 3, 2024.

[15] Comisión Federal de Comercio de Estados Unidos, "Con pérdidas por fraude en todo el país que en 2023 superaron los $10,000 millones de dólares, la FTC intensifica sus esfuerzos para proteger al público," FTC, Feb. 9, 2024. [Online]. Disponible: https://www.ftc.gov/es/noticias/con-perdidas-por-fraude-en-todo-el-pais-que-en-2023-superaron-los-10000-millones-de-dolares-la-ftc. Accedido en: Sep. 6, 2025.

[16] "BPS Alerta Sobre Nueva Modalidad de Estafa Mediante Sitios Web que Mencionan al Fonasa y Simulan Devoluciones," El Observador, Nacional. [Online]. Disponible: https://www.elobservador.com.uy/nacional/bps-alerta-nueva-modalidad-estafa-sitios-web-que-mencionan-al-fonasa-y-simulan-devoluciones-n6013799. Accedido en: Sept. 2, 2025.

[17] El País Uruguay, "La Policía en alerta ante aumento de estafas: una a una, cuáles son las principales modalidades en Uruguay," EL PAÍS Uruguay. [Online]. Disponible: https://www.elpais.com.uy/que-pasa/una-a-una-las-principales-estafas-en-uruguay-desde-el-cuento-del-tio-a-la-romantica-y-enganos-por-facebook. Accedido en: Nov. 13, 2024.

[18] Los Andes, "Abundan las estafas a través de phishing, vishing y smshing: qué son y cómo evitarlas," Los Andes, Nov. 24, 2022. [Online]. Disponible: https://www.losandes.com.ar/sociedad/abundan-las-estafas-a-traves-de-phishing-vishing-y-smshing-que-son-y-como-evitarlas. Accedido en: Sep. 6, 2025.

[19] El Observador, "El Banco República recibió 2.653 denuncias por estafa y frenó el 35%: ¿cuánto dinero se perdió?," El Observador, Montevideo, Uruguay, Apr. 3, 2024. [Online]. Disponible:https://www.elobservador.com.uy/nota/el-banco-republica-recibio-2-653-denuncias-por-estafa-y-freno-el-35-cuanto-dinero-perdio--20244212530. Accedido en: Nov. 10, 2024

[20] Ámbito, "El BROU recibió 40% menos de denuncias por fraudes y estafas en 2023," Ámbito, Uruguay, Jan. 30, 2024. [Online]. Disponible: https://www.ambito.com/uruguay/el-brou-recibio-40-menos-denuncias-fraudes-y-estafas-2023-n5933501. Accedido en: Nov. 10, 2024.

[21] BioCatch, "Comprender y abordar el impacto emocional del fraude financiero," BioCatch Blog, Jun. 18, 2024. [Online]. Disponible:https://www.biocatch.com/es/blog/abordar-el-impacto-emocional-del-fraude-financiero. Accedido en: Nov. 28, 2025.

[22] Perito Judicial, "Estafa Emocional: Cómo reclamar el dinero perdido," Perito Judicial, Feb. 11, 2025. [Online]. Disponible: https://peritojudicial.com/estafa-emocional/. Accedido en: Sep. 21, 2025.

[23] Top Doctors, "Consecuencias emocionales de una estafa," Top Doctors, Aug. 1, 2024. [Online]. Disponible:https://www.topdoctors.es/articulos-medicos/consecuencias-emocionales-de-una-estafa-estres-postraumatico/. Accedido en: Sep. 21, 2025.

[24] El Observador, "El Banco República recibió 2.653 denuncias por estafa y frenó el 35%: ¿cuánto dinero se perdió?," El Observador, Montevideo, Uruguay, Apr. 3, 2024. [Online]. Disponible:https://www.elobservador.com.uy/nota/el-banco-republica-recibio-2-653-denuncias-por-estafa-y-freno-el-35-cuanto-dinero-perdio--20244212530. Accedido en: Sep. 3, 2025.

[25] IDEO, "Design Thinking Defined," IDEO U. [Online]. Disponible: https://www.ideou.com/blogs/inspiration/what-is-design-thinking. Accedido en: Oct. 7, 2025.

[26] d.school Stanford, "Design Thinking Bootleg," Stanford University, Hasso Plattner Institute of Design. [Online]. Disponible:https://dschool.stanford.edu/tools/design-thinking-bootleg. Accedido en: Sep. 2, 2025.

[27] Miro, "¿Qué es Miro?," Miro Help Center. [Online]. Disponible: https://help.miro.com/hc/es/articles/360017730533--Qu%C3%A9-es-Miro. Accedido en: Oct. 7, 2025.

[28] Miro, "¿Qué es user persona?," Miro. [Online]. Disponible: https://miro.com/es/persona/que-es-user-persona/ .Accedido en: Oct. 7, 2025.

[29] Leanscape, "An Overview of the Pugh Matrix: What Is It and How Does It Work?," Leanscape. [Online]. Disponible: https://leanscape.io/an-overview-of-the-pugh-matrix-what-is-it-and-how-does-it-work/. Accedido en: Abr. 7, 2025.

[30] Proto.io, "Proto.io - Prototyping Platform for Web and Mobile Apps," [Online]. Disponible: https://proto.io/. Accedido en: Sep. 24, 2025.

[31] Nielsen Norman Group, "Wizard of Oz: A Method for Testing with Users," Nielsen Norman Group. [Online]. Disponible: https://www.nngroup.com/articles/wizard-of-oz/. Accedido en: Oct. 7, 2025.

[32] Nielsen Norman Group, "Thinking Aloud: The #1 Usability Tool," [Online]. Disponible: https://www.nngroup.com/articles/thinking-aloud-the-1-usability-tool/. Accedido en: Sep. 24, 2025.

[33] Nielsen Norman Group, "Task Scenarios for Usability Testing," Nielsen Norman Group. [Online]. Disponible: https://www.nngroup.com/articles/task-scenarios-usability-testing/ Accedido en: Oct. 7, 2025.

[34] Nielsen Norman Group, "Observer Guidelines for Usability Testing," Nielsen Norman Group. [Online]. Disponible: https://www.nngroup.com/articles/observer-guidelines/. Accedido en: Oct. 7, 2025.

[35] Meta Platforms Inc., "Facebook Marketplace Montevideo," Facebook. [Online]. Disponible: https://www.facebook.com/marketplace/montevideo/. Accedido en: Oct. 7, 2025.

[36] Meta Platforms Inc., "WhatsApp," WhatsApp. [Online]. Disponible: https://www.whatsapp.com/. Accedido en: Oct. 7, 2025.

[37] República Oriental del Uruguay, "Ley N° 18.331 - Protección de Datos Personales y Acción de Habeas Data," IMPO, Montevideo, Uruguay, 11 Ago. 2008. [Online]. Disponible: https://www.impo.com.uy/bases/leyes/18331-2008. Accedido en: Oct. 7, 2024.

[38] Unidad Reguladora y de Control de Datos Personales (URCDP), "Guía General de Protección de Datos Personales en Uruguay," URCDP, Montevideo, Uruguay. [Online]. Disponible: https://www.gub.uy/unidad-reguladora-control-datos-personales/comunicacion/publicaciones/guia-general-proteccion-datos-personales-uruguay. Accedido en: Sep. 14, 2024.

[39] Google LLC, "Android 11," Android Developers. [Online].Disponible:https://developer.android.com/about/versions/11?hl=es-419. Accedido en: Sep. 29, 2025.

[40] Sinch AB, "Sinch Platform Overview,"[Online].Disponible:https://www.sinch.com/products/voice/. Accedido en: Sep. 22, 2025.

[41] Sinch, "Voice API Documentation," [Online]. Disponible: https://developers.sinch.com/docs/voice/. Accedido en: Sep. 22, 2025.

[42] Truecaller AB, "Truecaller: Caller ID & Block," [Online]. Disponible: https://www.truecaller.com/. Accedido en: Sep. 22, 2025.

[43] RecorderGear, "PR200 Cell Phone Call Recorder," RecorderGear. [Online]. Disponible: https://recordergear.com/products/pr200-cell-phone-call-recorder Accedido en: Oct. 7, 2025.

[44] IT Pro Uruguay, "UGREEN Adaptador Bluetooth 5.0 a USB-C Switch PS," IT Pro. [Online]. Disponible: https://itpro.com.uy/product/ugreen-adaptador-bluetooth-5-0-a-usb-c-switch-ps/. Accedido en: Oct. 7, 2025.

[45] Hiya Inc., "Hiya: Caller ID & Block," [Online]. Disponible: https://hiya.com/. Accedido en: Sep. 22, 2025.

[46] RoboKiller, "RoboKiller: Stop Spam Calls," [Online]. Disponible: https://www.robokiller.com/. Accedido en: Sep. 22, 2025.

[47] Google LLC, "Call Blocker - Bloqueador de llamadas," Google Play Store. [Online]. Disponible:https://play.google.com/store/apps/details?id=com.unknownphone.callblocker. Accedido en: Oct. 7, 2025.

[48] TrapCall, "Mr. Number: Caller ID & Spam Protection," [Online]. Disponible: https://www.mrnumber.com/. Accedido en: Sep. 22, 2025.

[49] F. Romeling, "A guide to develop your AI-MVP," Medium, Apr. 27, 2023. [Online]. Disponible: https://medium.com/@filip_70113/a-guide-to-develop-your-ai-mvp-36f8e204e1cb. Accedido en: Oct. 4, 2025.

[50] D. Nüst, V. Sochat, B. Marwick, S. J. Eglen, T. Head, and T. Hirst, "Ten simple rules for writing Dockerfiles for reproducible data science," PLoS Comput. Biol., vol. 16, no. 11, e1008316, Nov. 2020. [Online]. Disponible: https://doi.org/10.1371/journal.pcbi.1008316. Accedido en: Oct. 5, 2025.

[51] Atlassian, "Continuous integration vs. delivery vs. deployment," [Online]. Disponible:https://www.atlassian.com/continuous-delivery/principles/continuous-integration-vs-delivery-vs-deployment. Accedido en: Oct. 5, 2025.

[52] IBM, "What is Artificial Intelligence (AI)?", IBM, 2023. [Online]. Disponible: https://www.ibm.com/topics/artificial-intelligence . Accedido en: Jun. 15, 2025.

[53] D. Jurafsky and J. H. Martin, Speech and Language Processing, 3rd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2023.

[54] Hugging Face, “The Annotated Transformer and the Rise of Modern NLP,” Hugging Face Documentation, 2024. [Online]. Disponible:https://huggingface.co/transformers . Accedido en Mar. 8, 2025.

[55] IBM, “What Is Transfer Learning?”, IBM THINK Topics, 1.7 años atrás. [Online]. Disponible: https://www.ibm.com/think/topics/transfer-learning. Accedido en: Ago. 11, 2025.

[56] Google Research, “BERT: Pre-trained Language Model,” Hugging Face Models, 2024. [Online]. Disponible: https://huggingface.co/bert-base-uncased. Accedido en: Set 21, 2024.

[57] Hugging Face, “DistilBERT: A Distilled Version of BERT,” Hugging Face Models, 2024. [Online]. Disponible: https://huggingface.co/distilbert-base-uncased. Accedido en: Set 21, 2024.

[58] Meta AI, “RoBERTa: A Robustly Optimized BERT Pretraining Approach,” Hugging Face Models, 2024. [Online]. Disponible: https://huggingface.co/roberta-base. Accedido en: Oct. 15, 2024.

[59] Facebook AI, “XLM-RoBERTa: Cross-Lingual Language Model,” Hugging Face Models, 2024. [Online]. Disponible: https://huggingface.co/xlm-roberta-base. Accedido en: Oct. 15, 2024.

[60] Kaggle, “Call Transcripts Scam Determinations,” *Kaggle*, 2024. [Online]. Disponible: https://www.kaggle.com/datasets/mealss/call-transcripts-scam-determinations. Accedido en: Feb. 12, 2025.

[61] Hugging Face, “Transformers and Model Hub,” Hugging Face Documentation, 2024. [Online]. Disponible: https://huggingface.co. Accedido en: Oct. 8, 2024.

[62] J. Devlin, M. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” Proceedings of NAACL-HLT, pp. 4171–4186, 2019.

[63] D. Jurafsky and J. H. Martin, Speech and Language Processing, 3rd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2023.

[64] Hugging Face, “TinyBERT: A Compact BERT Model for NLP Tasks,” Hugging Face Models, 2024. [Online]. Disponible: https://huggingface.co/huawei-noah/TinyBERT_General_4L_312D. Accedido en: Oct. 10, 2025.

[65] Hugging Face, “MobileBERT: A Compact Task-Agnostic BERT for NLP Tasks,” Hugging Face Models, 2024. [Online]. Disponible: https://huggingface.co/google/mobilebert-uncased. Accedido en: Oct. 10, 2025.

[66] Zilliz, “DistilBERT: A Distilled Version of BERT,” 2024. [Online]. Disponible: https://zilliz.com/learn/distilbert-distilled-version-of-bert. Accedido en Mar. 11, 2025.

[67] OpenAI, “GPT-3.5 Turbo,” OpenAI API Documentation, 2024. [Online]. Disponible: https://platform.openai.com/docs/models/gpt-3-5. Accedido en: Jun 10, 2025.

[68] Google LLC, Mountain View, CA, "Firebase - App Development Platform," [Online]. Disponible: https://firebase.google.com/. Accedido en: Oct. 9, 2025.

[69] OpenAI, San Francisco, CA, "Introducing Whisper," [Online]. Disponible: https://openai.com/index/whisper/. Accedido en: Oct 9, 2025.

[70] MongoDB Inc., "Documents - MongoDB Manual," MongoDB Inc., 2024. [Online]. Disponible: https://www.mongodb.com/docs/manual/core/document/. Accedido en: Jun. 10, 2025.

[71] Grafana Labs, "Grafana: The open observability platform," Grafana Labs, New York, NY, USA, 2024. [Online]. Disponible: https://grafana.com/. Accedido en: Jul. 14, 2025.

[72] Redis Ltd., "Redis - In-Memory Database for Caching and Real-Time Applications," [Online]. Disponible: https://redis.io/about/. Accedido en: Jun. 9, 2025.

[73] Amazon Web Services, "Amazon EC2 User Guide," 2024. [Online]. Disponible: https://docs.aws.amazon.com/ec2/. Accedido en: Sep. 30, 2025.

[74] Amazon Web Services, Inc., Seattle, WA, "Elastic IP Addresses - Amazon Elastic Compute Cloud," [Online] Disponible:https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/elastic-ip-addresses-eip.html. Accedido en: Oct. 9, 2025.

[75] E. Koc, "Retrofit and OkHttp: A Comprehensive Guide", Medium, Jan. 2023. [Online]. Disponible: https://medium.com/@erdi.koc/retrofit-and-okhttp-675d34eb7458. Accedido en: Oct. 15, 2024.

[76] IronPDF, "Gson Java: Una guía completa para el procesamiento JSON", Sept. 2023. [Online]. Disponible: https://ironpdf.com/es/java/blog/java-help/gson-java/. Accedido en: Oct. 15, 2024.

[77] J. Nielsen, Usability Engineering, San Francisco, CA, USA: Morgan Kaufmann, 1993.

[78] A. Cooper, R. Reimann, D. Cronin, and C. Noessel, About Face: The Essentials of Interaction Design, 4th ed., Hoboken, NJ, USA: Wiley, 2014.

[79] Android Developers, “Jetpack Compose Overview,” Android Developers Documentation, 2025. [Online]. Disponible: https://developer.android.com/jetpack/compose. Accedido en: Mar. 18, 2025.

[80] "Baffling Concept of True Positive and True Negative," Towards Data Science. [Online]. Disponible: https://towardsdatascience.com/baffling-concept-of-true-positive-and-true-negative-bffbc340f107/. Accedido en: Oct. 9, 2025.

[81] "Classification: Accuracy, Recall, Precision, and Related Metrics," Google for Developers, Machine Learning Crash Course. [Online]. Disponible: https://developers.google.com/machine-learning/crash-course/classification/accuracy-precision-recall. Accedido en: Oct. 9, 2025.

[82] K. Pernice, "Task Scenarios for Usability Testing," Nielsen Norman Group, May 2019. [Online]. Disponible: https://www.nngroup.com/articles/task-scenarios-usability-testing/. Accedido en: Oct. 9, 2025.

[83] R. C. Martin, Clean Code: A Handbook of Agile Software Craftsmanship. Upper Saddle River, NJ, USA: Prentice Hall, 2008.

[84] GitHub Inc., "GitHub Copilot: Your AI pair programmer," Oct. 2024. [Online]. Disponible: https://github.com/features/copilot. Accedido en: Oct. 9, 2025.

[85] SonarSource SA, "SonarQube: Code Quality and Code Security," Oct. 2024. [Online]. Disponible: https://www.sonarqube.org/. Accedido en: Oct. 9, 2025.

[86] A. Wiggins, "The Twelve-Factor App," 2011. [Online]. Disponible: https://12factor.net/. Accedido en: Oct. 9, 2025.

[87] Google LLC, "Firebase Analytics," Google Firebase Documentation, Oct. 2024. [Online]. Disponible: https://firebase.google.com/products/analytics. Accedido en: Oct. 14, 2025.

[88] Atlassian, "Gitflow Workflow," Atlassian Git Tutorial, 2024. [Online]. Disponible: https://www.atlassian.com/git/tutorials/comparing-workflows/gitflow-workflow. Accedido en: Oct. 14, 2025.

[89] GitHub Inc., "Understanding GitHub Actions," GitHub Docs, Oct. 2024. [Online]. Disponible: https://docs.github.com/en/actions. Accedido en: Oct. 14, 2025.

[90] Android Developers, "Improve your code with lint checks", Google, 2024. [Online]. Disponible: https://developer.android.com/studio/write/lint. Accedido en: Oct. 15, 2024.

[91] Atlassian, "Kanban - A brief introduction," Apr. 2025. [Online]. Disponible: https://www.atlassian.com/agile/kanban. Accedido en: Oct. 5, 2025.

[92] K. Schwaber and J. Sutherland, "The Scrum Guide," Scrum.org, Nov. 2020. [Online]. Disponible: https://scrumguides.org/scrum-guide.html Accedido en: Oct. 5, 2025

[93] M. Cohn, Mountain Goat Software, "User Stories and User Story Examples", [Online]. Disponible: https://www.mountaingoatsoftware.com/agile/user-stories. Accedido en: Oct. 5, 2025.

[94] Kanbanize, "What is WIP and Why Does it Matter?", [Online]. Disponible: https://kanbanize.com/kanban-resources/getting-started/what-is-wip. Accedido en: Oct. 5, 2025.

[95] Atlassian, "Confluence," 2025. [Online]. Disponible: https://www.atlassian.com/software/confluence. Accedido en: Oct. 5, 2025

[96] Clockify, "Clockify - Free Time Tracker and Timesheet App," 2025. [Online]. Disponible: https://clockify.me. Accedido en: Oct. 5, 2025.

[97] Cucumber, "Gherkin Reference," 2025. [Online]. Disponible: https://cucumber.io/docs/gherkin/reference/. Accedido en: Oct. 5, 2025.

[98] Atlassian, "Story Points and Estimation," 2025. [Online]. Disponible: https://www.atlassian.com/agile/project-management/estimation. Accedido en: Oct. 5, 2025.

[99] Scrum.org, "Ready or Not? Demystifying the Definition of Ready in Scrum," 2025. [Online]. Disponible: https://www.scrum.org/resources/blog/ready-or-not-demystifying-definition-ready-scrum. Accedido en: Oct. 5, 2025.

[100] Scrum.org, "What is a Definition of Done?," 2025. [Online]. Disponible: https://www.scrum.org/resources/what-definition-done. Accedido en: Oct. 5, 2025.

11. ANEXOS

ANEXO 1

Análisis de Competidores y Oportunidades para SafeCall

Competidor	Características Principales	Debilidades Identificadas	Oportunidades para SafeCall
Truecaller	• Base de datos comunitaria de 300+ millones de usuarios • Identificación de spam • Bloqueo automático • 40 mil millones de llamadas spam bloqueadas anualmente	• **Detección reactiva:** Solo identifica números ya reportados como spam • **Sin análisis en tiempo real:** No detecta fraudes durante la llamada en curso • **Dependencia de base de datos externa:** Requiere conexión y comunidad activa • **No detecta voces sintéticas/IA:** No analiza el contenido de la llamada	• Detección proactiva en tiempo real durante la llamada • Análisis de contenido y voz con IA • Funcionamiento local sin dependencia de bases de datos comunitarias • Detección de voces generadas por IA
Hiya	• Protección contra llamadas no	• **Enfoque en identificación, no en contenido:** No	• Análisis de contenido conversacional en

	deseadas • Tecnología de IA para identificación • Bloqueo preventivo	analiza qué se dice en la llamada • **Sin procesamiento en tiempo real:** Actúa antes o después de la llamada, no durante • **No previene fraudes complejos:** No detecta técnicas de ingeniería social durante la conversación	tiempo real • Detección de patrones de ingeniería social • Alertas durante la llamada activa • Protección contra fraudes sofisticados
RoboKiller	• Filtrado automático con machine learning • Enfocado en bloqueo de robocalls • Base de datos actualizada	• **Limitado a robocalls automatizadas:** No protege contra estafadores humanos o híbridos • **Sin análisis de voz en tiempo real:** No detecta voces clonadas o generadas por IA • **Bloqueo preventivo únicamente:** No asiste durante	• Detección de fraudes humanos y automatizados • Análisis de autenticidad de voz • Protección durante llamadas en curso • Identificación de cambios de comportamiento durante la llamada

		llamadas legítimas que se tornan fraudulentas	
Call Recorder / CallApp	• Grabación de llamadas • Transcripción voz a texto • Gestión de llamadas básica	• **Procesamiento post-llamada:** Solo actúan después de finalizada la llamada • **Sin prevención activa:** No alertan durante el fraude en curso • **Análisis manual requerido:** El usuario debe revisar grabaciones • **No detectan fraude automáticamente:** Son herramientas pasivas	• Prevención activa en tiempo real • Alertas inmediatas durante la llamada • Análisis automático sin intervención del usuario • Evita el fraude antes de que se concrete
AI-SPY / AI Voice Detector	• Detección de audio generado por IA • Análisis de audios grabados	• **Solo funcionan con audios pre-grabados:** No trabajan en tiempo real • **Proceso manual:** Usuario debe	• Integración directa con sistema de llamadas • Procesamiento automático en tiempo real • Optimización para

		grabar y subir el audio • **Baja precisión reportada:** Usuarios reportan falsos positivos/negativos • **Limitaciones de idioma:** Problemas con idiomas distintos al inglés • **No integrado con llamadas:** Funcionan como herramientas separadas	español y contexto uruguayo • Sin necesidad de acción manual del usuario • Mayor precisión contextual

Tabla 15 - Análisis de Competidores y Oportunidades para SafeCall

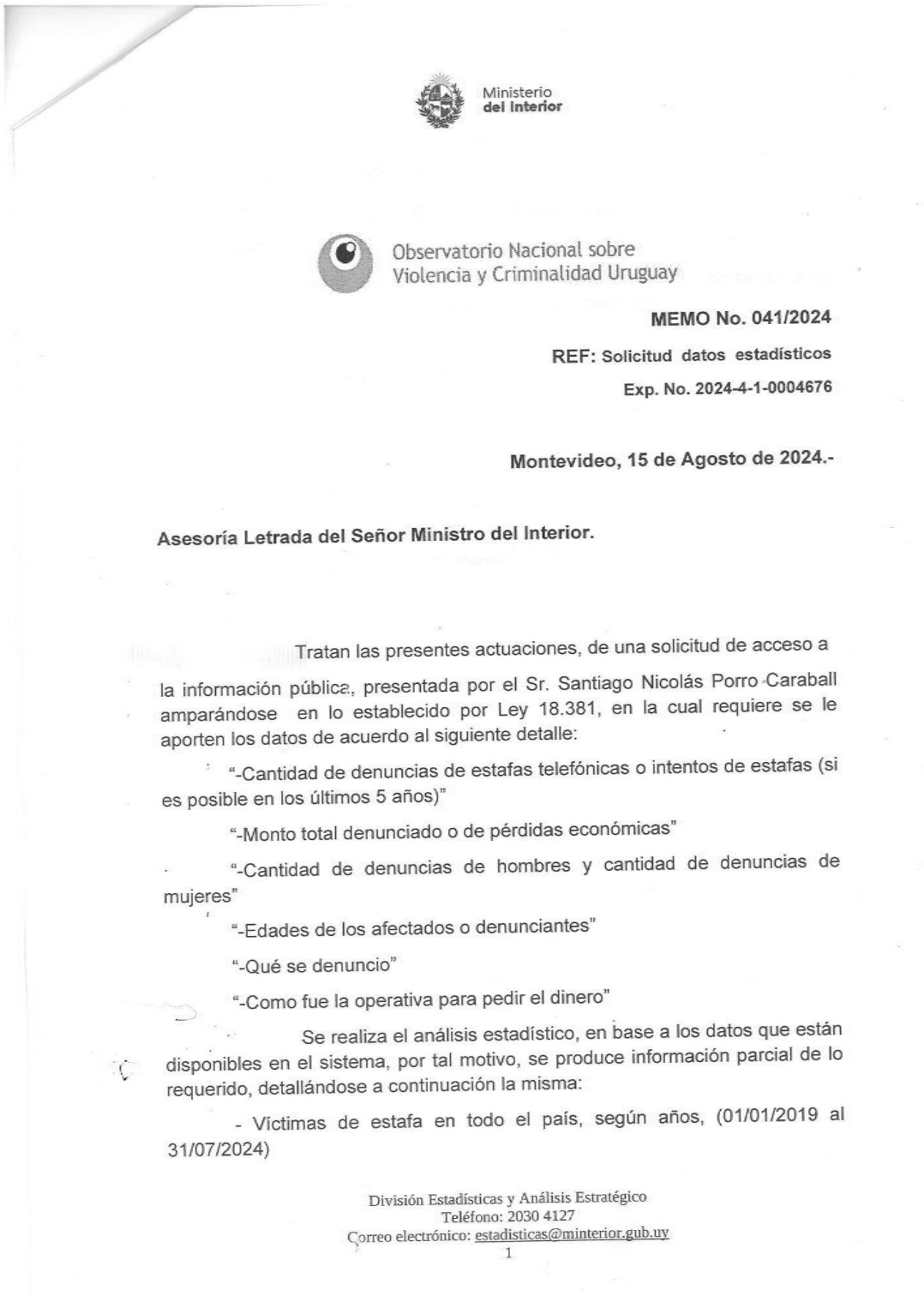
Oportunidades Potenciales Identificadas para SafeCall

El análisis competitivo reveló oportunidades significativas que SafeCall podría aprovechar en su desarrollo. La principal brecha detectada es la ausencia de detección de fraudes en tiempo real durante la llamada activa en el mercado actual. Mientras Truecaller, Hiya y RoboKiller se limitan a identificar y bloquear números spam de forma reactiva, existía la posibilidad de desarrollar una solución que analizara el contenido y contexto de la conversación mientras ocurre, detectando patrones de ingeniería social como urgencia artificial, amenazas o solicitud de datos sensibles.

Otra oportunidad importante sería la capacidad de identificar voces generadas o clonadas por IA, un diferenciador crítico ante el creciente uso de tecnología de clonación de voz en estafas, área que competidores como AI-SPY abordan solo de forma manual y post-llamada.

El procesamiento local representaría una ventaja competitiva en privacidad frente a soluciones dependientes de bases de datos comunitarias externas. Estas oportunidades sugerían que SafeCall podría posicionarse como una solución integral que combinara prevención, detección y protección en tiempo real.

Solicitud de Información realizada al Ministerio del Interior



Ministerio
del Interior

Observatorio Nacional sobre
Violencia y Criminalidad Uruguay

MEMO No. 041/2024

REF: Solicitud datos estadísticos

Exp. No. 2024-4-1-0004676

Montevideo, 15 de Agosto de 2024.-

Asesoría Letrada del Señor Ministro del Interior.

Tratan las presentes actuaciones, de una solicitud de acceso a la información pública, presentada por el Sr. Santiago Nicolás Porro Caraball amparándose en lo establecido por Ley 18.381, en la cual requiere se le aporten los datos de acuerdo al siguiente detalle:

"-Cantidad de denuncias de estafas telefónicas o intentos de estafas (si es posible en los últimos 5 años)"

"-Monto total denunciado o de pérdidas económicas"

"-Cantidad de denuncias de hombres y cantidad de denuncias de mujeres"

"-Edades de los afectados o denunciantes"

"-Qué se denuncio"

"-Como fue la operativa para pedir el dinero"

Se realiza el análisis estadístico, en base a los datos que están disponibles en el sistema, por tal motivo, se produce información parcial de lo requerido, detallándose a continuación la misma:

- Víctimas de estafa en todo el país, según años, (01/01/2019 al 31/07/2024)

División Estadísticas y Análisis Estratégico
Teléfono: 2030 4127
Correo electrónico: estadisticas@minterior.gub.uy
1

Ilustración 25 - Documento Oficial de Solicitud de Informacion al Ministerio del Interior

Ministerio
del Interior

- Víctimas de estafa según año y sexo
- Víctimas de estafa según tramo de edad y año
- Víctimas de estafa según edad y sexo

Atento a lo precedente, se eleva el presente, quedando a su disposición para cualquier consulta que considere oportuna.-

Saluda a Usted Atentamente

DIRECTOR DE DIVISIÓN ESTADÍSTICAS
Y ANÁLISIS ESTRATÉGICO
Soc. Javier DONNANGELO

División Estadísticas y Análisis Estratégico
Teléfono: 2030 4127
Correo electrónico: estadisticas@minterior.gub.uy

Ilustración 26 - Documento Oficial de Solicitud de Informacion al Ministerio del Interior (cont.)

Víctimas de estafa según tramos de edad y año
Todo el país- 01 de enero de 2019 al 31 de julio de 2024

TRAMOS DE EDAD	2019	2020	2021	2022	2023	2024
70 a 74 años	181	320	491	591	548	417
75 a 79 años	170	334	362	431	490	299
80 a 84 años	146	294	282	336	345	211
85 a 89 años	116	196	156	214	257	137
90 a 94 años	50	76	74	105	104	50
95 a 99 años	7	5	14	20	21	11
100 años y más	1	3	1	1	3	2

FUENTE: Datos elaborados en base al Sistema de Gestión de Seguridad Pública.
FECHA DE ELABORACIÓN: 15/08/2024
DEPENDENCIA: División de Estadísticas y Análisis Estratégico, Observatorio Nacional sobre Violencia y Criminalidad, Ministerio del Interior
NOTA: No distingue por modalidad de estafa.

Víctimas de estafa según tramos de edad y sexo.
Todo el país- 01 de enero de 2019 al 31 de julio de 2024

TRAMOS DE EDAD	MUJERES	HOMBRES
70 a 74 años	1520	1028
75 a 79 años	1385	701
80 a 84 años	1165	449
85 a 89 años	807	269
90 a 94 años	354	105
95 a 99 años	58	20
100 años y más	8	3
TOTAL	5297	2575

FUENTE: Datos elaborados en base al Sistema de Gestión de Seguridad Pública.
FECHA DE ELABORACIÓN: 15/08/2024
DEPENDENCIA: División de Estadísticas y Análisis Estratégico, Observatorio Nacional sobre Violencia y Criminalidad, Ministerio del Interior
NOTA: No distingue por modalidad de estafa.

Ilustración 27 - Datos Oficiales de Estafas Telefónicas en Uruguay 2024

Víctimas de estafa según año
Todo el país- 01 de enero de 2019 al 31 de julio de 2024

AÑO	VICTIMAS
2019	671
2020	1228
2021	1380
2022	1698
2023	1768
2024	1127
Total	7872

FUENTE: Datos elaborados en base al Sistema de Gestión de Seguridad Pública.
FECHA DE ELABORACIÓN: 15/08/2024
DEPENDENCIA: División de Estadísticas y Análisis Estratégico, Observatorio Nacional sobre Violencia y Criminalidad, Ministerio del Interior
NOTA: No distingue por modalidad de estafa.

Víctimas de estafa según año y sexo
Todo el país- 01 de enero de 2019 al 31 de julio de 2024

AÑO	MUJERES	HOMBRES
2019	494	177
2020	887	341
2021	927	453
2022	1134	564
2023	1138	630
2024	717	410
Total	5297	2575

FUENTE: Datos elaborados en base al Sistema de Gestión de Seguridad Pública.
FECHA DE ELABORACIÓN: 15/08/2024
DEPENDENCIA: División de Estadísticas y Análisis Estratégico, Observatorio Nacional sobre Violencia y Criminalidad, Ministerio del Interior
NOTA: No distingue por modalidad de estafa.

Ilustración 28 - Victimas de Estafa por año y genero

Encuesta realizada

Dimensión	Preguntas clave	Importancia
Prevalencia	¿Cuántas personas han recibido intentos de estafa? ¿Con qué frecuencia?	Dimensionar la urgencia del problema y validar la necesidad de una solución
Perfil demográfico	¿Qué rangos etarios son más afectados?	Validar si la vulnerabilidad se concentra en adultos mayores o es transversal
Modalidades	¿Qué tipos de estafas son más comunes? (premios, suplantación bancaria, cuento del tío)	Orientar el entrenamiento del modelo de detección hacia patrones prevalentes en Uruguay
Señales de alerta	¿Cómo identifican las personas una llamada sospechosa?	Comprender qué mecanismos de detección intuitivos funcionan y cuáles fallan
Comportamientos	¿Cómo responden ante llamadas de números desconocidos?	Identificar patrones de riesgo en la población
Protección actual	¿Qué herramientas o métodos utilizan hoy?	Revelar brechas en la adopción de soluciones existentes
Demanda de solución	¿Qué características valorarían? ¿Cuánto pagarían?	Definir el alcance del MVP y validar la viabilidad de una solución.
Barreras de adopción	¿Qué preocupaciones	Anticipar obstáculos para

	tienen sobre soluciones tecnológicas?	la adopción del producto

Tabla 16 - Preguntas realizadas en encuesta

Resultados de Encuesta

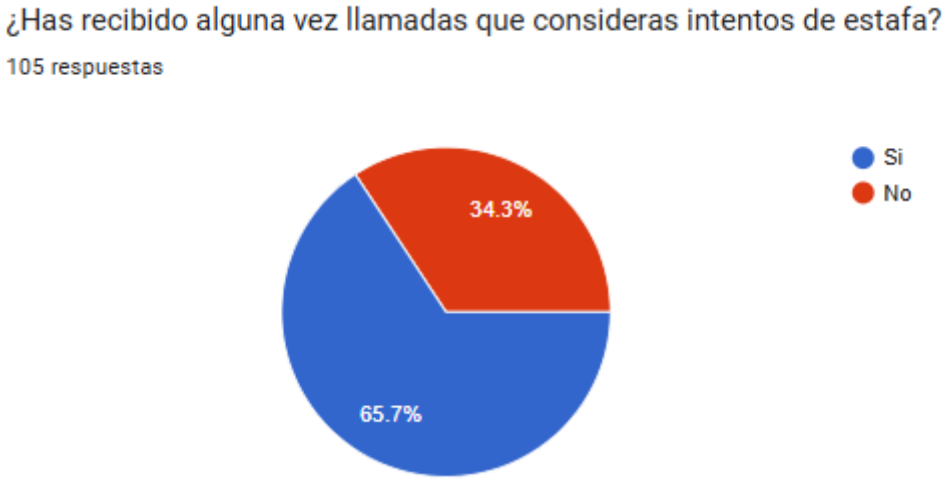


Ilustración 29 - Pregunta sobre intentos de estafa

Ilustración 30 - Pregunta Rango de Edad

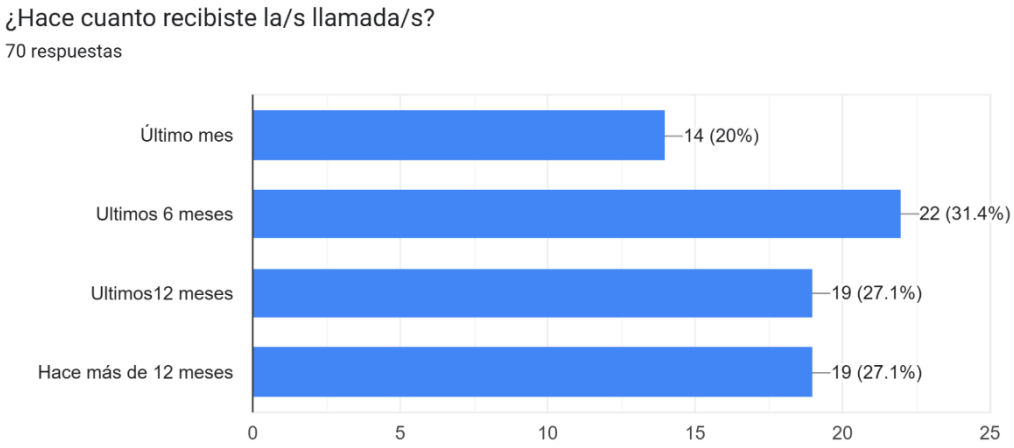


Ilustración 31 - Pregunta ubicación temporal de llamada

Ilustración 32 - Pregunta frecuencia de llamada

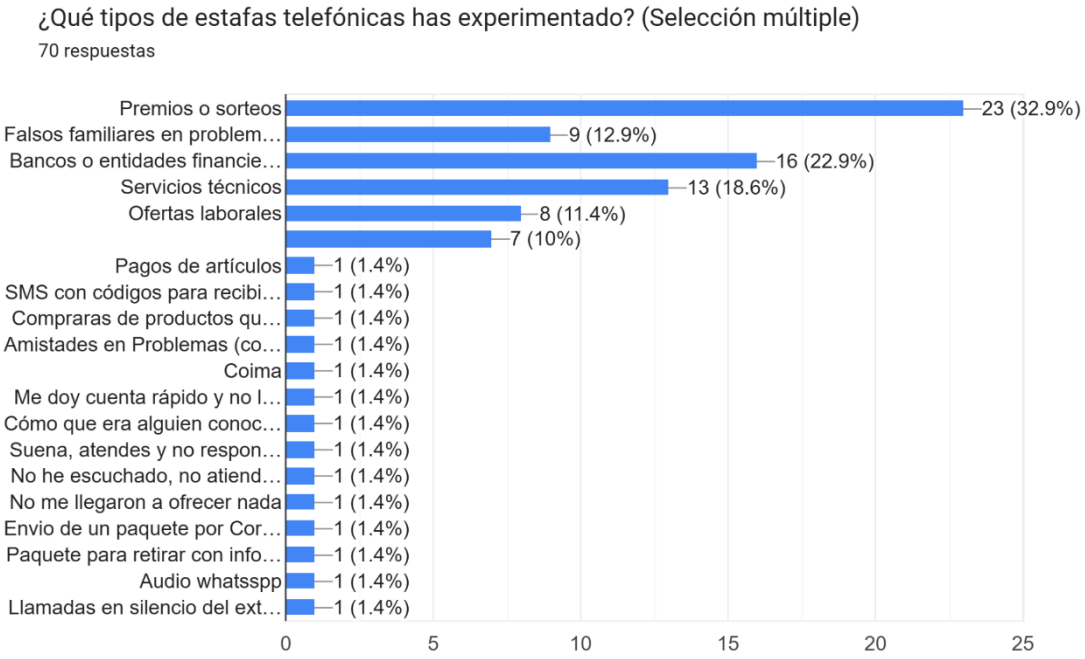


Ilustración 33 - Pregunta sobre tipo de estafa recibida

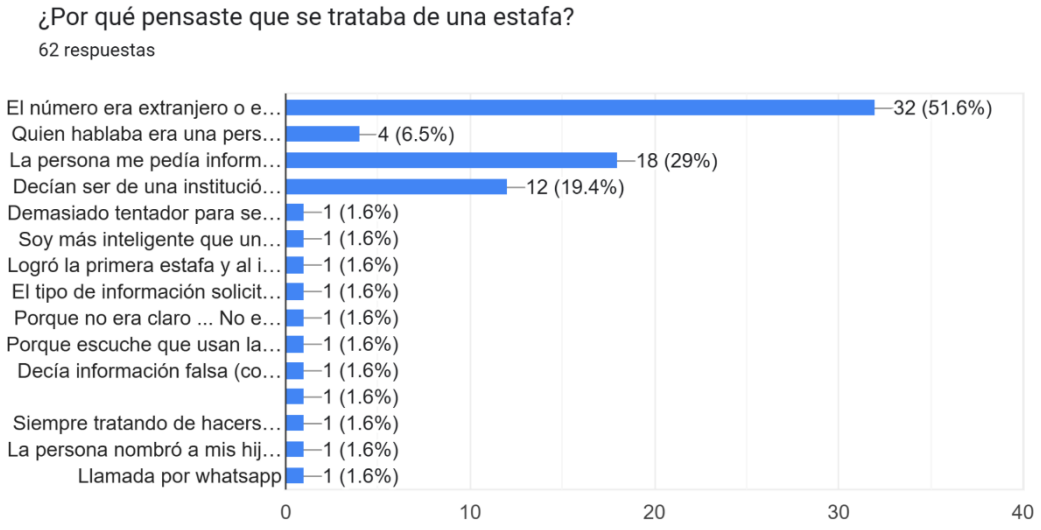


Ilustración 34 - Pregunta sobre reconocimiento de estafa

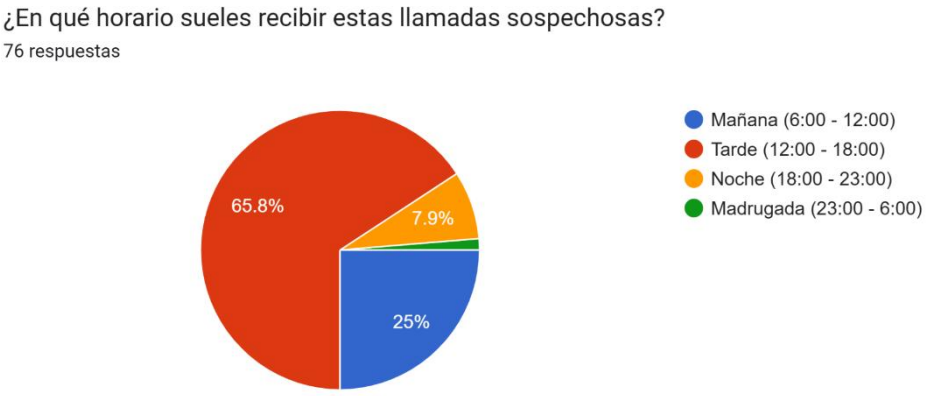


Ilustración 35 - Pregunta sobre horario de llamada

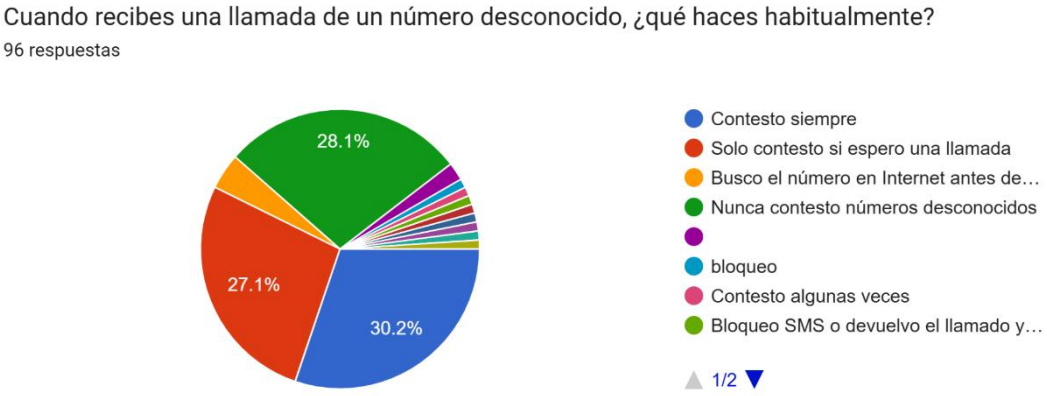


Ilustración 36 - Pregunta sobre reacción de usuario a llamada desconocida

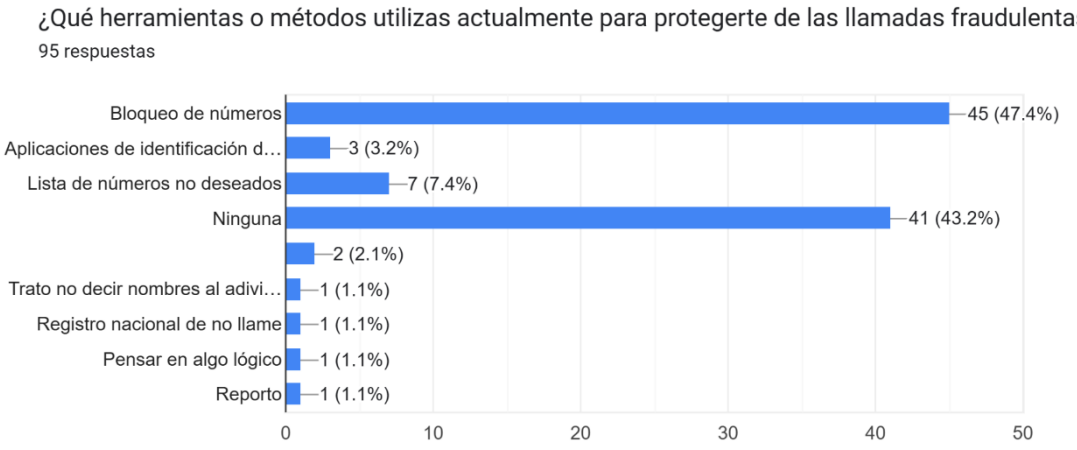


Ilustración 37 - Pregunta herramientas de protección

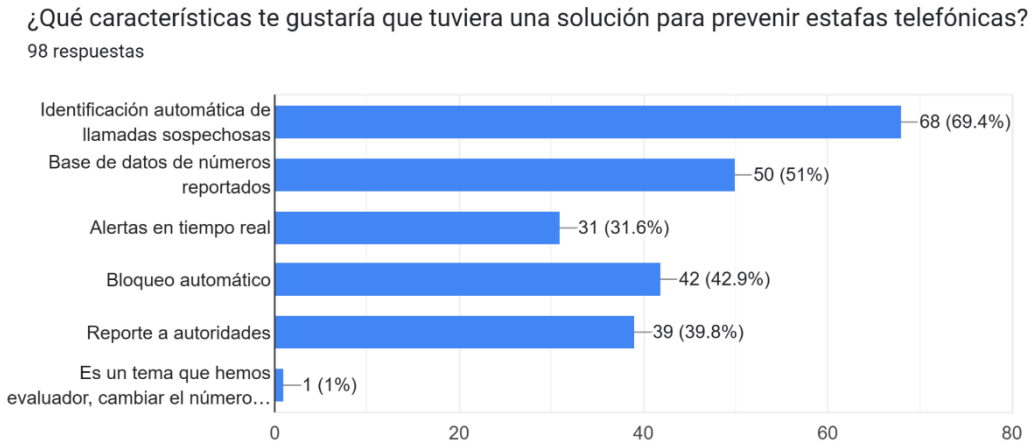


Ilustración 38 - Pregunta sobre características de solución

Ilustración 39 - Preguntas sobre monetización

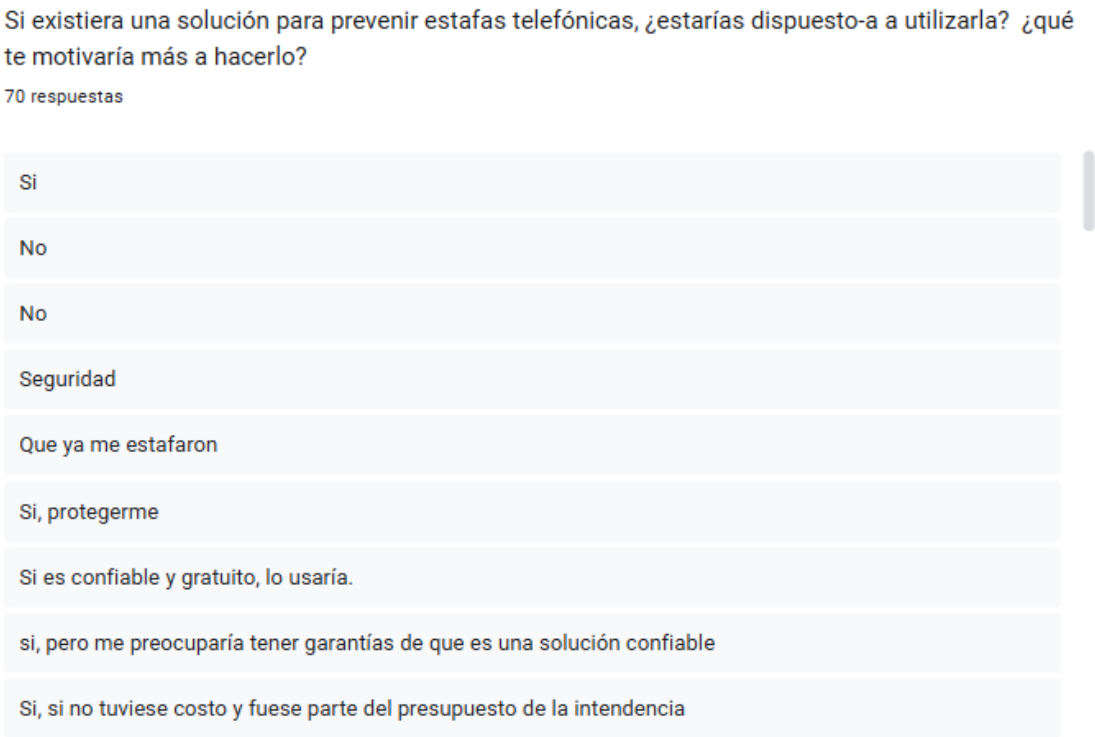


Ilustración 40 - Pregunta sobre uso de aplicación

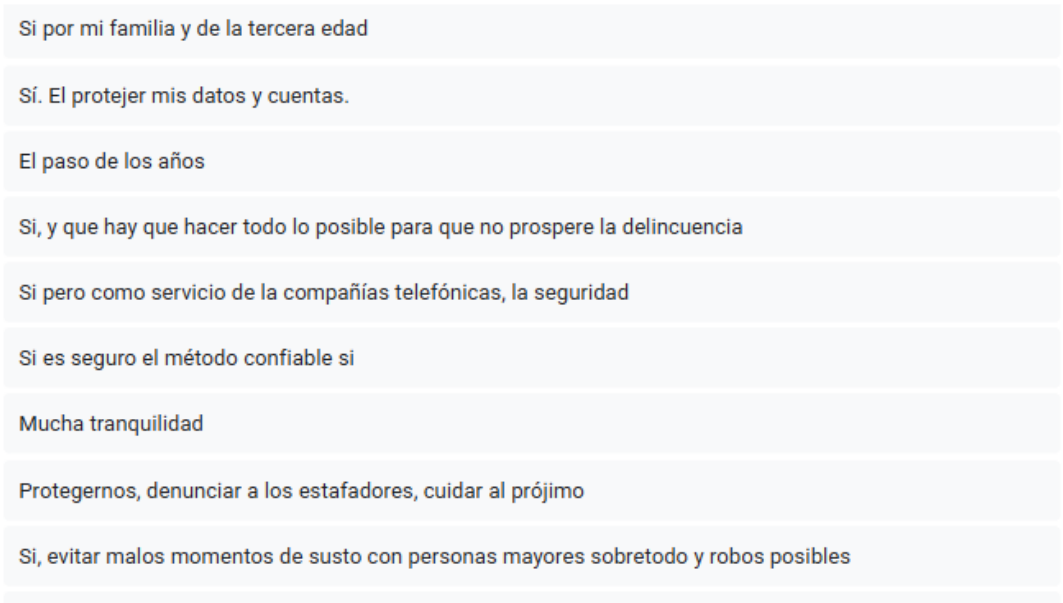


Ilustración 41 - Pregunta sobre uso de aplicación (cont.)

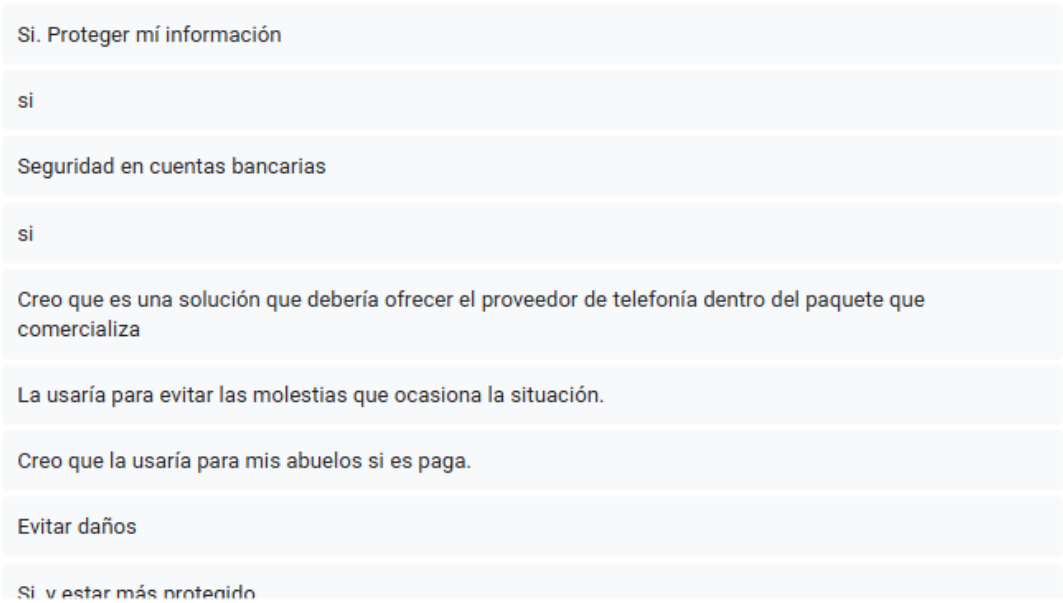


Ilustración 42 - Pregunta sobre uso de aplicación (cont.)

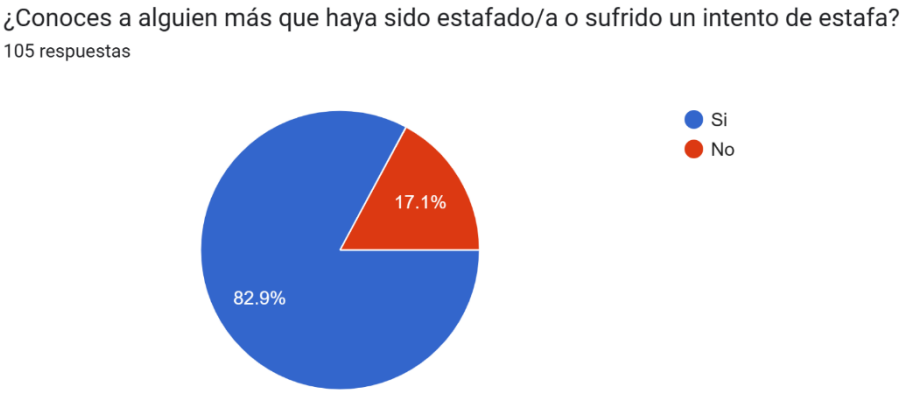


Ilustración 43 - Pregunta casos cercanos de estafa

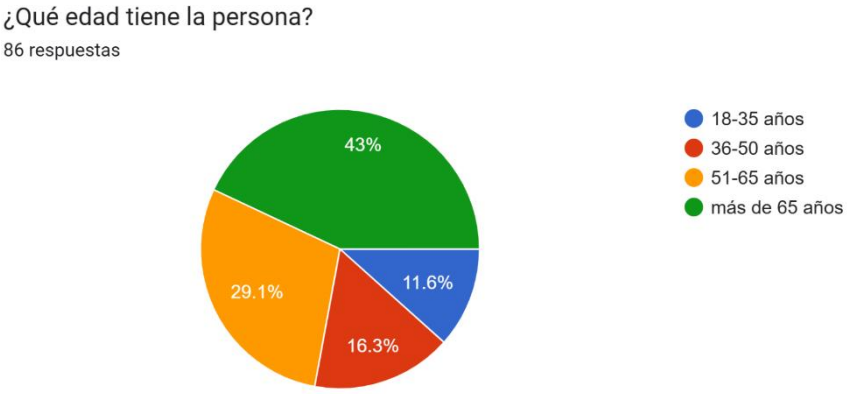


Ilustración 44 - Pregunta de edad sobre caso cercano

Entrevistas realizadas a Usuarios

Entrevista 1:

Nombre:	Paula Alvarez (sobre sus abuelos)
Edad:	90 años (abuelos)
Ocupación:	Jubilados
Conocimiento tecnológico:	No usan aplicaciones. Uso muy básico del celular. Requieren ayuda de cuidadora.
Principales Descubrimientos: • Cayeron en estafa telefónica tipo 'cuento del tío' • Se hicieron pasar por sobrino en situación de emergencia pidiendo dinero • La hija intervino y cortó la estafa llamando al supuesto familiar • Después quedaron más alertas y la familia los concientizó • Actualmente casi no atienden el teléfono y cortan si no conocen el número	

Tabla 17 - Entrevista 1

Entrevista 2:

Nombre:	Paula Alvarez
Edad:	29 años
Ocupación:	Consultora Técnica, Profesora y Geóloga
Conocimiento tecnológico:	Conocimiento medio-avanzado. Usa aplicaciones pero con precaución.
Principales Descubrimientos: • Recibió mensajes de número desconocido de otro país intentando sacar conversación	

- También recibió mensajes de falso paquete retenido en aduana
- Desconfía siempre de llamadas, evita dar información propia
- Busca contactar entidades por sus propios medios
- Utilizaría aplicación si no cobra y no ocupa mucho espacio de almacenamiento
- Requiere que se declare explícitamente qué se hará con sus datos
- Le gustaría que el acceso al micrófono sea solo durante llamadas
- Busca respaldo de entidad telefónica o URSEC

Tabla 18 - Entrevista 2

Entrevista 3:

Nombre:	Amanda
Edad:	29 años
Ocupación:	Enablement Lead
Conocimiento tecnológico:	Medio. Usuario promedio de tecnología

Principales Descubrimientos:

- Cayó en estafa vendiendo TV por Marketplace
- Le enviaron falso comprobante de transferencia por monto mayor
- Llamaron haciéndose pasar por gerente del banco, llorando y presionando
- Aprovecharon momento de nerviosismo (viernes en la tarde, feriado bancario)
- No sabían información del banco, querían extraer más datos
- Desearía alertas como Two-Factor Authentication 2FA o token de transferencia
- Aceptaría que escuche todas las llamadas
- No daría acceso a galería de su teléfono
- Quiere que se le avise en el momento sin tomar decisiones
- Busca respaldo institucional o empresa telefónica internacional

Tabla 19 - Entrevista 3

Entrevista 4:

Nombre:	Sonia
Edad:	80 años
Ocupación:	Jubilada
Conocimiento tecnológico:	Uso muy rudimentario de tecnología. Uso del celular muy básico.
Principales Descubrimientos: • La intentaron estafar por llamada haciéndose pasar por un hijo • Sospechó por tono de voz y forma de tratarla (grosera, apurada) • Cortó inmediatamente y llamó a los hijos para verificar • Se sintió invadida por la experiencia • No da ningún dato por teléfono • Hoy casi no atiende el teléfono, corta si no conoce • Instalaría app si es sugerida por alguien de confianza • Le gustaría que detecte estafa y avise a familiares • Conoce vecinos que han recibido llamadas fingiendo deudas	

Tabla 20 - Entrevista 4

Entrevista 5:

Nombre:	Noella
Edad:	47 años
Ocupación:	Psicóloga
Conocimiento tecnológico:	Usuaria competente de tecnología
Principales Descubrimientos:	

- No ha sido víctima directa pero está consciente del riesgo
- Le molesta recibir llamados constantes de números desconocidos

- Le gustaría que identifique números reportados como peligrosos
- Quiere saber si el número es extranjero y qué información tiene
- Daría acceso si la app es segura y conoce su procedencia
- Valida identidad de entidades por información que manejan
- Busca respaldo de entidad importante (ej. Antel)
- Le haría sentir más segura la aplicación con ese respaldo
- No tendría problema si conoce cómo trabaja la solución

Tabla 21 - Entrevista 5

Entrevista 6:

Nombre:	Maria Jesus
Edad:	32
Ocupación:	Experta en ciberseguridad
Conocimiento tecnológico:	Alto conocimiento y dominio de tecnologías. Uso profesional avanzado.

Principales Descubrimientos:

- No ha recibido llamadas por estafa personalmente
- Le molesta recibir llamados de numeros desconocidos o extranjeros en horario de trabajo y sobre todo fuera de el
- Utilizaría aplicaciones de detección
- No tendría problema en dar acceso a todas las llamadas
- Conoce dos casos relativamente jóvenes vía marketplace con falsas transferencias
- Conoce familiar estafado por WhatsApp
- Se hicieron pasar por ente público pidiendo actualizar datos
- No le gustaría que acceda a información más allá de llamadas

• Le gustaría poder denunciar desde la aplicación • La app la haría sentir más segura • Prefiere apps fáciles de usar con poca configuración • Requiere confianza sobre almacenamiento y uso de datos

Tabla 22 - Entrevista 6

Entrevista 7:

Nombre:	Raúl
Edad:	84 años
Ocupación:	Jubilado
Conocimiento tecnológico:	Prácticamente nulo. Solo utiliza teléfono para llamadas y mensajes de texto
Principales Descubrimientos: • Valora que la solución sea sencilla de usar • No lo han llegado a estafar pero recibe llamados muy seguido y confunde fácilmente la voz de las personas • Le gustaría que "el teléfono se de cuenta solo de la estafa y avise a mi familia" • Importante que pueda usar en la casa y en la calle	

Tabla 23 - Entrevista 7

Entrevista 8:

Nombre:	Irma
Edad:	82 años
Ocupación:	Ama de casa

Conocimiento tecnológico:	Conocimiento muy básico. Tiene familia en frontera y se comunica con ellos por teléfono
Principales Descubrimientos: • Fue víctima del "cuento del tío". • Se hicieron pasar por un familiar y creyendo que era uno de los que suele comunicarse, no desconfió. • No se llegó a concretar la estafa. Antes de ir a realizar un depósito, su hija que vive en frente de su casa cruzó y hablando con ella la hizo entrar en razón • Necesita que la solución detecte sola la llamada sospechosa y le avise a ella y a su hija • Tiene problema en la vista y necesita carteles y letras grandes	

Tabla 24 - Entrevista 8

Entrevistas realizadas a Expertos

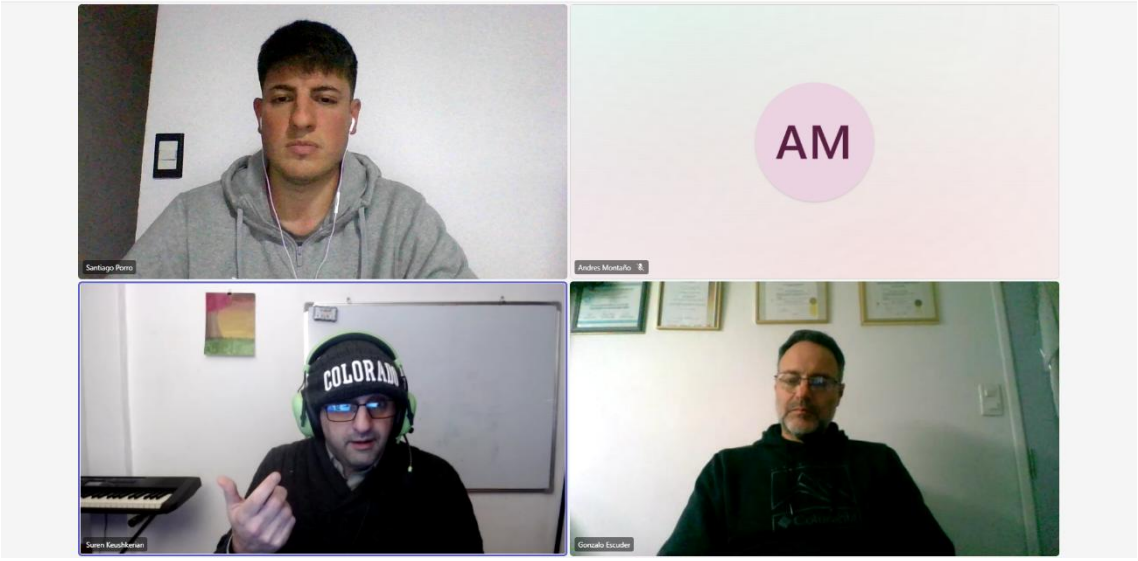


Ilustración 45 - Entrevista con Gonzalo Escuder (Antel)

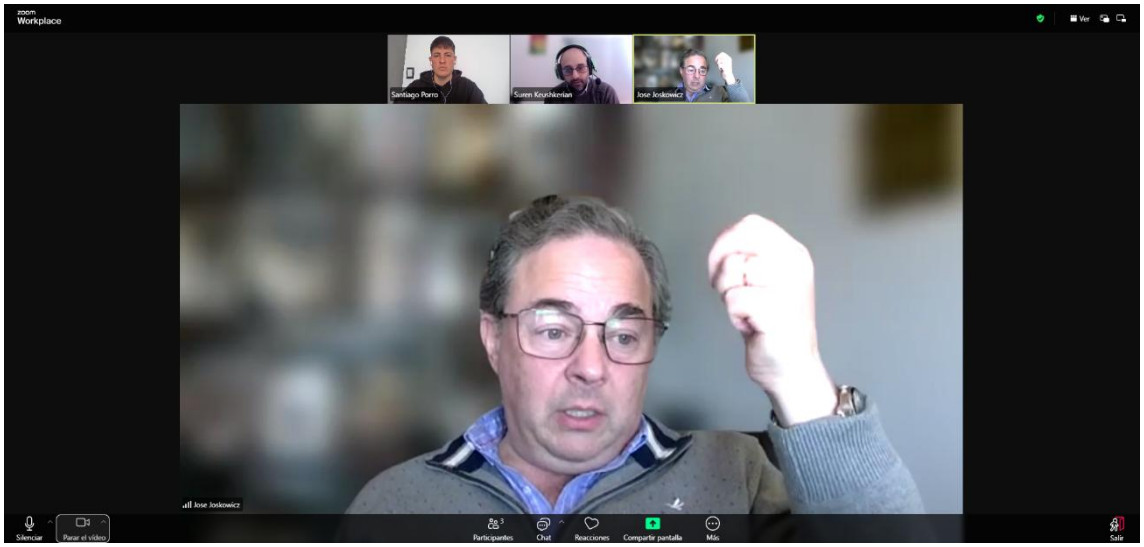


Ilustración 46 - Entrevista a José Joskowicz (Isbel)

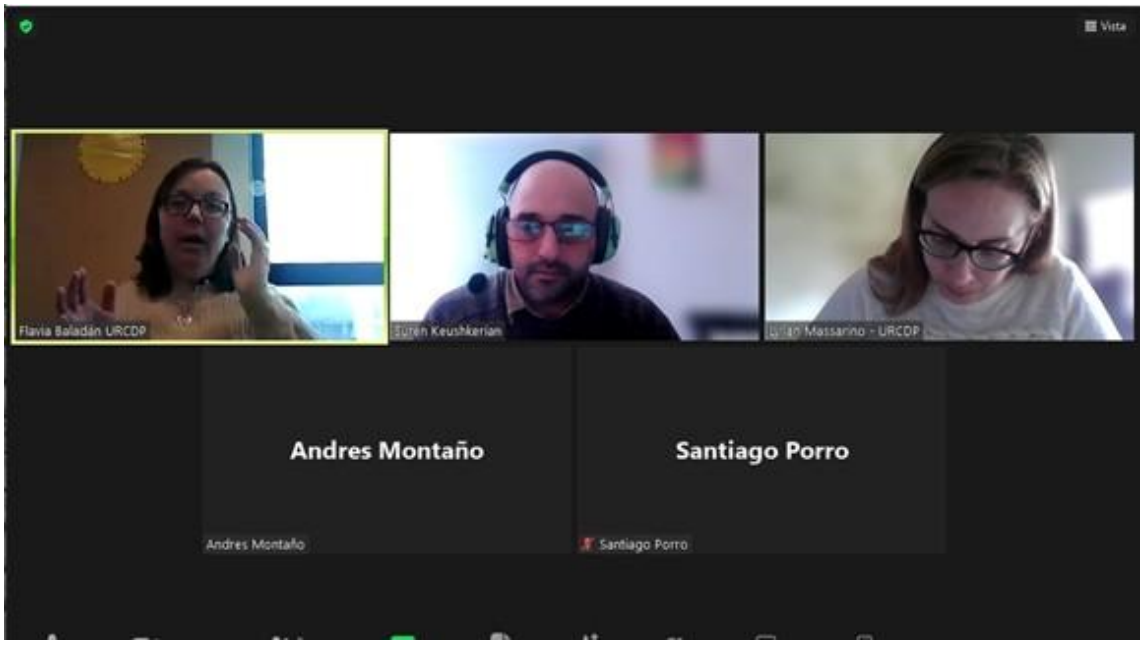


Ilustración 47 - Entrevista a Lylian Massarino y Flavia Baladan (URCDP)

Tablero de Miró - “Saturar y Agrupar” hallazgos

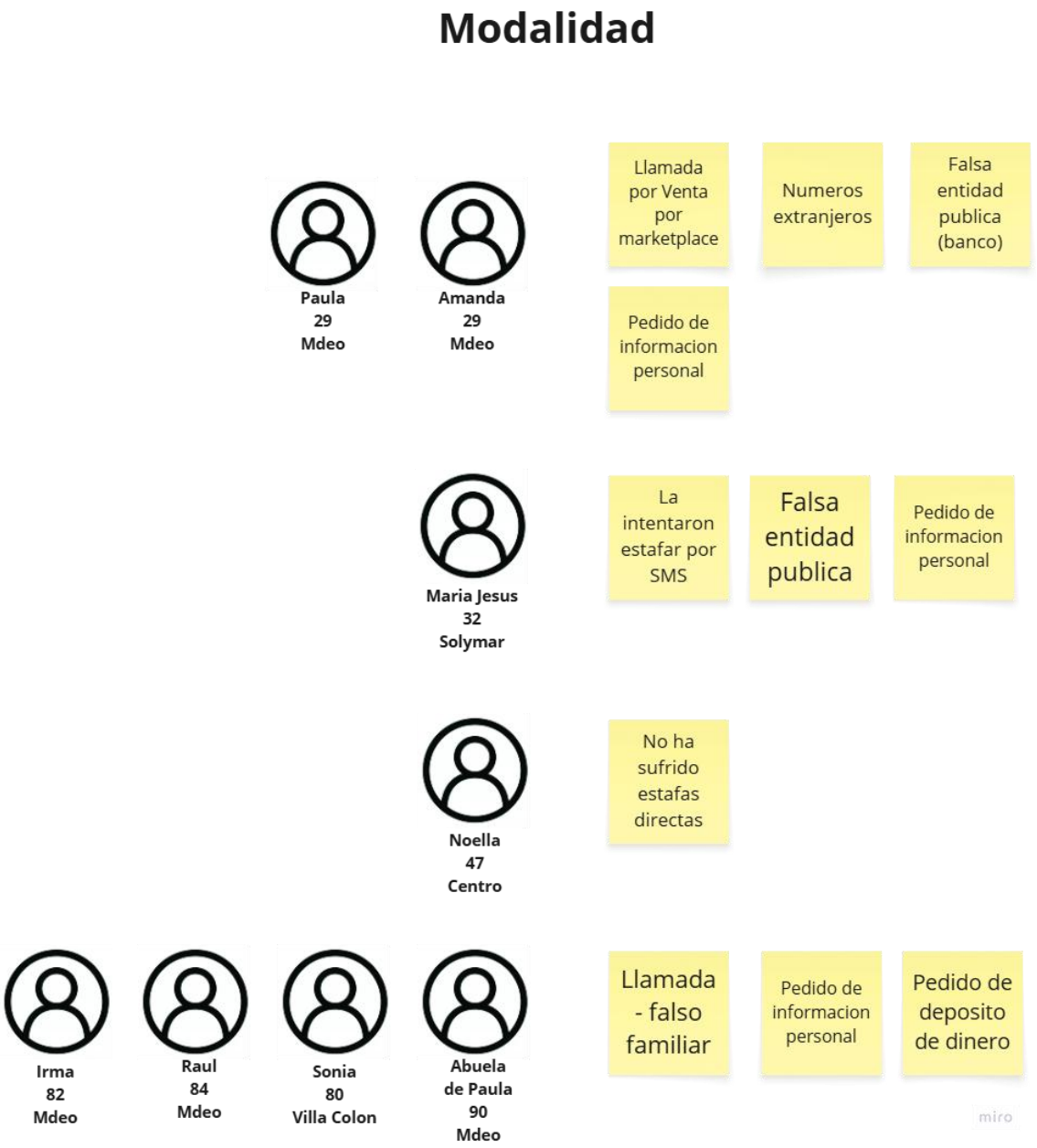


Ilustración 48 - Agrupación por modalidad

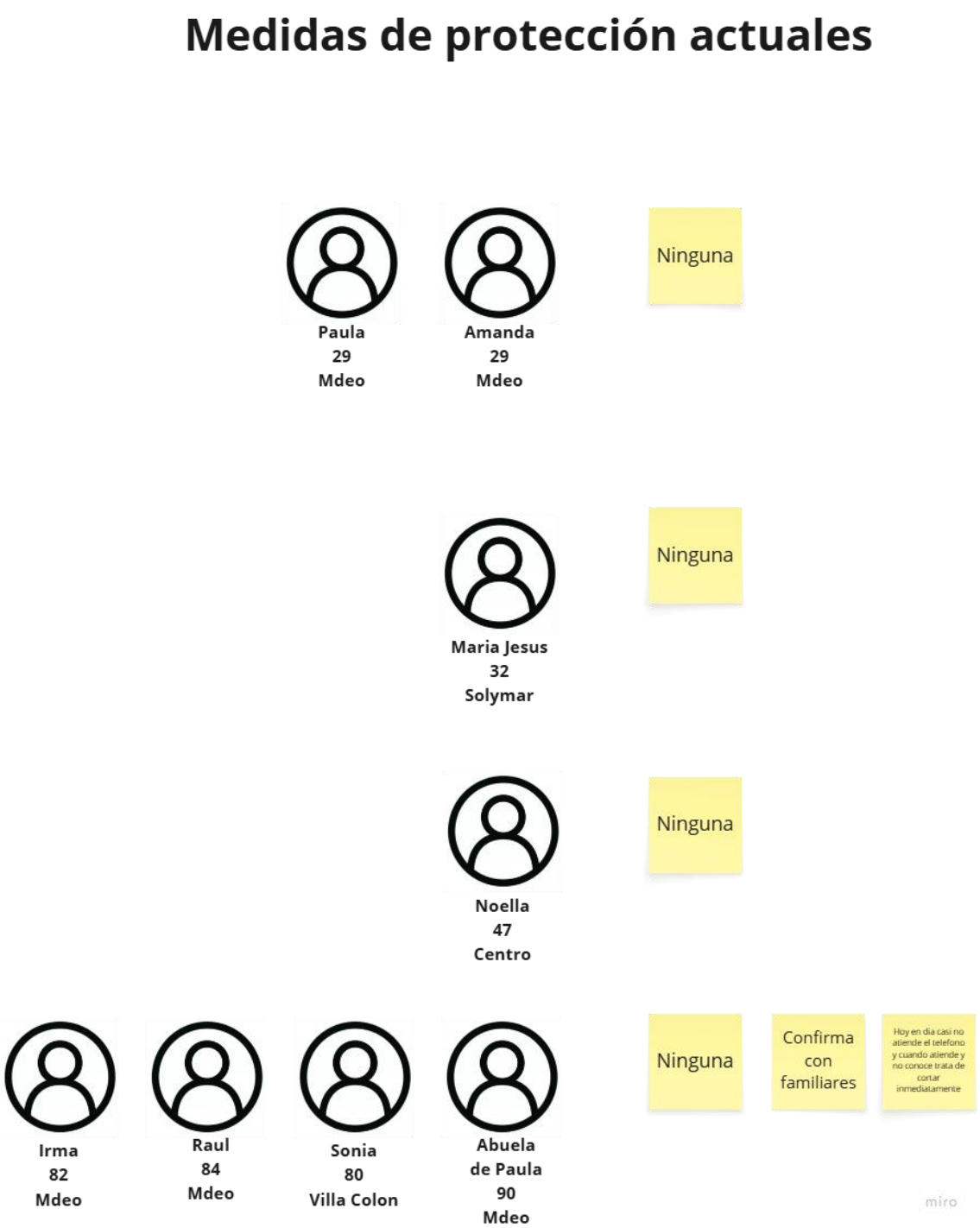


Ilustración 49 - Agrupación por medidas de protección

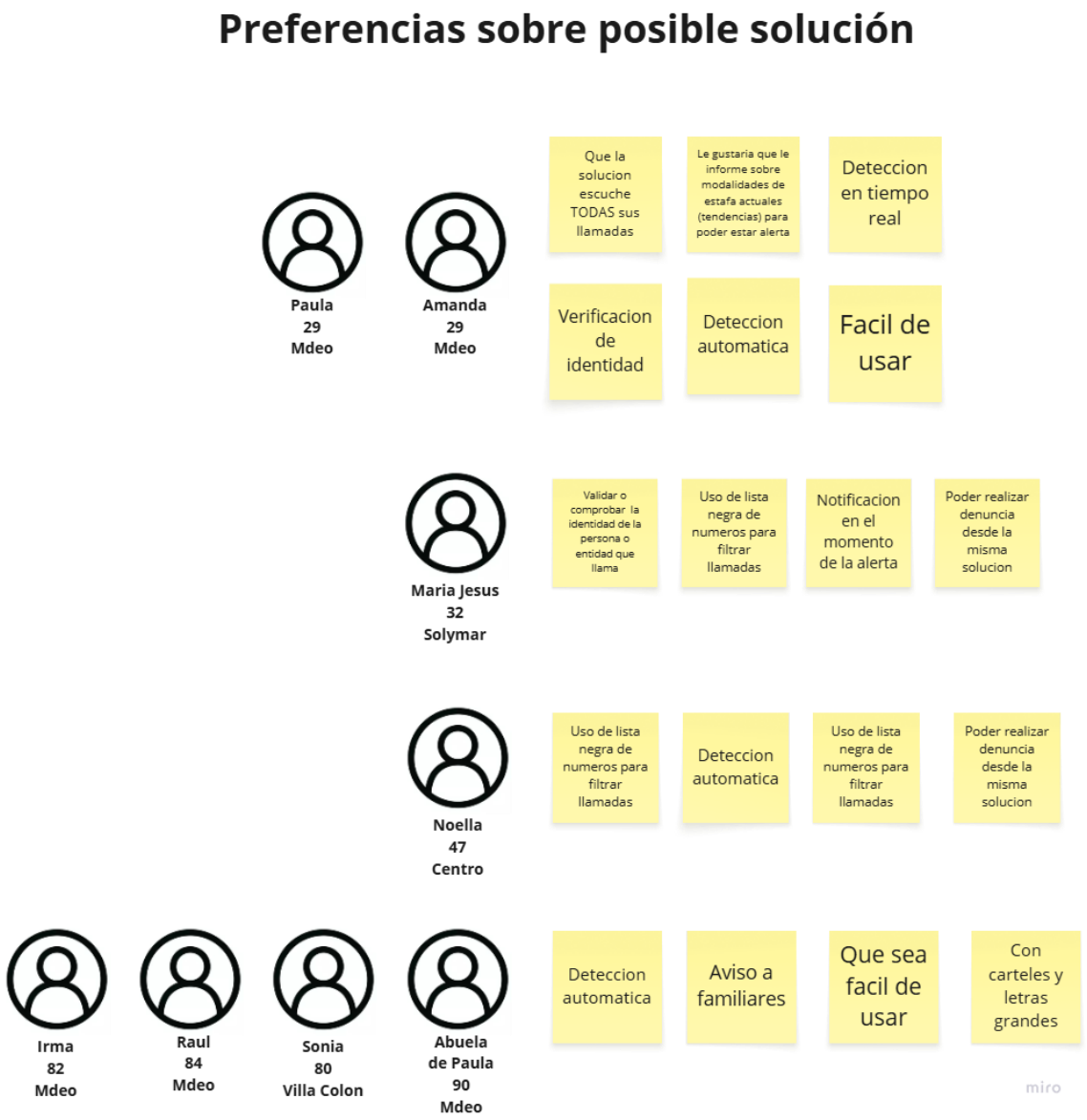


Ilustración 50 - Agrupación por preferencias de solución

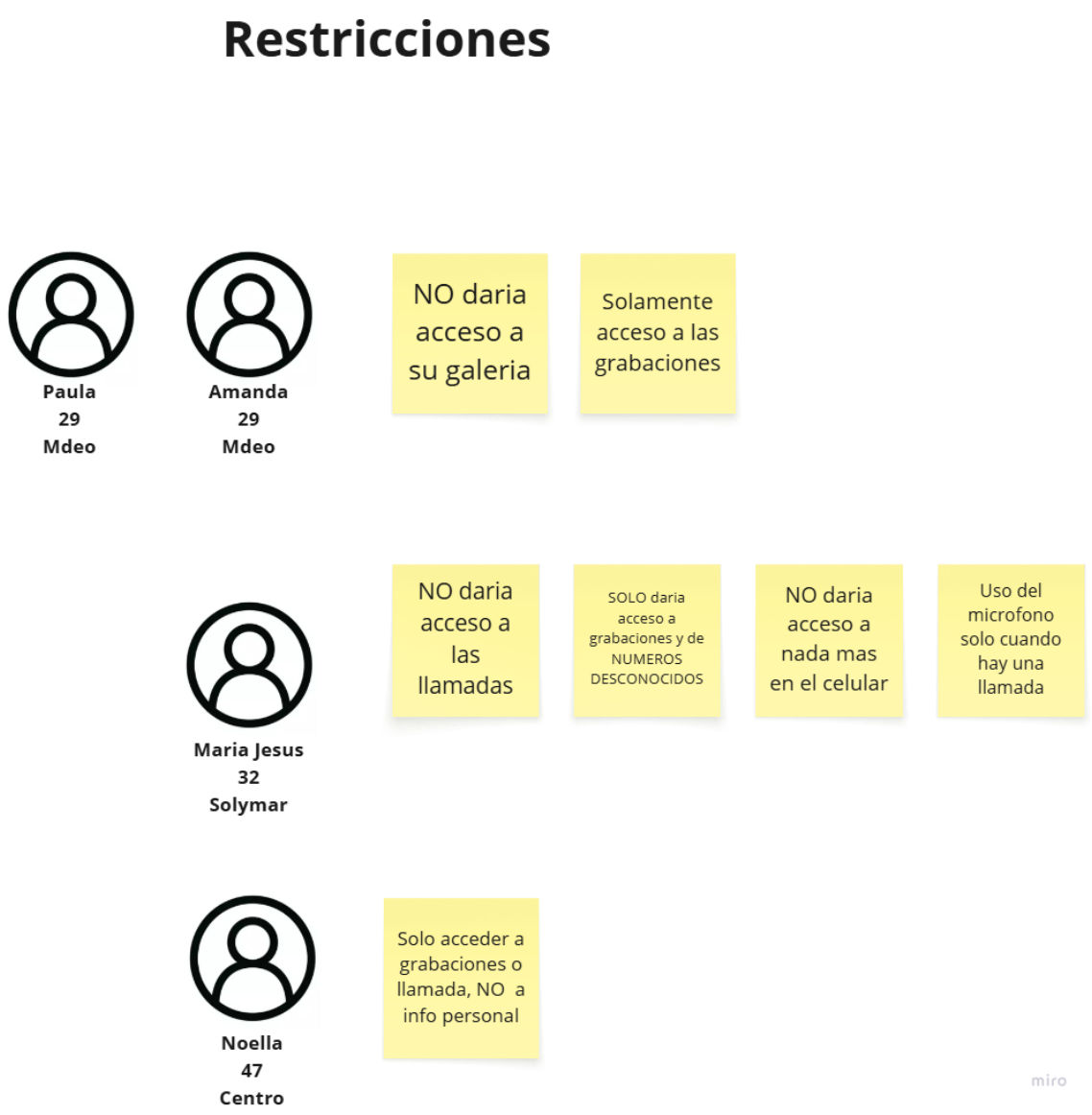


Ilustración 51 - Agrupación por restricciones de solución

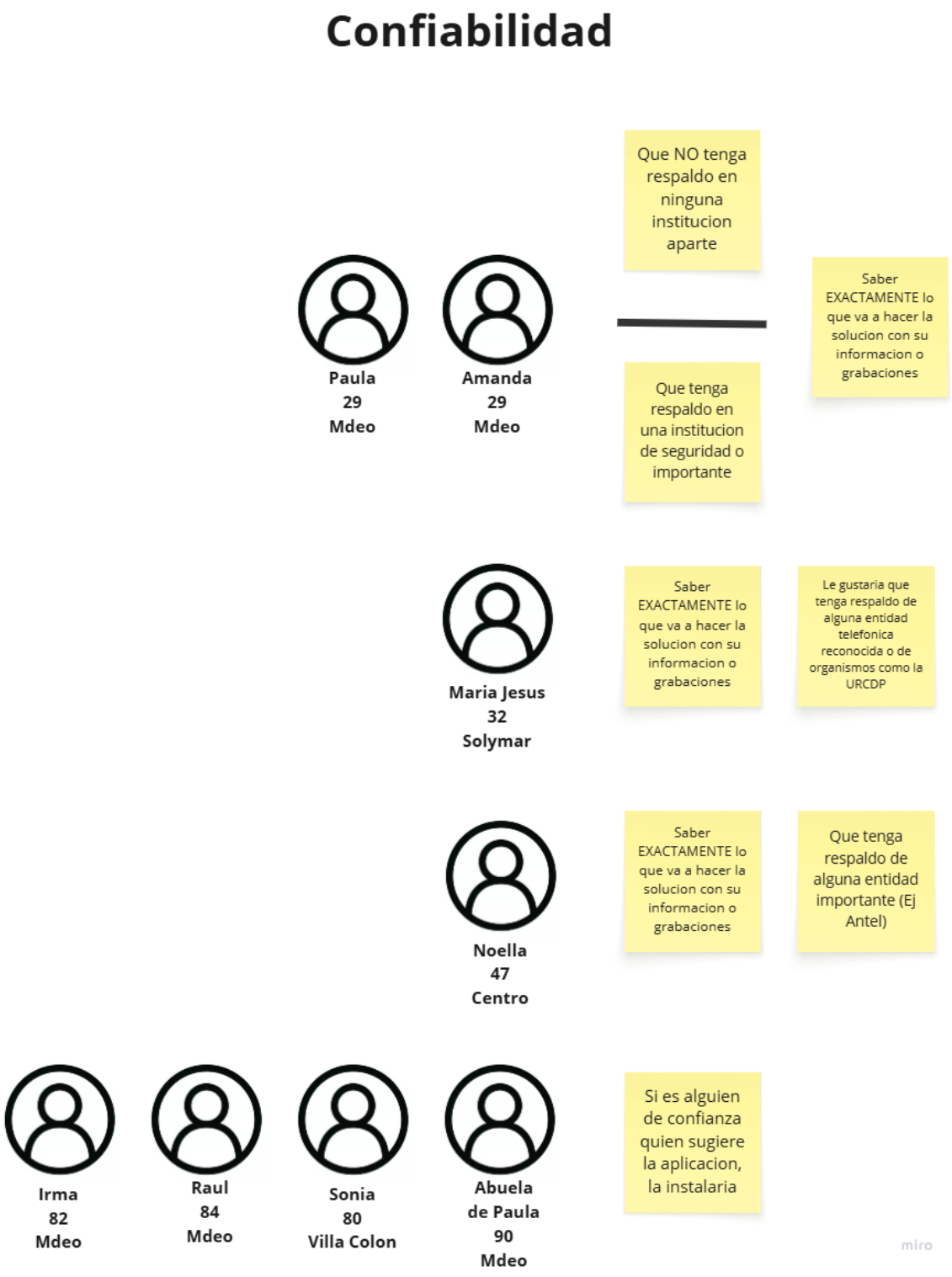


Ilustración 52 - Agrupación por confiabilidad de solución

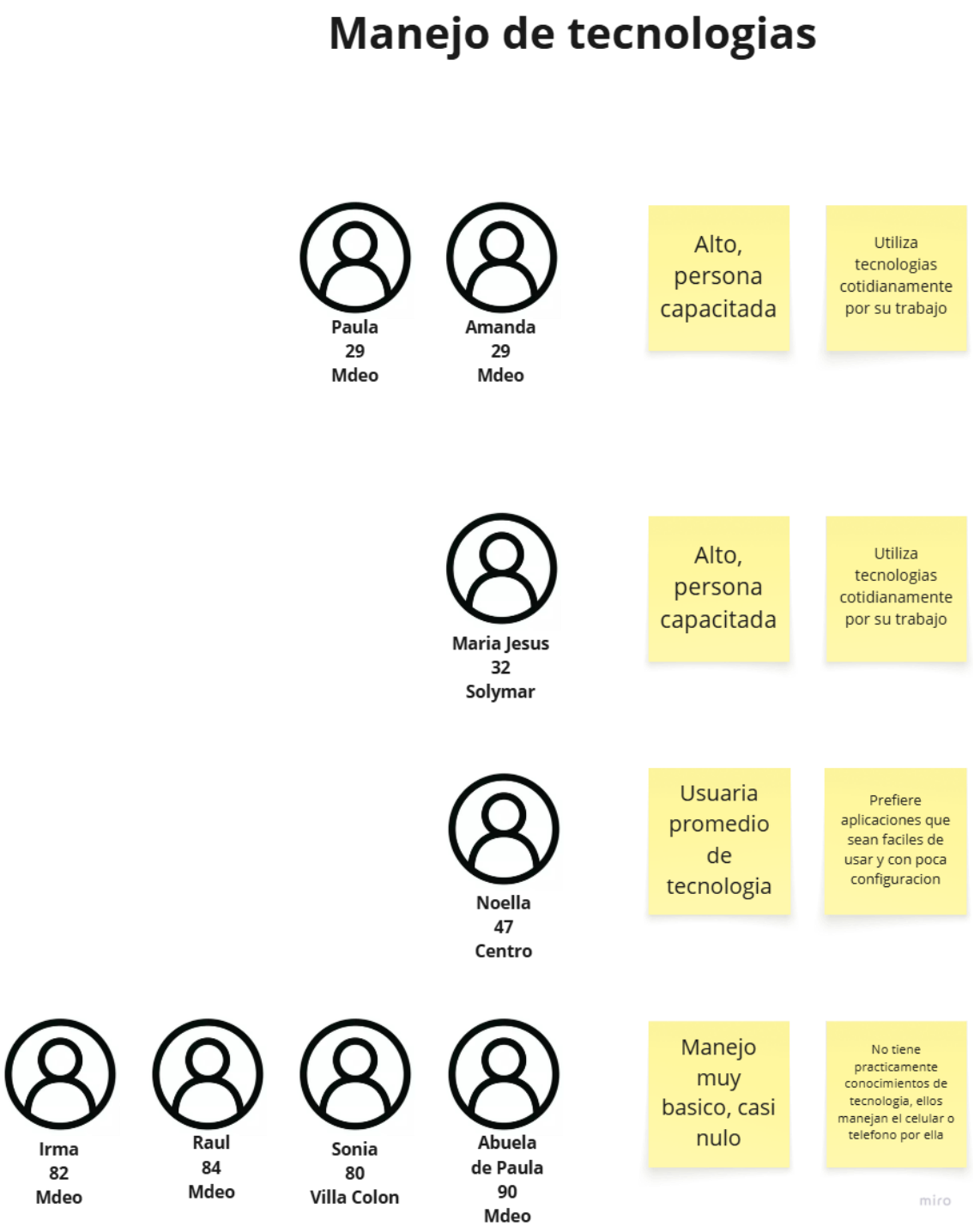


Ilustración 53 - Agrupación por manejo de tecnologías

Derivación de Perfiles

Perfil 1:

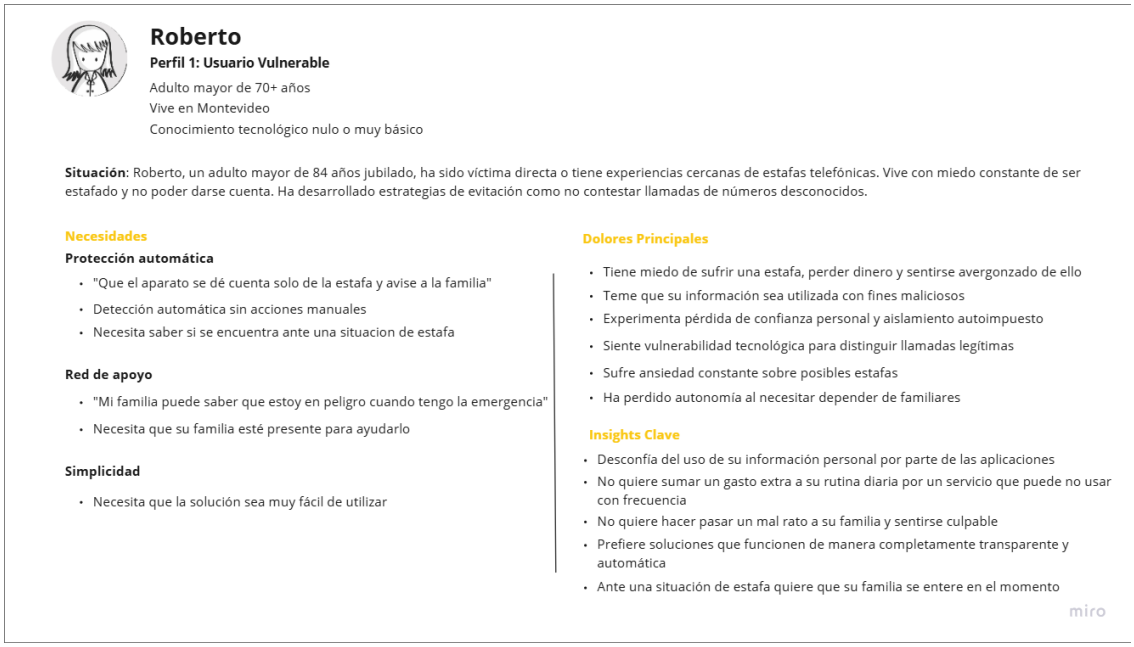


Ilustración 54 - Descripción de Perfil 1

Perfil 2:

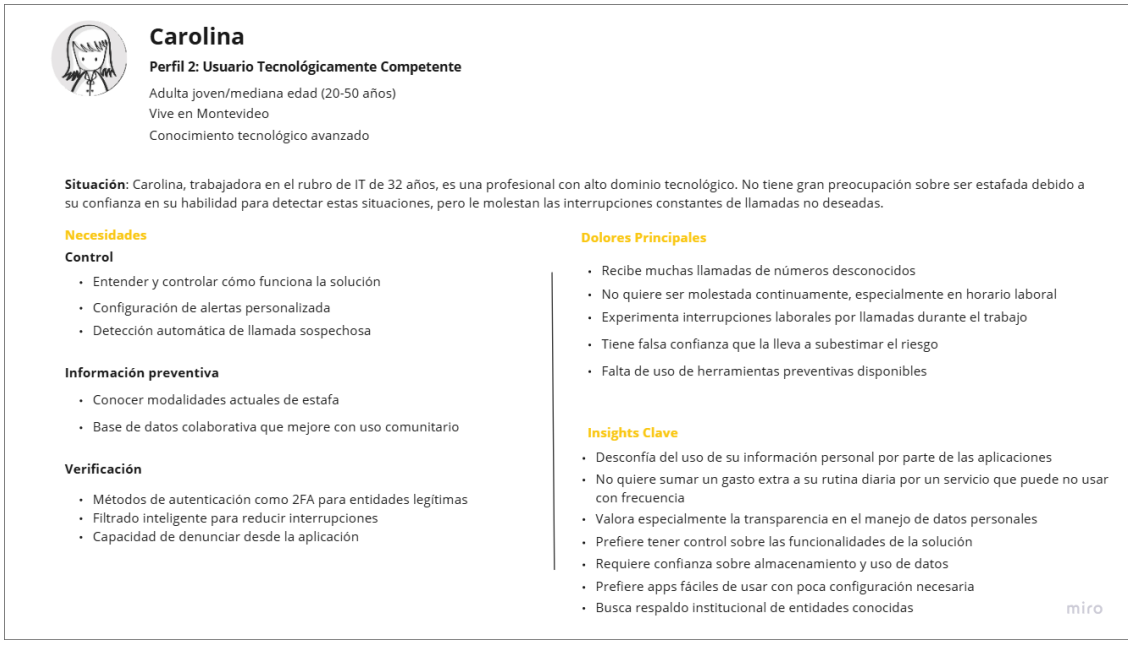


Ilustración 55 - Descripción de Perfil 2

Ideación – Brainstorming

Para comenzar con este proceso, se planteó la siguiente pregunta en formato HMW (*How Might We*):

¿Cómo podemos ayudar a evitar estafas telefónicas de números desconocidos, sean locales o extranjeros, para adultos de todas las edades, con conocimientos básicos o avanzados de tecnología, residentes en Uruguay, considerando las pocas medidas de protección que se tienen en cuenta actualmente y las siguientes modalidades de estafa: sorteos o premios y simulación de llamada de entidad bancaria?

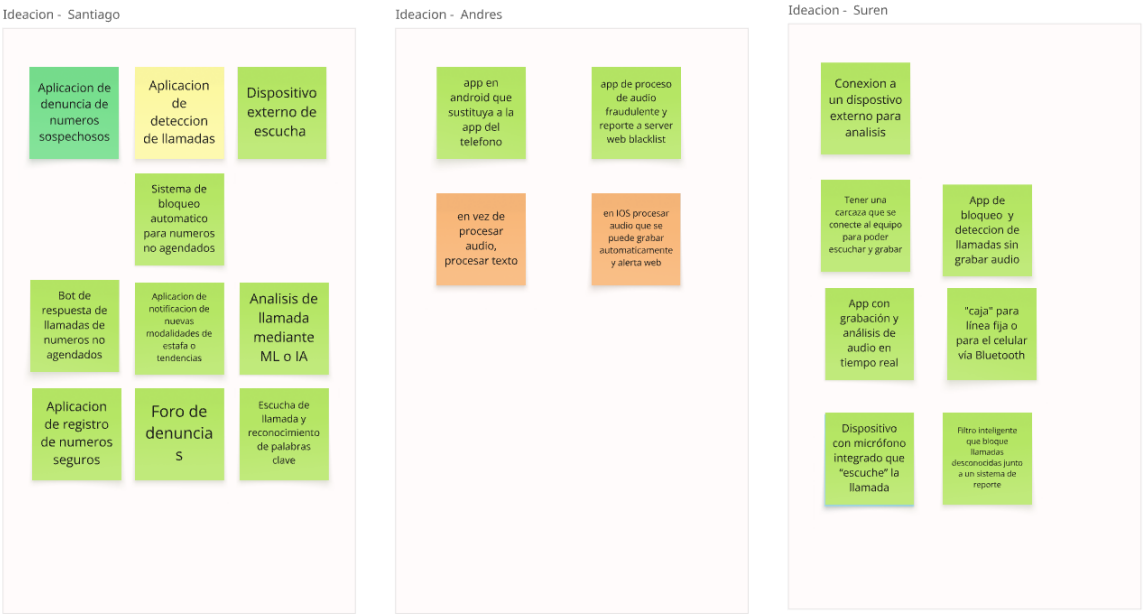


Ilustración 56 - Ideación primera ronda

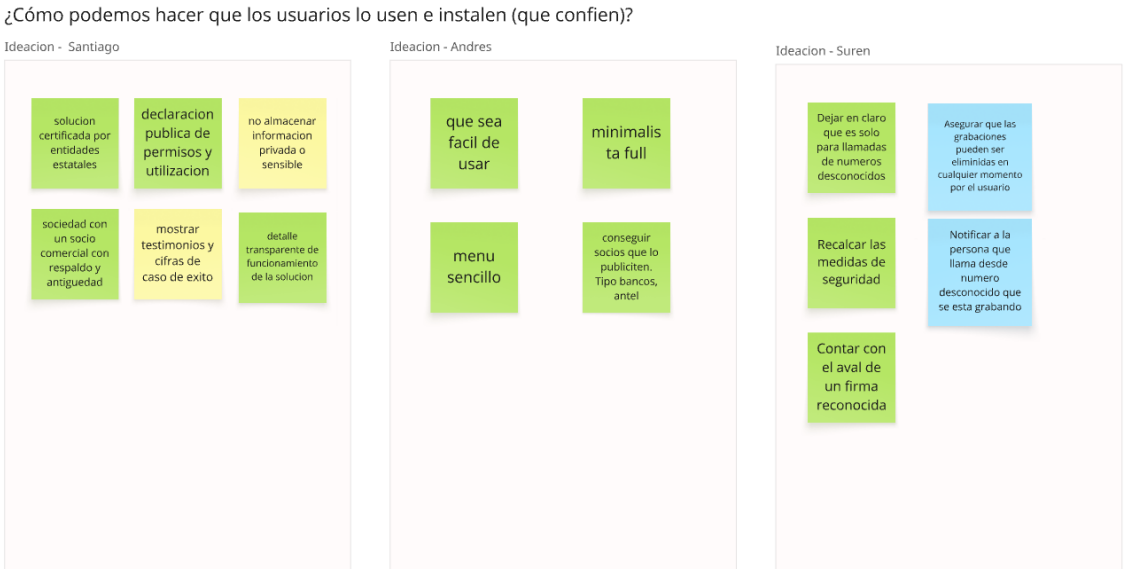


Ilustración 57 - Ideación segunda ronda

Ilustración 58 - Ideación tercera ronda

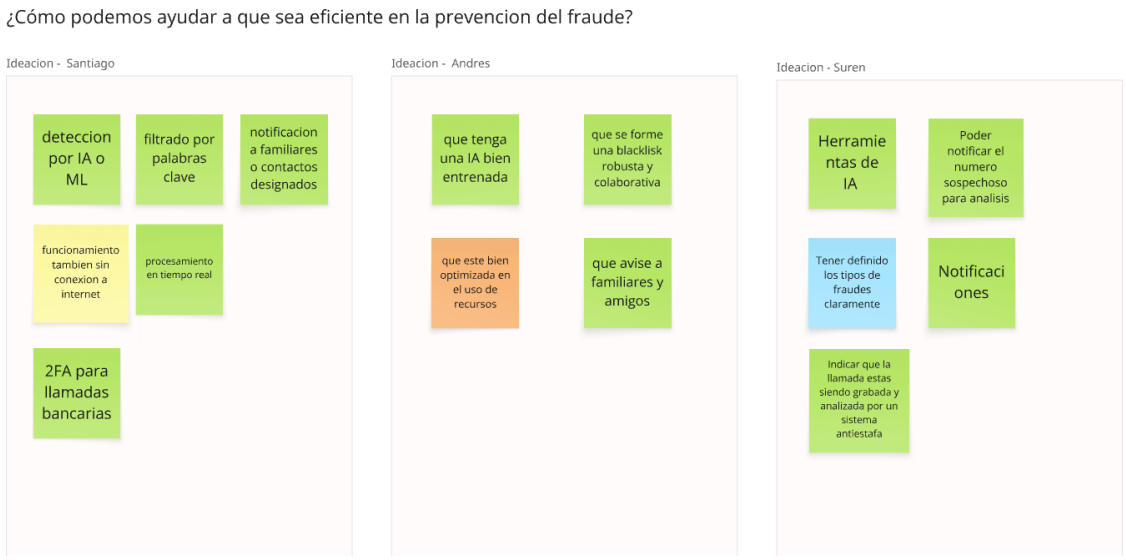


Ilustración 59 - Ideación cuarta ronda

Ideación – Agrupación de ideas

Por posible tipo de solución:

Ilustración 60 - Agrupación de ideas por posibles tipos de solución

Características compartidas:

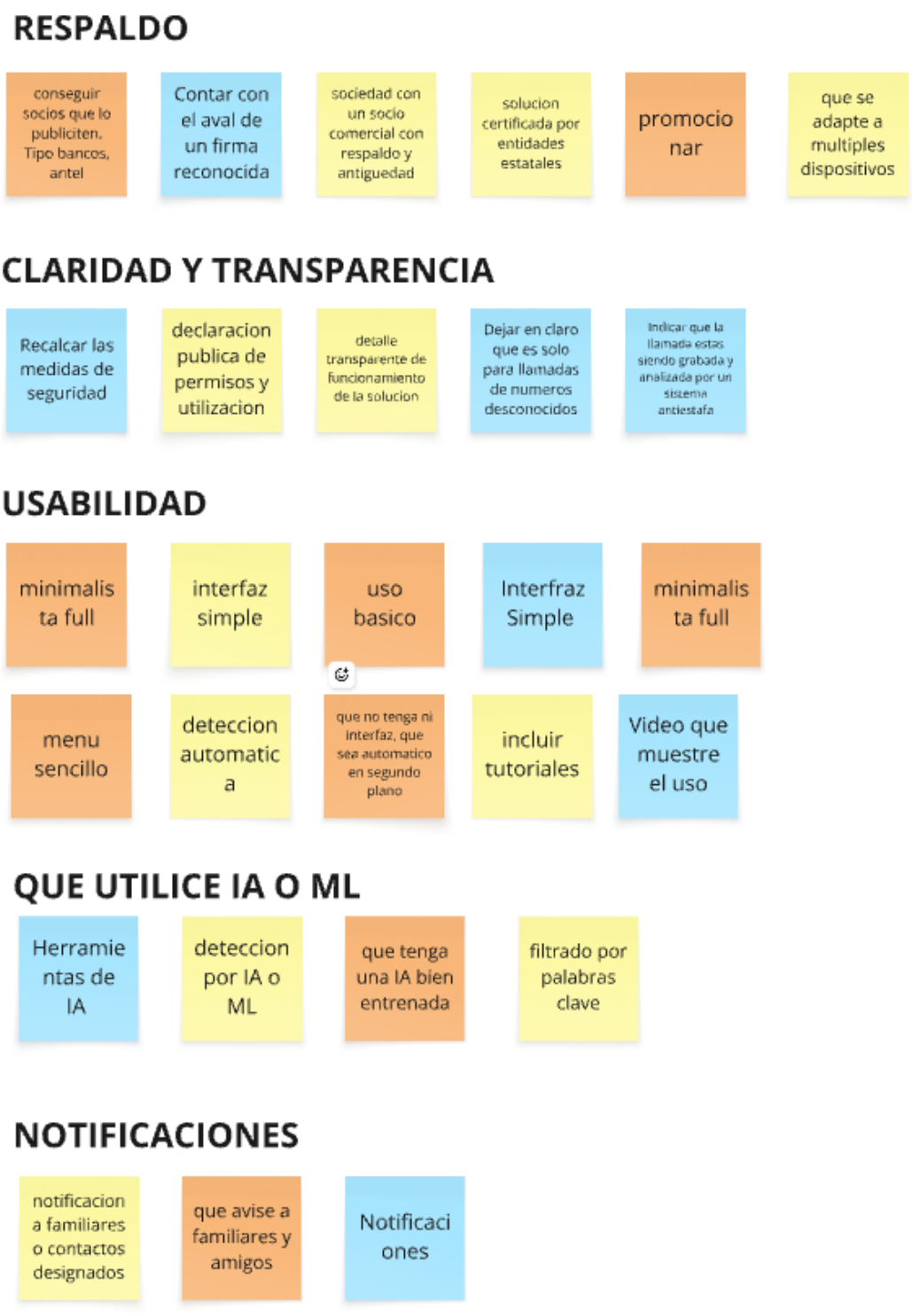


Ilustración 61 - Agrupación de características compartidas

Comparativa de Conceptos mediante “Matriz de Pugh”

Criterio	Concepto 1	Concepto 2	Concepto 3
Detección del Fraude	+	+	+
Velocidad de Detección (Real Time)	+	-	+
Facilidad de Uso	+	+	-
Seguridad	S	+	S
Factibilidad	-	-	S
Costo de Desarrollo	-	+	-
Denuncia de Número	+	S	+
Notificación de Fraude	+	+	+
TOTAL +	**5**	**5**	**4**
TOTAL -	**2**	**2**	**2**
TOTAL	**3**	**3**	**2**

Tabla 25 - Comparativa de conceptos con Matriz de Pugh

Nota: (+) Mejor, (S) Similar, (-) Peor

Prototipos realizados

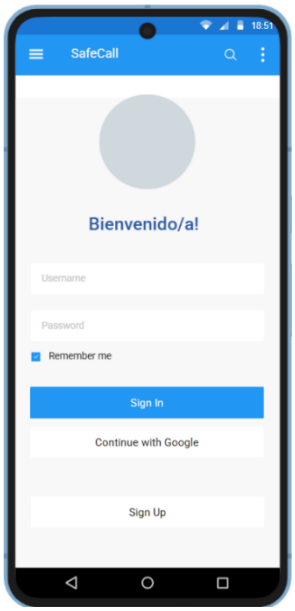


Ilustración 62 - Pantalla de Inicio

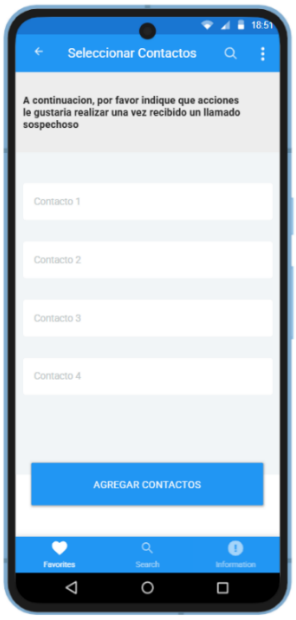


Ilustración 63 - Pantalla Selección de Contactos

Ilustración 64 - Pantalla Principal

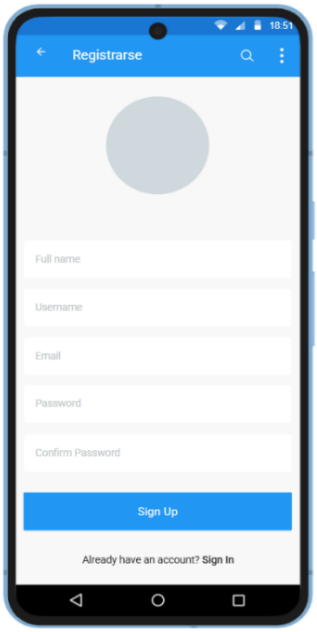


Ilustración 65 - Pantalla Registro

Ilustración 66 - Pantalla Detección Estafa

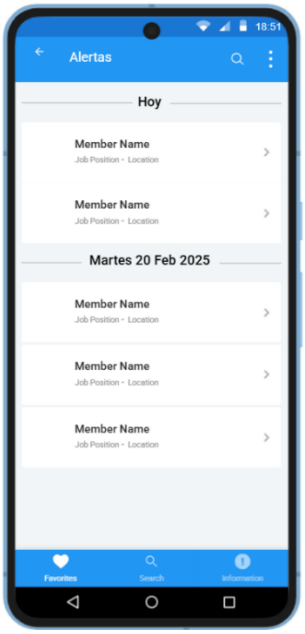


Ilustración 67 - Pantalla Alertas

Ilustración 68 - Pantalla Configuración Alerta

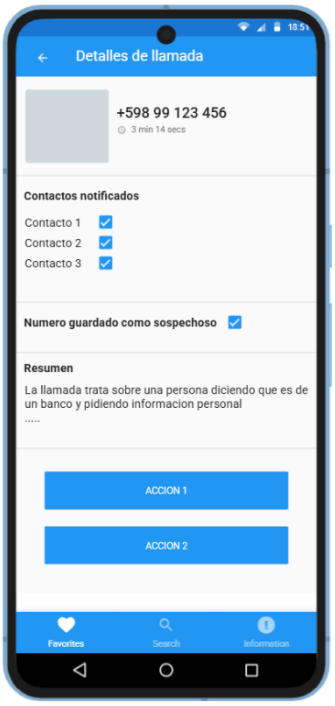


Ilustración 69 - Pantalla Detalle de Llamada

Resultados del Product Discovery

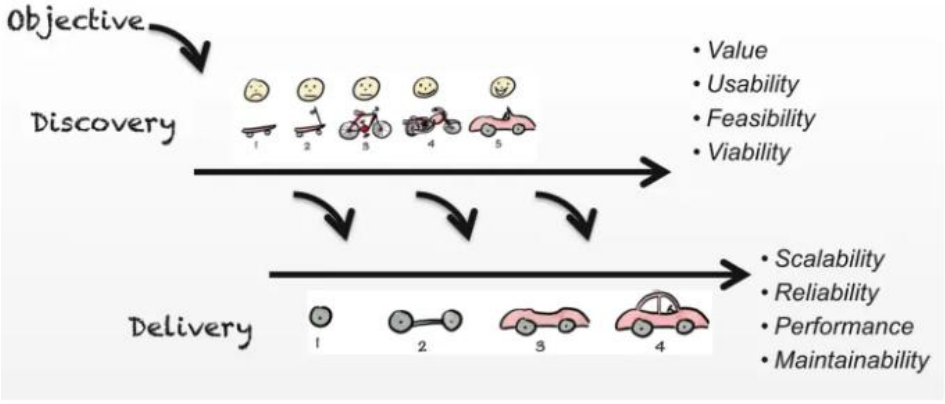
Se presenta la síntesis de los resultados, asignando a cada uno un identificador único (H01, H02, etc.) que es utilizado como referencia directa en la especificación de requerimientos funcionales. Esta estructura permite establecer una justificación desde los datos de investigación hasta las decisiones de diseño del producto.

ID	Hallazgo	Fuente	Fase del Design Thinking
H01	El 65.7% de los encuestados reportó haber recibido al menos una llamada telefónica sospechosa de estafa en los últimos 12 meses	Encuesta	Empatía (Investigación Primaria)
H02	El 70% de los usuarios encuestados valoró como "muy útil" o "extremadamente útil" contar con una herramienta de identificación automática de llamadas fraudulentas	Encuesta	Empatía (Investigación Primaria)
H03	Los usuarios expresaron la necesidad de poder alertar inmediatamente a familiares o personas de confianza cuando reciben llamadas sospechosas, especialmente en el caso de adultos mayores	Entrevistas cualitativas con usuarios/víctimas	Empatía (Investigación Primaria)
H04	Expertos en seguridad recomendaron implementar un sistema de lista negra colaborativa que permita compartir números reportados entre usuarios para fortalecer la detección comunitaria	Entrevistas con expertos (INTERPOL, URCDP)	Empatía (Investigación Primaria)

H05	Durante la validación inicial del concepto, los usuarios calificaron la utilidad percibida de la solución con un promedio de 4.5/5, confirmando la pertinencia del enfoque propuesto	Pruebas de usabilidad con usuarios	Prototipación y Pruebas - Validación
H06	Los usuarios solicitaron durante las pruebas de validación poder gestionar una lista personalizada de contactos de seguridad que recibirán notificaciones automáticas	Pruebas de usabilidad con usuarios	Prototipación y Pruebas - Validación
H07	Los usuarios manifestaron la necesidad de poder bloquear números telefónicos identificados como problemáticos para evitar futuras llamadas no deseadas	Entrevistas cualitativas con usuarios/víctimas	Empatía (Investigación Primaria)
H08	Expertos sugirieron facilitar el proceso de denuncia formal de números fraudulentos directamente desde la aplicación, para incrementar las tasas de reporte ante autoridades	Entrevistas con expertos (operadores móviles, URCDP)	Empatía (Investigación Primaria)
H09	Los usuarios evaluaron la claridad de la interfaz propuesta con un promedio de 4.3/5 durante las pruebas de validación, identificando oportunidades de mejora en la presentación de información	Pruebas de usabilidad con usuarios	Prototipación y Pruebas - Validación

Tabla 26 - Resultados Product Discovery

ANEXO 2



Discovery vs. Delivery. Source: https://www.svpg.com/skateboards-vs-cars-revisited/

Ilustración 70 - Proceso Dual Discovery-Delivery

2.1 Tabla de RNF

ID	Descripción	Justificación	Atributo de Calidad
RNF01	Tiempode análisis de llamada ≤ 30 segundos	Umbral que permite actuar rápidamente ante intentos de estafa que necesitan concretarse después de la llamada (no inmediatos)	Performance
RNF02	Interfaz intuitiva con texto descriptivo	Derivado de entrevistas con Perfil 1 (adultos mayores vulnerables) que requiere soluciones que funcionen de manera completamente transparente sin requerir aprendizaje complejo	Usabilidad
RNF03	Compatibilidad Android versión superior a 11	Limitación técnica por políticas de privacidad de Android	Performance

RNF04	Precisión mínima del 80% en detección de estafas	Balance entre viabilidad técnica y aporte de valor para un MVP	Performance
RNF05	Acceso limitado solo a llamadas grabadas	Responde a la prohibición de acceso a información no relacionada con llamadas expresada categóricamente por ambos perfiles.	Seguridad
RNF06	Funcionamiento sin registro de usuario	Elimina necesidad de guardar información de los usuarios cuidando la privacidad y seguridad	Privacidad
RNF07	Configuración inicial con pasos guiados	Responde a las características demográficas de los usuarios y a las limitaciones de conocimientos tecnológicos que poseen	Usabilidad
RNF08	Notificaciones con colores contrastantes y texto visible	Considera el contexto de vulnerabilidad y nerviosismo durante situaciones de estafa y las limitaciones físicas.	Usabilidad
RNF09	Despliegues reproducibles y consistentes mediante contenedores	Uso de contenedores Docker para garantizar que el sistema pueda desplegarse de manera confiable en distintos entornos	Deployabilidad

RNF10	Soporte para integración y entrega continuas (CI/CD)	Permite actualizaciones rápidas y controladas, incorporando mejoras sin afectar a usuarios existentes	Deployabilidad

Tabla 27 - Requerimientos No Funcionales

2.2 Tabla de RNF para una solución en producción.

ID	Descripción	Justificación	Atributo de Calidad
RNFP01	El sistema debe implementar cifrado end-to-end y controles de acceso para proteger datos personales y grabaciones de llamadas	Manejo de información altamente sensible que requiere cumplimiento de normativas de protección de datos (URCDP, GDPR), según Ley 18331	Seguridad, Privacidad
RNFP02	El sistema debe garantizar disponibilidad operativa ≥ 99.5% mediante arquitectura redundante y monitoreo continuo	La efectividad depende de que el servicio esté siempre disponible. Una caída durante una llamada fraudulenta anula el valor protector	Disponibilidad
RNFP03	El sistema debe procesar la transcripción de audio en servidores propios sin enviar datos a APIs de terceros	Evita exposición de contenido de llamadas a servicios externos (OpenAI API, Google Cloud). Mantiene control total sobre datos	Privacidad, Seguridad,

		sensibles en infraestructura propia	
RNFP04	El sistema debe anonimizar automáticamente datos sensibles (nombres, direcciones, números de cuenta) en registros y logs antes de almacenarlos	Previene fugas de información personal identificable (PII) en caso de brecha de seguridad o acceso no autorizado a logs del sistema	Privacidad, Seguridad,

Tabla 28 - Requerimientos No Funcionales en Produccion

ANEXO 3

Este anexo complementa el Capítulo 5 con detalles técnicos sobre el proceso de experimentación, selección y validación del modelo de inteligencia artificial utilizado en SafeCall. La información aquí presentada profundiza aspectos conceptuales que, si bien son relevantes para comprender el rigor metodológico aplicado, podrían interrumpir la fluidez narrativa del cuerpo principal de la tesis.

3.1. Arquitectura del Dataset de Entrenamiento

Licencia

Nombre del dataset: Call Transcripts Scam Determinations

Autor / Fuente: Plataforma Kaggle – comunidad abierta de aprendizaje automático y ciencia de datos.

URL: https://www.kaggle.com/datasets

Licencia: CC0: Public Domain

Uso libre, sin restricciones. El *dataset* puede ser copiado, modificado, distribuido y utilizado incluso con fines comerciales, siempre que se mantenga la atribución original.

Estructura general

El *dataset* contiene 650 registros, cada uno correspondiente a un paso dentro de una conversación. Está conformado por las siguientes columnas:

Columna	Tipo	Descripción
`CONVERSATION_ID`	int64	Identificador único de conversación. Hay 65 conversaciones distintas.
`CONVERSATION_STEP`	int64	Paso secuencial dentro de la conversación. Hasta 17 pasos por conversación.

TEXT	object	Contenido textual de cada intervención.
CONTEXT	object	Contexto resumido o funcional de la línea. Tiene 3 valores faltantes.
LABEL	object	Etiqueta asociada a la intervención. Pueden ser categorías como "scam", "neutral", etc.
FEATURES	object	Características estilísticas del mensaje. Ausentes en 79 casos (~12%).
ANNOTATIONS	object	Anotaciones específicas que marcan estrategias o señales. Faltan en 395 casos (~61%).

Tabla 29 - Etiquetas del Dataset

Calidad del contenido

Valores faltantes:

- CONTEXT: 3 ausencias.
- FEATURES: 79 (12.2%).
- ANNOTATIONS: 395 (60.8%).

Distribución de longitud del texto (TEXT):

- Promedio: 140.8 caracteres
- Rango: de 9 a 347 caracteres
- Mediana: 135
- El 75% de los textos tiene menos de 180 caracteres, lo que indica intervenciones de longitud moderada, propias de conversaciones naturales.

Etiquetas (LABEL)

Hay 21 valores distintos, aunque algunos están normalizados por errores de formato (espacios o mayúsculas). Ejemplos:

Etiqueta	**Frecuencia**
`neutral` (con y sin espacios)	158 + 35
`scam` (con variantes)	138 + 82
`scam_response` (con variantes)	117 + 3
`legitimate` (con variantes)	43 + 14
`suspicious`	32
Otros (minoría)	≤ 8

Tabla 30 - Frecuencia de etiquetas

Observación: es recomendable normalizar estos valores para evitar ambigüedades (ej. remover espacios y unificar mayúsculas/minúsculas). En este proceso de *fine-tuning* se normalizan.

Características estilísticas

Presentes en un 87.8% de los ejemplos.

Hay 457 valores distintos, lo que sugiere una taxonomía rica pero también muy dispersa.

Ejemplos comunes: *welcoming, inclusive, positive_tone, offers_options, prompt_action*.

Anotaciones

- Sólo presentes en el 39.2% de los casos.
- Existen 242 valores únicos, indicando un sistema de etiquetado detallado para marcar indicios de estafa u otros patrones relevantes.

- Algunas anotaciones frecuentes:
- "urgency_claim" (4 veces)
- "call_termination", "documentation_promise" (3 o 2 veces)
- Indicadores de presión, amenazas o requerimientos sospechosos.

Estas anotaciones son importantes para el entrenamiento supervisado en tareas de detección de fraude, ya que representan señales semánticas específicas.

Distribución por clase

La distribución de etiquetas está levemente desbalanceada:

- Clases como neutral, scam y scam_response dominan el dataset.
- Clases minoritarias como polite_ending o standard_opening son anecdóticas y podrían no tener valor predictivo fuerte.
- Esto justifica el uso de F1-score como métrica principal, dado que accuracy podría ser engañosa en un dataset desbalanceado.

Observaciones adicionales

- Contexto conversacional: La estructura CONVERSATION_ID y CONVERSATION_STEP permite reconstruir el flujo conversacional, lo que abre la posibilidad de usar modelos basados en contexto o arquitecturas como Transformers conversacionales.
- Riqueza semántica: Los campos FEATURES y ANNOTATIONS enriquecen el dataset más allá del texto crudo, permitiendo análisis multicapas: léxico, estilístico y estratégico.
- Potencial de mejora: una limpieza y consolidación de etiquetas redundantes mejoraría la robustez del entrenamiento. Además, rellenar o estandarizar FEATURES y ANNOTATIONS podría facilitar tareas de clasificación multi-etiqueta o interpretabilidad.

3.2. Estrategia de Experimentación con Hiperparámetros

Para identificar la configuración óptima del modelo de detección, se realizó un proceso sistemático de experimentación con 20 variaciones de hiperparámetros sobre el modelo base DistilBERT. Este enfoque metódico permitió evaluar el impacto de cada parámetro en el rendimiento final.

Los hiperparámetros son configuraciones que definen cómo aprende el modelo, no qué aprende. Funcionan como "perillas de ajuste" que controlan:

Epochs (épocas): Cuántas veces el modelo revisa todos los datos de entrenamiento. Pocas épocas = aprendizaje insuficiente; muchas épocas = riesgo de memorización.

Batch size (tamaño de lote): Cuántos ejemplos procesa simultáneamente. Lotes pequeños = mayor variabilidad; lotes grandes = mayor estabilidad.

Learning rate (tasa de aprendizaje): Qué tan grandes son los "pasos" que da el modelo al aprender. Muy alta = inestabilidad; muy baja = lentitud.

Max length (longitud máxima): Cuántas palabras procesa en cada texto. Más contexto vs. más recursos computacionales.

Tabla de experimentos extendida

EXP	Objetivo	epochs	batch_size	learning_rate	max_length	Comentario
1	Baseline	4	8	2e-5	128	Configuración base para comparar otros resultados.
2	Aumentar cantidad de épocas	8	8	2e-5	128	Más épocas permiten mayor aprendizaje, pero pueden sobreajustar.
3	Reducir cantidad de épocas	2	8	2e-5	128	Menos épocas aceleran el entrenamiento, pero pueden generar underfitting.
4	Aumentar tamaño de batch	4	16	2e-5	128	Batches grandes estabilizan el gradiente pero requieren más memoria.
5	Disminuir tamaño de batch	4	4	2e-5	128	Batches chicos introducen más ruido pero pueden mejorar la generalización.

6	Learning rate más alto	4	8	5e-5	128	Puede acelerar el entrenamiento, pero arriesga inestabilidad.
7	Learning rate más bajo	4	8	1e-5	128	Más preciso, pero más lento. Puede necesitar más épocas.
8	Secuencia de entrada más corta	4	8	2e-5	128	Reduce tiempo y memoria, pero puede perder contexto relevante.
9	Secuencia de entrada más larga	4	8	2e-5	64	Captura más contexto, útil para textos largos, pero más costoso.
10	Variación combinada: +épocas, +batch, +LR, +longitud	8	16	3e-5	256	Busca mejorar todo a la vez, pero puede dificultar análisis fino.
11	Batch muy grande + learning rate	4	32	1e-5	128	Estabilidad con bajo riesgo de sobreajuste, pero alto uso de

	bajo					RAM.
12	Batch muy chico + learning rate alto	4	2	5e-5	128	Muy ruidoso, puede ser útil para evitar mínimos locales.
13	Pocas épocas + secuencia larga	2	8	2e-5	256	Poco tiempo de aprendizaje con mucho contexto; puede no aprovecharse bien.
14	Muchas épocas + secuencia corta	12	8	2e-5	64	Riesgo de sobreajuste por poca variabilidad de entrada
15	Valores intermedios en todos los parámetros	6	12	3e-5	128	Equilibrio entre costo y performance. Útil como candidato final.
16	Mismo que baseline pero cambiando semilla (seed)	4	8	2e-5	128	Prueba de reproducibilidad y robustez frente a aleatoriedad.

17	Muchas épocas + learning rate muy bajo	12	8	5e-6	128	Aprendizaje muy fino; ideal si se observa sobreajuste en otras configs.
18	Batch grande + secuencia larga + LR bajo	6	32	1e-5	256	Entrenamiento costoso pero potencialmente muy preciso.
19	Batch chico + secuencia corta + LR alto	6	4	5e-5	64	Entrenamiento ruidoso y rápido; útil para explorar límites.
20	Combinación extrema: muchas épocas, batch grande, secuencia larga, LR bajo	16	32	1e-5	256	Configuración intensiva, busca el máximo potencial del modelo.

Tabla 31 - Resultados de Experimentos

1. Experimento 1 – Baseline

Este experimento establece la configuración base: 4 épocas, *batch size* 8, *learning rate* 2e-5 y una longitud máxima de secuencia de 128. Sirve como punto de comparación para todos los demás. La elección de estos valores es razonable para tareas de clasificación textual moderadas, sin ser ni muy exigentes ni muy superficiales.

2. Experimento 2 – Más épocas

Se duplican las épocas (8 vs. 4) respecto al *baseline*, manteniendo todo lo demás igual. Este experimento busca comprobar si el modelo mejora con más iteraciones de entrenamiento o si empieza a sobreajustar. También permite observar la evolución del loss en función del tiempo de exposición a los datos.

3. Experimento 3 – Menos épocas

Se reducen las épocas a 2, lo cual debería disminuir el tiempo de entrenamiento significativamente. Se espera que el modelo no logre aprender del todo (*underfitting*), lo que lo convierte en una prueba de mínima capacidad de aprendizaje.

4. Experimento 4 – Batch más grande

Aquí se duplica el *batch* a 16. Esto puede estabilizar el gradiente y reducir el ruido en el entrenamiento, pero puede dificultar la generalización si el modelo se vuelve "perezoso" al ver muchas muestras por paso. También pone a prueba la capacidad de procesamiento de la GPU.

5. Experimento 5 – Batch más pequeño

Se reduce el *batch* a 4. Los lotes más chicos introducen mayor ruido en el gradiente, lo que puede favorecer la exploración de distintos mínimos. Es útil para tareas donde se quiere evitar quedar atrapado en mínimos locales.

6. Experimento 6 – Learning rate más alto

Sube el LR a 5e-5. Esto debería acelerar el aprendizaje, pero puede causar saltos erráticos y dificultad para converger. Se lo compara con el *baseline* y con el experimento 7, que hace lo contrario.

7. Experimento 7 – Learning rate más bajo

Reduce el LR a 1e-5. Es más lento para aprender, pero más preciso. Puede ser útil cuando se observa inestabilidad con tasas mayores. Se compara especialmente con los experimentos 1 y 6.

8. Experimento 8 – Secuencia más corta

Reduce el *max_length* a 64. Este valor limita cuánto texto entra al modelo, lo que reduce tiempo y memoria. Sirve para analizar si el contexto corto es suficiente o si se pierden datos.

9. Experimento 9 – Secuencia más larga

Incrementa *max_length* a 256. Ideal si las entradas son largas y se sospecha que el modelo necesita más contexto. Se espera un mayor uso de memoria y posible mejora en tareas dependientes del final de la frase.

10. Experimento 10 – Combinación de mejoras

Integra varias modificaciones: más épocas, *batch* más grande, LR intermedio (3e-5) y secuencia larga. Busca comprobar si mejoras combinadas superan la suma de partes individuales. Sirve como stress test.

11. Experimento 11 – Batch muy grande + LR bajo

Batch de 32 y LR de 1e-5. Es un *setup* conservador y estable. La idea es sacrificar velocidad por estabilidad. Útil para tareas donde se quiera minimizar el sobreajuste.

12. Experimento 12 – Batch muy chico + LR alto

Batch de 2 y LR de 5e-5. Esta es una configuración agresiva: más ruido y más velocidad. Se espera alta varianza, por lo que puede ser útil para evaluar robustez del modelo a condiciones extremas.

13. Experimento 13 – Pocas épocas + secuencia larga

Se combinan solo dos cambios: pocas épocas (2) y secuencia larga (256). Se quiere analizar si tener más contexto compensa el poco tiempo de entrenamiento.

14. Experimento 14 – Muchas épocas + secuencia corta

Doce épocas con secuencia corta (64). Lo opuesto al anterior: se entrena mucho, pero con contexto limitado. Es útil para ver si el modelo puede "compensar" la falta de longitud con exposición repetida.

15. Experimento 15 – Parámetros intermedios

Valores medios en todo: *batch* 12, épocas 6, LR 3e-5. Busca ser una alternativa al *baseline* con un poco más de ambición. Se espera mejor rendimiento sin sobrecargar recursos.

16. Experimento 16 – Cambiar seed (semilla aleatoria)

Misma configuración que el *baseline*, pero con una semilla explícita (--seed 123). Evalúa la estabilidad del resultado ante cambios aleatorios. Importante para evaluar reproducibilidad.

17. Experimento 17 – Muchas épocas + LR muy bajo

Doce épocas con LR de 5e-6. Busca un entrenamiento muy fino y progresivo, útil para evitar sobreajuste y converger lentamente. Suele usarse cuando ya se tiene un modelo razonable y se quiere pulir.

18. Experimento 18 – Batch grande + secuencia larga + LR bajo

Es una configuración exigente: *batch* 32, secuencia 256, LR 1e-5. Se espera un entrenamiento estable y de alta calidad, pero costoso en tiempo y memoria. Ideal para *setups* potentes.

19. Experimento 19 – Batch chico + secuencia corta + LR alto

Pequeño *batch*, texto corto y alta tasa de aprendizaje. Contrapone velocidad y riesgo. Puede fallar o sorprender. Se incluye como prueba de comportamiento en el "peor escenario".

20. Experimento 20 – Configuración extrema de máxima exigencia

Este experimento combina lo más ambicioso: 16 épocas, batch 32, secuencia 256 y LR bajo. Busca el mejor rendimiento posible con máxima carga computacional. Es útil como caso de cierre para ver si vale la pena el costo.

3.3. Métricas

Resultados Visuales

F1-Score por Experimento

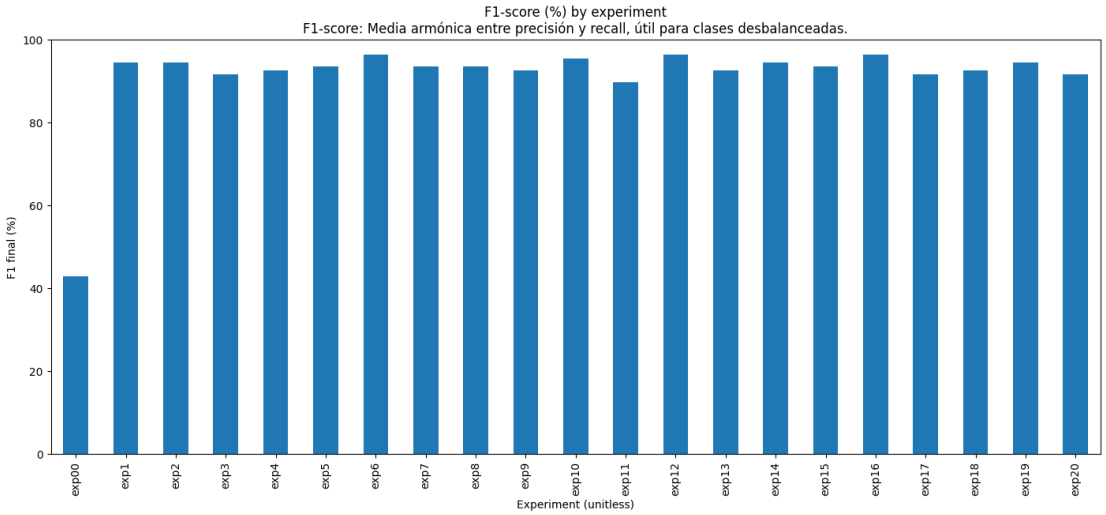


Ilustración 71 - F1-Score por Experimento

El *F1-score* se utiliza como métrica principal por ser especialmente adecuada en contextos con clases desbalanceadas, como ocurre típicamente en detección de fraude o *spam*. Combina precisión y *recall* mediante su media armónica, penalizando tanto los falsos positivos (por ejemplo, marcar como estafa un mensaje legítimo) como los falsos negativos (no detectar una estafa real). Se calcula mediante:

$$F1 = 2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}}$$

Precisión por Experimento

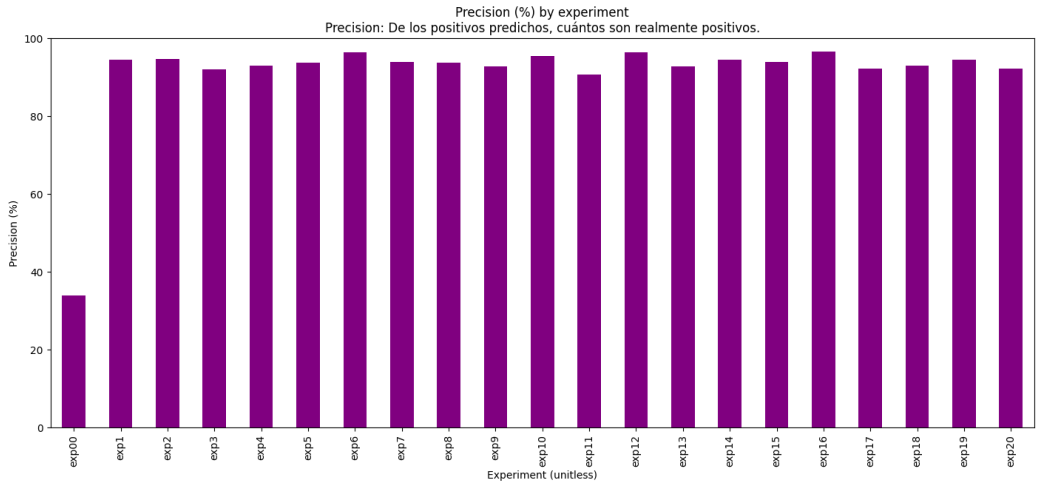


Ilustración 72 - Precisión por experimento

Además del *F1-score*, es relevante analizar métricas que describen distintos aspectos del rendimiento del modelo. En esta sección se estudia la métrica de Precisión, que mide la proporción de verdaderos positivos sobre el total de predicciones positivas realizadas. Su fórmula es:

$$Precision = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

En contextos de detección de estafas, una alta precisión indica que los mensajes clasificados como fraudulentos lo son efectivamente, lo cual importa para evitar falsos positivos (es decir, marcar mensajes legítimos como estafa).

La precisión se mantuvo consistentemente alta (>85%) en la mayoría de los experimentos, con exp15 destacándose por minimizar falsos positivos sin sacrificar detección.

Recall por Experimento

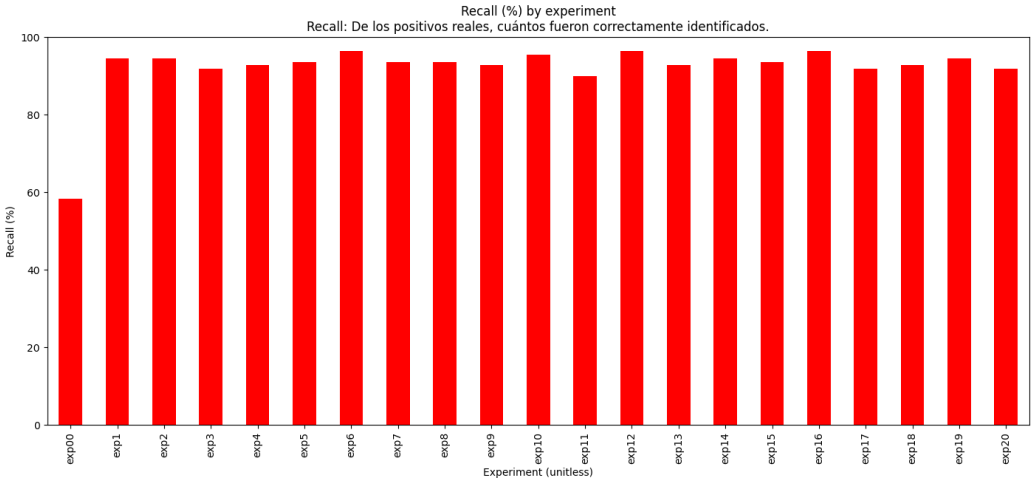


Ilustración 73 - Recall por experimento

La métrica *Recall*, también conocida como sensibilidad o tasa de verdaderos positivos, mide la capacidad del modelo para identificar correctamente los ejemplos positivos reales. En el contexto de detección de estafas telefónicas, se interpreta como el porcentaje de estafas reales que el sistema logra detectar correctamente. Su fórmula es:

$$Recall = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

Un valor alto de *recall* es fundamental para este tipo de tareas, ya que minimiza el riesgo de pasar por alto una estafa y permitir que ocurra sin ser detectada. Mostro valores superiores a 90% en las configuraciones balanceadas, confirmando que el modelo detecta la gran mayoría de estafas reales.

Accuracy por Experimento

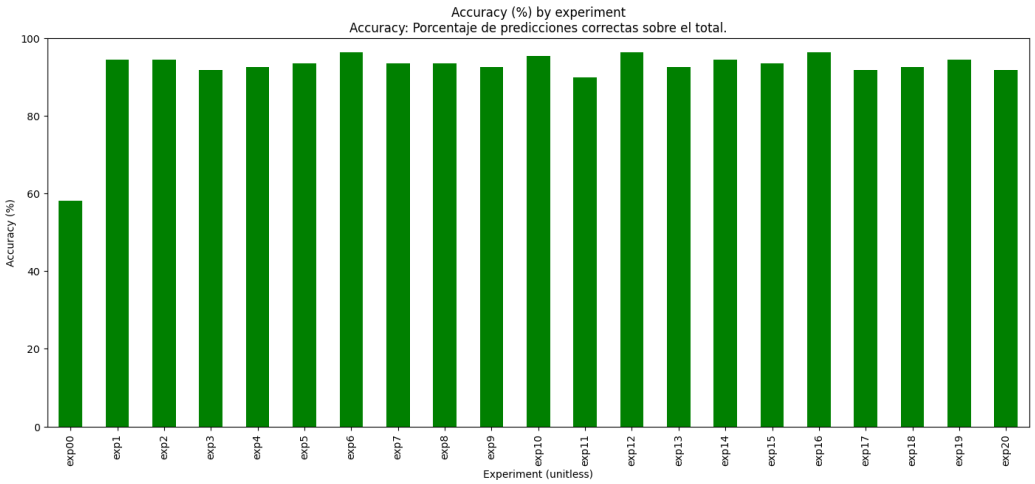


Ilustración 74 - Acurracy por experimento

La métrica *Accuracy* o exactitud indica la proporción de predicciones correctas realizadas por el modelo sobre el total de casos. Es una medida global de rendimiento, que considera tanto verdaderos positivos como verdaderos negativos. Su fórmula es:

$$Accuracy = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total de Instancias}}$$

En tareas de clasificación binaria como la detección de estafas telefónicas, esta métrica representa el porcentaje de mensajes (fraudulentos y legítimos) correctamente clasificados.

Todos los modelos ajustados superaron el 90% de *accuracy*, con mínimas variaciones entre experimentos (91-96%).

Loss (Pérdida) por Experimento

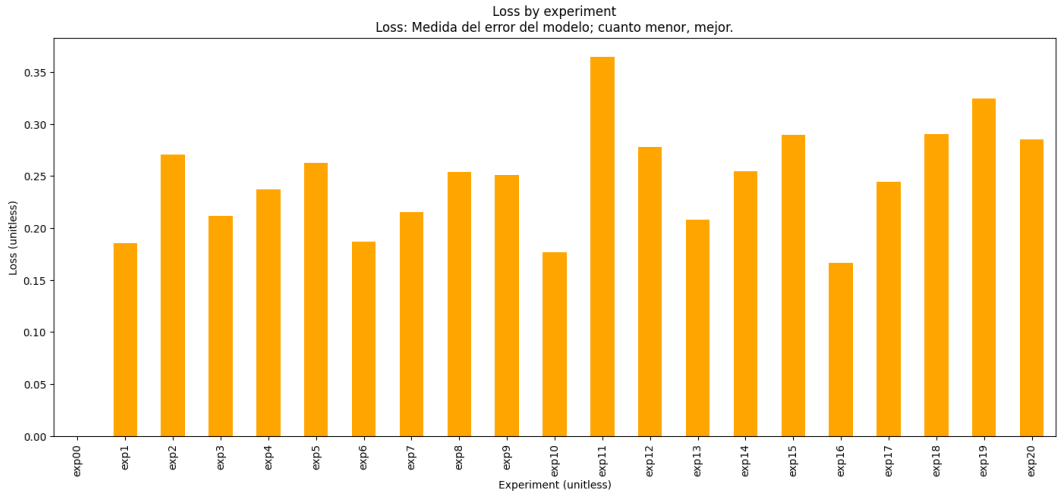


Ilustración 75 - Perdida por experimento

La métrica *Loss* (función de pérdida) es una medida numérica que cuantifica el error del modelo durante el entrenamiento o evaluación. En clasificación binaria como la presente, la función de pérdida más habitual es la Binary Cross-Entropy, que penaliza la diferencia entre las probabilidades predichas y las verdaderas etiquetas.

A diferencia de las métricas como *accuracy* o *recall*, el valor de *loss* no es una proporción ni un porcentaje, sino una cantidad escalar cuyo mínimo ideal es cero, indicando predicciones perfectas. Valores más altos implican mayor incertidumbre o errores del modelo.

La Ilustración 75 presenta los valores de pérdida final para los experimentos exp00 a exp20. El experimento exp00, correspondiente al modelo base sin entrenamiento específico, muestra un valor de pérdida comparable al de algunos modelos finamente ajustados, pero esto puede deberse a un ajuste incorrecto o una convergencia superficial sin especialización.

En el resto de los experimentos se observan fluctuaciones notables en la pérdida, con valores que oscilan entre aproximadamente 0.17 y 0.36.

Análisis de Sobreajuste (Overfitting)

La diferencia entre la pérdida final en evaluación (*eval_loss*) y la pérdida final en entrenamiento (*train_loss*) para los experimentos exp01 a exp20. Esta métrica permite analizar el grado de generalización de cada modelo y detectar fenómenos de *overfitting* (sobreajuste) o *underfitting* (subajuste).

La diferencia se define como:

$$\Delta Loss = evaLoss - trainLoss$$

- Valores positivos grandes (barras naranjas) indican *overfitting*, ya que el modelo aprende demasiado bien el conjunto de entrenamiento y pierde capacidad de generalización en evaluación.
- Valores muy cercanos a cero (barras verdes) reflejan un entrenamiento balanceado.
- Valores negativos (barras rojas, no presentes en esta gráfica) podrían sugerir underfitting, aunque en la práctica suelen ser poco frecuentes al final del entrenamiento.

Modelos con sobreajuste significativo

Los experimentos con mayor diferencia positiva —indicando posible overfitting— son:

- exp19 ($\Delta \approx 0.332$): configuración extrema con batch pequeño (4) y secuencia corta, lo que posiblemente llevó a una alta especialización en pocos ejemplos.
- exp14 y exp15 ($\Delta \approx 0.302$): exp14 con 12 epochs y secuencia corta, y exp15 con hiperparámetros intermedios, pero probablemente con sobreentrenamiento por mayor cantidad de iteraciones.
- exp02 ($\Delta \approx 0.268$): un modelo entrenado por más epochs que el baseline, mostrando que más iteraciones no siempre mejoran el desempeño si se pierde generalización.
- exp12 y exp05 también presentan valores altos ($\Delta \approx 0.238$ y 0.234).

Modelos balanceados

Algunos experimentos se destacan por tener una diferencia muy baja o incluso negativa:

- exp02 (Δ = -0.023) y exp11 (Δ = -0.014): aquí, la pérdida en evaluación fue levemente menor que la de entrenamiento. Esto puede deberse a regularización o al azar, pero sugiere ausencia de sobreajuste.
- exp13, exp07, exp18 y exp20 presentan diferencias bajas ($\Delta < 0.05$), por lo que se consideran modelos equilibrados entre aprendizaje y generalización.

Relación con otras métricas

Un comportamiento balanceado en términos de *eval_loss* - *train_loss* no garantiza altos valores de *F1-score*, pero sí es deseable como condición para evitar sobre-entrenamiento. En cambio, los modelos con alto *overfitting* pueden mostrar métricas aceptables en entrenamiento, pero con baja estabilidad frente a nuevos datos.

Desempeños-Inferencia

Se muestra la probabilidad de detección de estafa (*scam*) para cada modelo entrenado (de exp00 a exp20). Esta métrica refleja cómo cada versión del modelo califica una instancia específica como potencialmente fraudulenta, expresada en una escala de 0 a 1 (o 0% a 100%).

El modelo base sin *fine-tuning* (exp00) muestra una probabilidad moderada (~45%), lo que evidencia una capacidad inicial limitada para identificar el patrón de estafa sin adaptación específica al dominio.

La mayoría de los modelos fine-tuneados superan el 70% de probabilidad, con varios alcanzando valores cercanos o iguales al 100%, lo que indica una alta confianza en la detección del mensaje como estafa.

Modelos con mayor probabilidad de detección

Los siguientes experimentos destacan por lograr las puntuaciones más altas, cercanas al 100%:

- exp06, exp12, exp15 y exp19: estos modelos presentan una alta sensibilidad al identificar patrones de fraude, reflejo del impacto del *fine-tuning*.
- exp02, exp14, exp13 y exp01 también superan el 80%, consolidando un grupo de configuraciones eficaces.

Modelos con probabilidad más baja

Algunos modelos presentan una probabilidad de detección inferior al 60%, indicando un comportamiento más conservador o posiblemente una menor adecuación al patrón de la muestra evaluada:

- exp04, exp18 y exp03: podrían corresponder a configuraciones con menos *epochs*, secuencias más cortas o menor *batch size*, lo cual puede limitar su capacidad de generalización.
- exp11, exp16 y exp20 se sitúan en un nivel intermedio (~50–65%).

Relación con otras métricas

Los modelos con alta probabilidad de detección de estafa tienden a coincidir con aquellos que obtuvieron F1-score, *precision* y *recall* elevados. Esto sugiere que *el fine-tuning* ha permitido una mejor discriminación de patrones de lenguaje asociados a fraude, sin comprometer la generalización.

El análisis evidencia que el proceso de *fine-tuning* mejora considerablemente la capacidad del modelo para identificar estafas. Sin embargo, no todas las configuraciones tienen el mismo impacto. Algunos modelos alcanzan un comportamiento sobresaliente, mientras que otros aún muestran márgenes de mejora. La elección del modelo final deberá considerar no solo la probabilidad de detección, sino

también su robustez general frente a múltiples ejemplos, para evitar casos de sobreajuste puntual o falsos positivos.

3.4 Ejemplos de Guiones de Estafas

Guion 5

"Hola, te contactamos del soporte de tu banco online porque notamos movimientos raros en tu cuenta hace poco. Para evitar que te la bloqueen, necesitamos chequear algunos datos rápido. ¿Nos pasás tu nombre completo y el número de tu tarjeta? ¿Qué tipo de movimientos vieron? ¿Es algo grave? Por seguridad no podemos dar detalles por este medio, pero fueron compras desde otro país y por montos altos. Si no verificamos ya, podrías quedarte sin acceso a tu cuenta. Prefiero que me manden un mail oficial antes de dar datos. Entendemos, pero si no lo hacemos ahora, tu cuenta puede ser bloqueada y podrías perder plata. ¿Podés decirnos los últimos cuatro números de tu tarjeta y el código que te va a llegar por SMS? Todavía no me llegó ningún SMS, ¿cuánto tarda? Suele llegar en unos minutos, revisá tu celu y avisanos. Si no, lo reenviamos. Ok, sigo esperando. ¿Lo pueden reenviar? Listo, ya lo mandamos de nuevo. Mientras tanto, ¿hiciste alguna compra grande afuera últimamente? No, nada raro. Por eso me preocupa esto. Justamente, por eso necesitamos confirmar que sos vos. Si no terminamos el proceso ahora, tu cuenta puede quedar comprometida. Ya me llegó el SMS, ¿qué hago? Decinos el código así bloqueamos los movimientos sospechosos. ¿Esto es seguro? Nunca paso estos datos por chat. Es solo para proteger tu cuenta, si no lo hacés ahora no podemos garantizar la seguridad y podrías perder acceso. Bueno, el código es 927314. Perfecto, gracias. En breve te llega la confirmación de que tu cuenta está protegida. Si necesitas algo más, escribinos de nuevo. ¡Buen día!"

Guion 6

"Hola, te habla el equipo de seguridad de tu banco digital. Detectamos varios intentos de acceso no autorizados a tu cuenta durante la madrugada desde diferentes ubicaciones. Por tu seguridad, bloqueamos temporalmente algunas operaciones y necesitamos verificar tu identidad para restablecer el acceso completo. ¿Podés confirmarnos tu nombre completo y tu número de documento? ¿Qué tipo de accesos detectaron? ¿Alguien intentó transferir dinero? Por motivos de confidencialidad no podemos dar detalles exactos, pero hubo intentos de transferencias y cambios de contraseña desde dispositivos desconocidos. Si no verificamos tu identidad ahora, tu cuenta seguirá limitada y podrías perder acceso a tus fondos. ¿Podés decirnos también los últimos cuatro números de tu tarjeta y el correo asociado a tu cuenta? Prefiero esperar a que me llegue una notificación oficial antes de dar datos. Entendemos tu preocupación, pero si no actuamos rápido, podrías ser víctima de fraude y perder dinero. ¿Te llegó algún SMS o correo con un código de verificación? Sí, me llegó un SMS con un código, pero no estoy seguro si es seguro compartirlo. Es seguro, este procedimiento es para proteger tu cuenta. Si no nos das el código, no podremos desbloquear tu acceso y podrías perder el control de tu cuenta. Bueno, el código es 384920. Perfecto, muchas gracias. Ahora necesitamos que confirmes la última compra que aparece en tu resumen, ¿la reconocés? No, esa compra no la hice yo. Justamente por eso me preocupa. Por eso es fundamental que sigamos con la verificación. Si no completamos el proceso ahora, tu cuenta puede quedar bloqueada indefinidamente y podrías perder acceso a tu dinero. ¿Podés también confirmar tu dirección actual y tu fecha de nacimiento? Sí, mi dirección es Calle Falsa 123 y mi fecha de nacimiento es 01/01/1990. Gracias, ya casi terminamos. Por último, ¿podés decirnos el código de seguridad que aparece al dorso de tu tarjeta? ¿Eso es necesario? Nunca me pidieron ese dato antes. Es solo para confirmar que sos el titular y evitar fraudes. Si no lo hacés ahora, no podremos garantizar la seguridad de tu cuenta. Bueno, el código es 123. Perfecto, muchas gracias. En breve recibirás un correo confirmando que tu cuenta está protegida. Si necesitás algo más, podés comunicarte con nosotros nuevamente. ¡Que tengas buen día!"

Guion 8

"Buenos días, le habla Carlos Rodríguez del departamento de seguridad del Banco República. ¿Hablo con [nombre]? Sí, soy yo. ¿Qué pasó? Señor/a, lo llamamos porque detectamos movimientos sospechosos en su tarjeta de débito terminada en 4829. Por seguridad, necesitamos verificar algunos datos antes de que bloqueemos la cuenta definitivamente. Si no confirma ahora, su cuenta será bloqueada y podría perder su dinero. ¿Movimientos sospechosos? Pero yo no hice ninguna compra rara... Exactamente por eso lo llamamos. Vemos tres intentos de compra online desde España por un total de 850 dólares. Para proteger su dinero, necesito que me confirme los últimos 4 dígitos de su cédula y el código de seguridad de su tarjeta. También necesito que me dé su clave del e-BROU para que pueda bloquear la cuenta desde aquí mismo. Es un procedimiento de emergencia y debe hacerse ya. Pero... ¿no puedo ir al banco mejor? Los bancos ya cerraron y esto es una emergencia. Su dinero está en riesgo ahora mismo. La clave del e-BROU, por favor, para proteger su cuenta."

Guion 9

"Buena tarde, le habla Patricia Suárez de Santander Uruguay. ¿Tiene un minuto? Es sobre la actualización de datos que debe hacer por ley. Si no actualiza hoy, mañana no va a poder usar su tarjeta ni acceder a su dinero. Por nueva normativa del Banco Central, todos los clientes deben actualizar sus datos antes del 30 de este mes, si no se suspende el acceso a la cuenta. Le voy a hacer unas preguntas rápidas. Su número de cuenta corriente, ¿sigue siendo el que termina en 156? Sí, ese es. Perfecto. Ahora necesito verificar que tiene acceso a su home banking. ¿Cuál es su usuario de Santander Online? Es mi cédula... 42345678. Correcto. Y la clave que usa para ingresar, la necesito para verificar que funciona correctamente en el sistema nuevo. Si no me la da ahora, su cuenta será bloqueada automáticamente. Pero esa clave no se la doy a nadie... Señor/a, es solo para verificación del sistema. Es obligatorio por ley y urgente."

Guion 10

"Hola, buen día. Le habla Andrés Martínez del BBVA. Tengo excelentes noticias para usted. Usted ha sido preseleccionado para un préstamo personal de hasta 150.000 pesos a una tasa súper preferencial del 15% anual. Es una promoción especial para clientes como usted. Solo necesito confirmar algunos datos para liberar los fondos hoy mismo. Si no acepta ahora, la oferta se pierde para siempre. Necesito su número de cuenta del BBVA y la clave de 4 dígitos de su tarjeta para hacer la transferencia inmediata. También necesito que me confirme los 3 números del dorso de la tarjeta para completar la operación. Si no me da estos datos ahora, no podrá acceder al préstamo. Señor/a, son 150.000 pesos que puede tener en su cuenta en una hora. Pero si no acepta ya, la oferta se cancela."

Guion 12

"Buena tarde, le habla Jorge Silva de OSE. Tenemos un problema con su última factura y necesitamos verificar datos. Si no actualiza sus datos ahora, su servicio podría ser suspendido. El pago está registrado, pero hay un error en el sistema con su número de cliente. Por la migración del sistema, necesito que me dé su cédula completa y su teléfono de contacto para actualizar los datos. Y el titular de la cuenta, ¿es usted o hay más nombres? Por último, ¿en qué banco tiene domiciliado el débito automático? Si no me da estos datos ahora, su cuenta podría ser bloqueada y el servicio interrumpido."

Guion 16

"¡Hola! ¡Hola! ¿Es la mamá de Martín? Señora, cálmese. Su hijo está conmigo. Hubo un problema, tuvo un accidente y tengo que llevarlo al hospital, pero necesito plata para pagar la internación. Si no transfiere el dinero ahora, no lo operan. Vaya al cajero más cercano y transfiera a esta cuenta: 778899 del BROU. Cada minuto cuenta. Si no lo hace ya, su hijo puede quedar grave. Vaya ya al cajero, son 25.000 pesos para salvar a su hijo."

3.5 Análisis Modelo Hibrido

Tabla de recursos de los modelos

Modelo	Parámetros	Memoria (VRAM)	Tiempo de Entrenamiento	Tamaño en Disco
DistilBERT (exp15)	66 millones	Baja (~2.5 GB)	1x (baseline)	~250 MB
XLM-RoBERTa Base	270 millones	Alta (~5.5-6 GB)	~2.5×	~1 GB
XLM-RoBERTa Large	560 millones	Muy alta (~12 GB)	~4-5×	~2.1 GB
RoBERTa Multilingual (MarIA)	125 millones	Media-alta (~4 GB)	~2×	~500 MB

Tabla 32 - Recursos de los modelos

Durante la etapa inicial del desarrollo del sistema de detección de estafas, se optó por el uso de DistilBERT Multilingüe como base para *fine-tuning*. Esta decisión se sustentó en su excelente balance entre precisión y eficiencia computacional, siendo un modelo considerablemente más liviano que alternativas como RoBERTa o XLM-RoBERTa, lo que lo hacía ideal para entornos con recursos limitados. En particular, se evaluaba la posibilidad de desplegar el modelo directamente en un dispositivo Android, donde la limitación de procesamiento, memoria y almacenamiento es un factor crítico.

No obstante, conforme el proyecto avanzó y el sistema comenzó a ser analizado en un entorno de cómputo más robusto específicamente en infraestructura basada en Amazon Web Services (AWS), se habilitó la posibilidad de trabajar con modelos más complejos y demandantes. Este cambio en el entorno operativo permitió reevaluar los compromisos entre precisión y costo computacional, especialmente al observar un

comportamiento decreciente en el rendimiento de DistilBERT en presencia de entradas más extensas.

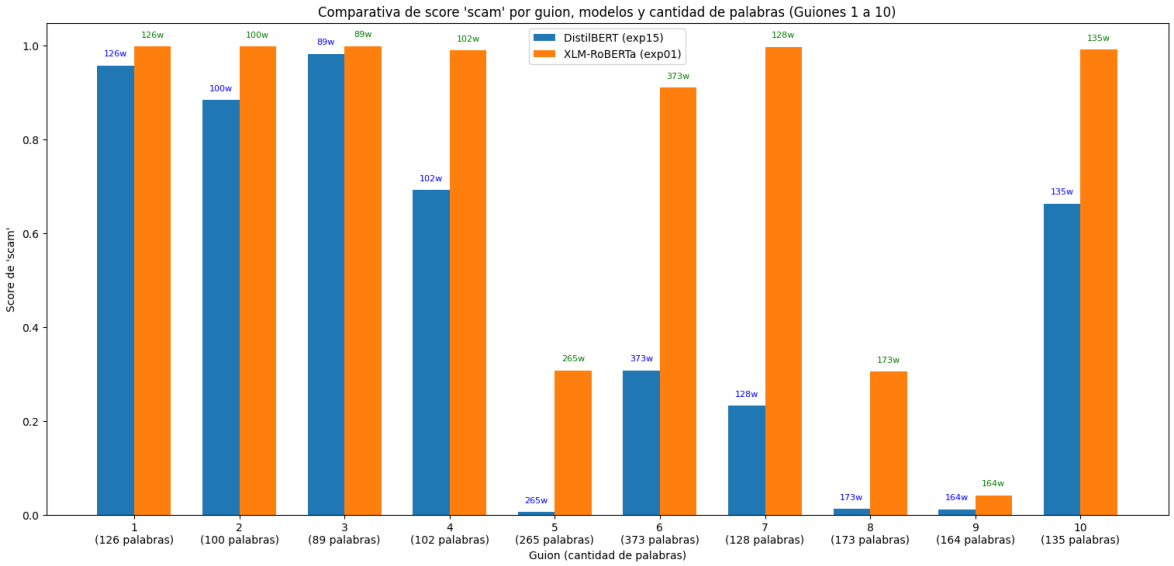


Ilustración 76 - Comparativa de score "scam“ por guion - Guiones 1 al 10

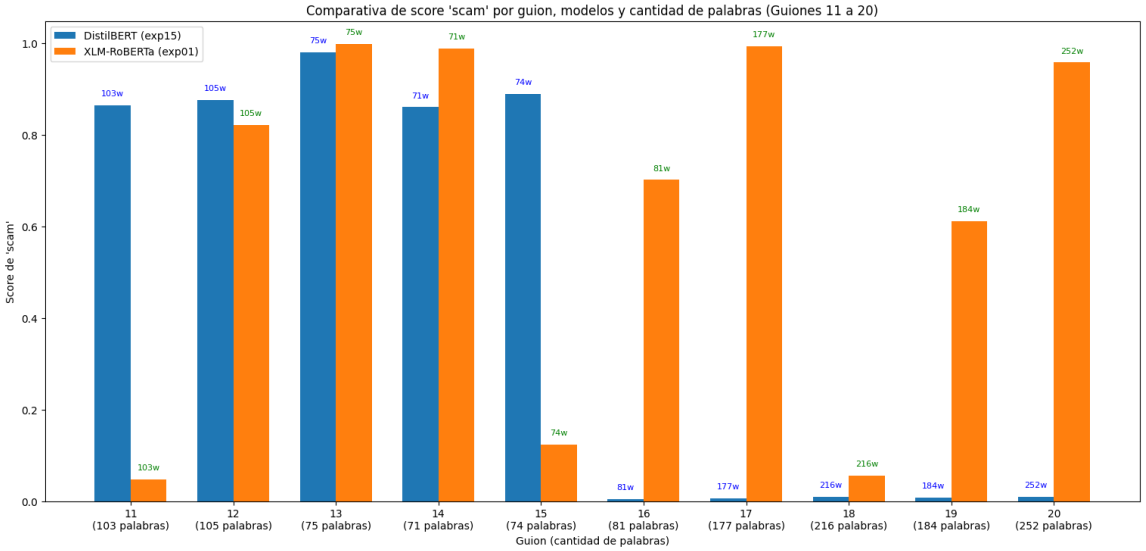


Ilustración 77 - Comparativa de score "scam“ por guion - Guiones 11 al 20

Análisis según longitud del texto

Los resultados experimentales indican que el modelo DistilBERT (exp15) mantiene un alto desempeño en textos de hasta 100 palabras, mostrando una predicción precisa y confiable en la mayoría de los casos. Sin embargo, al procesar textos más largos, se evidenció una pérdida significativa de sensibilidad ante patrones de estafa.

En contraste, el modelo XLM-RoBERTa, fine-tuneado con los mismos parámetros y *dataset*, logra mantener una performance sólida y consistente a lo largo de distintas longitudes de texto, mostrando mayor capacidad de generalización y profundidad semántica.

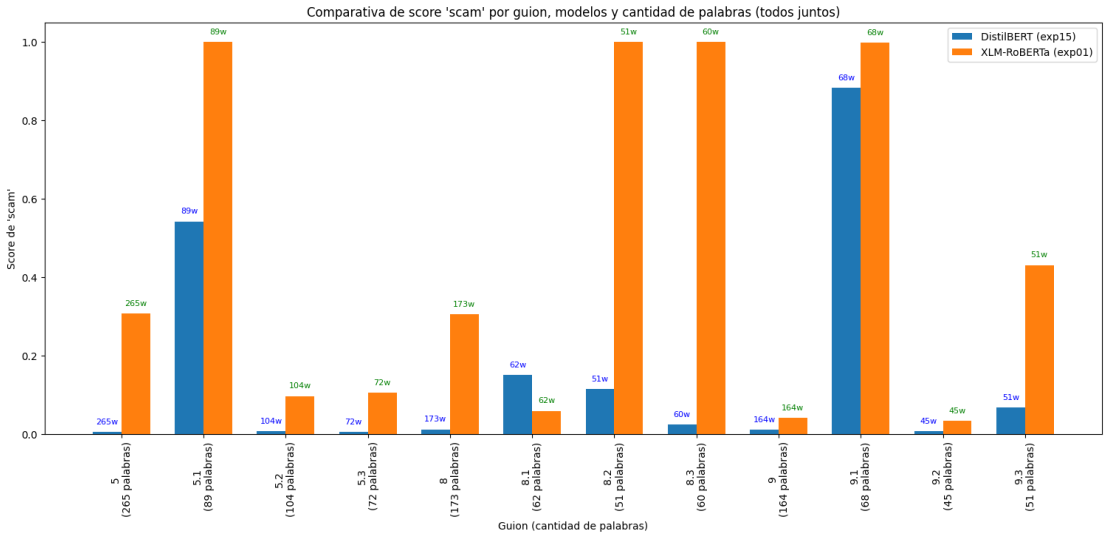


Ilustración 78 - Comparativa de score "scam" por guion y palabars (todos juntos)

Casos de uso representativos y situaciones especiales

Para ilustrar de forma práctica el desempeño de los modelos fine-tuneados en distintos contextos reales, se analizaron múltiples guiones con características variadas en longitud, tono, estructura y tipo de engaño. A continuación, se presentan casos representativos que reflejan tanto los aciertos como las limitaciones de los modelos evaluados, con especial énfasis en las diferencias entre DistilBERT (exp15) y XLM-RoBERTa (exp01).

Caso 1: Falla por longitud en DistilBERT

Guion 5 — Suplantación del banco y verificación por SMS

- Palabras: 265
- DistilBERT: Predice not_scam (score bajo)
- XLM-RoBERTa: También predice not_scam (score ≈ 0.31)
- Resultado: Ambos modelos fallan con el texto completo.

La longitud y complejidad del texto supera la capacidad de atención de ambos modelos. Aunque contiene múltiples indicios de estafa (verificación por SMS, urgencia bancaria), la carga semántica se diluye en el contexto general. Este tipo de caso justifica la adopción de una estrategia de partición de texto.

Caso 2: Mejora mediante segmentación (splitting)

Guion 5, dividido en tres partes

- 5.1 (89 palabras): XLM-RoBERTa predice scam con 99.8%
- 5.2 (104 palabras): Predice not_scam, pero con score moderado
- 5.3 (72 palabras): Predice not_scam

La partición permite focalizar el análisis en el segmento donde se solicita el código de verificación. Esto demuestra que la estrategia de *splitting* puede recuperar detecciones críticas en textos extensos, elevando la sensibilidad del sistema.

Caso 3: Detalles específicos del fraude detectados en el fragmento

Guion 8 — Compra sospechosa y solicitud de credenciales

- Texto completo (173 palabras): XLM-RoBERTa predice not_scam
- Fragmentos 8.2 y 8.3: Predicción scam con 99.8%

Aunque el texto completo no es clasificado como fraudulento, los fragmentos donde se solicitan credenciales explícitas sí lo son. Esto destaca la capacidad del modelo para responder bien ante frases cuando se reducen las distracciones del contexto.

Caso 4: Estafa altamente camuflada

Guion 9 — Supuesta actualización de datos por normativa

- Texto completo (164 palabras): XLM-RoBERTa predice not_scam (score ≈ 0.05)
- Fragmento 9.1 (68 palabras): Scam con 99.6%
- Fragmento 9.3 (51 palabras): Score ambigua (≈ 0.43)

El lenguaje formal y administrativo oculta la intención fraudulenta, fallando en texto completo. No obstante, el modelo detecta el engaño justo donde se solicita la clave. Este guion valida la importancia de contextualizar la semántica a nivel de secciones más pequeñas.

Caso 5: Guion corto con alta detección directa

Guion 16 — Falsa emergencia médica de un familiar

- Texto completo (≈ 80 palabras)
- DistilBERT y XLM-RoBERTa: Ambos clasifican correctamente como *scam*

Este caso demuestra que textos cortos, emocionales y directos suelen ser fácilmente detectables. La claridad en la intención maliciosa facilita la tarea de los modelos, sin necesidad de recurrir a estrategias auxiliares.

Caso 6: Falla de DistilBERT por ambigüedad semántica

Guion 12 — Consulta por reintegro con datos del titular

- Texto completo (≈ 140 palabras)
- DistilBERT: Predice not_scam
- XLM-RoBERTa: Scam con score alto (≈ 96%)

La estructura del mensaje es formal y gentil, imitando a una atención real al cliente. DistilBERT no capta el trasfondo fraudulento, pero XLM-RoBERTa logra identificarlo al procesar más eficazmente el contexto semántico. Este ejemplo resalta la necesidad de usar modelos con mayor capacidad contextual para casos limítrofes.

Caso 7: Score bajo pero respuesta correcta

Guion 6 — Reclamo por débito no autorizado

- Texto completo (≈ 110 palabras)
- XLM-RoBERTa: Clasifica *scam* con score ≈ 0.61

Aunque la probabilidad no es extrema, la clasificación es correcta. Este tipo de ejemplo justifica utilizar umbrales dinámicos o considerar la confianza del modelo en relación con el tipo de estafa y contexto.

Caso 8: Guion extremadamente largo con contenido crítico al medio

Guion 10 — Estafa por WhatsApp con historia elaborada

- Texto completo (≈ 300 palabras): Ambos modelos fallan
- Fragmento 10.2 (parte media): XLM-RoBERTa predice *scam* con score alto

Los modelos fallan al tratar el texto en su totalidad debido a la carga narrativa extensa. Sin embargo, la parte central del guion contiene el pedido de dinero y datos personales, y es detectado exitosamente.

ANEXO 4

4.1 Arquitectura del Sistema

Instancia de EC2 en AWS

Ilustración 79 - Instancia EC2 en AWS

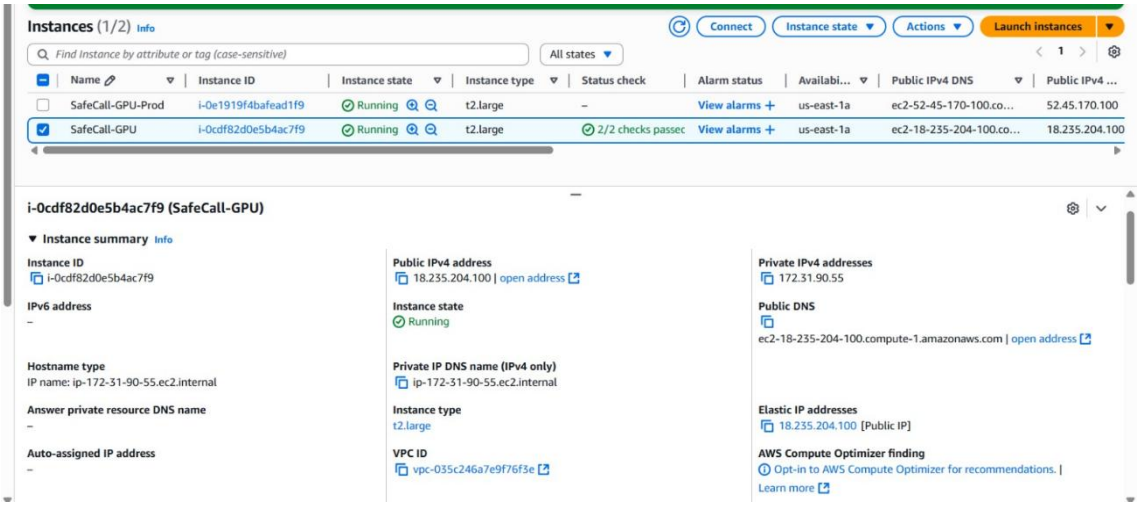


Ilustración 80 - Maquinas Develop y Produccion

Principales ADRs

ADR-001: Arquitectura Cliente-Servidor para Distribución de Procesamiento

Estado: Aceptado

Fecha: Abril 2025

Participantes:

- Suren Keushkerian
- Santiago Porro
- Andrés Montaño

Contexto

SafeCall requiere realizar tareas computacionalmente intensivas (transcripción de audio con Whisper, análisis con modelo RoBERTa) que no pueden ejecutarse eficientemente en dispositivos móviles. Al mismo tiempo, la aplicación debe mantener una interfaz responsiva y una experiencia de usuario fluida para adultos mayores.

Opciones Consideradas

- Procesamiento completamente local en el dispositivo
  - Pros: Máxima privacidad, funciona offline
  - Contras: Consumo excesivo de batería, tiempos de procesamiento lentos, limitaciones de hardware en dispositivos de gama media
- Procesamiento completamente en la nube
  - Pros: Capacidad de cómputo ilimitada, actualizaciones centralizadas
  - Contras: Dependencia total de conectividad, mayor latencia
- Arquitectura cliente-servidor híbrida
  - Pros: Cliente ligero para usabilidad, servidor robusto para procesamiento pesado
  - Contras: Requiere conectividad, mayor complejidad arquitectónica

Decisión

Implementar una arquitectura cliente-servidor donde la aplicación móvil Android actúa como cliente ligero enfocado en usabilidad y respuesta rápida, mientras que el *backend* en la nube asume el trabajo pesado de transcripción y análisis.

Justificación

- Usabilidad: Mantiene la aplicación móvil liviana y responsiva, evitando consumo excesivo de batería que afectaría la experiencia del usuario
- Deployabilidad: Facilita actualizaciones del modelo ML sin requerir actualizaciones de la aplicación

Implementación

- Cliente móvil: Gestiona UI, monitoreo de llamadas, integración con Firebase, envío de audios al *backend*
- *Backend*: FastAPI con módulos de preprocesamiento, transcripción (Whisper), inferencia (RoBERTa), almacenamiento (MongoDB, Redis)
- Comunicación: Protocolos HTTP seguros mediante Retrofit y OkHttp

Consecuencias

Positivas:

- Experiencia de usuario fluida y responsiva
- Facilidad para actualizar modelos ML sin afectar a usuarios

Negativas:

- Dependencia de conectividad a Internet
- Costos operativos recurrentes de infraestructura cloud

ADR-002: Patrón MVVM para Arquitectura Android

Estado: Aceptado

Fecha: Mayo 2025

Participantes:

- Suren Keushkerian
- Santiago Porro
- Andrés Montaño

Contexto

La aplicación móvil requiere una arquitectura que facilite la mantenibilidad, separe responsabilidades claramente y permita una experiencia de usuario consistente. Dado el público objetivo (adultos mayores), es necesario que la UI sea robusta y libre de errores.

Opciones Consideradas

1. Arquitectura sin patrón definido
    - Pros: Desarrollo inicial más rápido
    - Contras: Código difícil de mantener, lógica mezclada con UI
2. MVC (Model-View-Controller)
    - Pros: Patrón conocido
    - Contras: Acoplamiento entre *View* y *Controller* en Android
3. MVVM (Model-View-ViewModel)
    - Pros: Separación clara de responsabilidades, soporte nativo de Android, facilita *testing*
    - Contras: Curva de aprendizaje inicial

Decisión

Adoptar el patrón arquitectónico MVVM (Model-View-ViewModel) para la aplicación Android.

Justificación

- Usabilidad: La separación clara entre capas reduce errores que podrían afectar la experiencia del usuario
- Performance: ViewModel con StateFlow permite actualizaciones reactivas eficientes sin recargas manuales
- Deployabilidad: Facilita la evolución y mantenimiento del código

Implementación

Capa Model:

- Lógica de negocio y comunicación con backend
- Retrofit + OkHttp para solicitudes HTTP
- kotlinx-serialization + Gson para transformación JSON
- Integración con Firebase (Firestore, Analytics)

Capa ViewModel:

- StateFlow para gestión reactiva de estado
- Exposición de estado a la interfaz

Capa View:

- Jetpack Compose para UI declarativa
- Navigation Compose para flujo entre pantallas
- Sistema de alertas y notificaciones

Consecuencias

Positivas:

- Código mantenible y testeable
- UI responsiva con actualizaciones inmediatas
- Facilita la aplicación de heurísticas de Nielsen

Negativas:

- Requiere aprendizaje de conceptos de programación reactiva

ADR-003: Jetpack Compose para Interfaz de Usuario

Estado: Aceptado

Fecha: Mayo 2025

Participantes:

- Suren Keushkerian
- Santiago Porro
- Andrés Montaño

Contexto

La interfaz debe ser clara, accesible y adaptable para adultos mayores. Se necesita una solución que permita desarrollo rápido, consistencia visual y facilite la aplicación de principios de usabilidad (heurísticas de Nielsen).

Opciones Consideradas

1. XML tradicional de Android
    - Pros: Conocimiento previo del equipo
    - Contras: Más verboso, menos flexible para cambios

Jetpack Compose

- Pros: UI declarativa, modular, adaptable, temas dinámicos
- Contras: Tecnología más reciente

Decisión

Utilizar Jetpack Compose para la construcción de la interfaz de usuario.

Justificación

- Usabilidad: Permite implementar fácilmente las heurísticas de Nielsen (visibilidad del estado, prevención de errores, reconocimiento antes que recuerdo)
- Usabilidad: Temas dinámicos para modificar tipografía y aspectos visuales según necesidades de accesibilidad
- Performance: UI declarativa eficiente con recomposiciones optimizadas

Implementación

- UI declarativa y modular
- Gestión de estado con StateFlow
- Temas dinámicos configurables
- Navigation Compose para flujo de navegación
- SharedPreferences para persistencia de configuraciones ligeras
- Sistema de alertas con mensajes emergentes, códigos de color y señales sonoras

Consecuencias

Positivas:

- Interfaz clara y adaptable
- Facilita personalización visual para adultos mayores
- Desarrollo más rápido de componentes
- Aplicación directa de heurísticas de Nielsen

Negativas:

- Tecnología relativamente nueva (menos recursos de ayuda)

ADR-004: Modularización del Backend en Etapas Especializadas

Estado: Aceptado

Fecha: Abril 2025

Participantes:

- Suren Keushkerian
- Santiago Porro
- Andrés Montaño

Contexto

El backend debe procesar audio, transcribir, analizar con ML y devolver resultados rápidamente (≤30s). Se necesita una arquitectura que optimice tiempos de ejecución y facilite actualizaciones de componentes.

Opciones Consideradas

Backend monolítico sin separación de módulos

- Pros: Simplicidad inicial
- Contras: Difícil de mantener y optimizar

Microservicios independientes

- Pros: Escalabilidad máxima
- Contras: Complejidad excesiva para Prototipo funcional, *overhead* de comunicación

Backend modular con separación de etapas

- Pros: Optimización por etapa, facilita actualizaciones
- Contras: Requiere diseño cuidadoso de interfaces

Decisión

Implementar *backend* modular con separación clara en etapas: preprocesamiento, transcripción, inferencia y monitoreo.

Justificación

- Performance: Permite optimizar cada etapa independientemente para cumplir con RNF01
- Deployabilidad: Facilita reemplazo o actualización de componentes (ej: cambiar modelo ML) sin impactar el resto del sistema
- Performance: Permite procesamiento secuencial eficiente con trazabilidad completa

Implementación

Módulo Preprocesamiento:

- Normalización y segmentación de audio

Módulo Transcripción:

- Whisper para conversión voz-a-texto
- Preparación de texto para análisis

Módulo Inferencia:

- Modelo RoBERTa con *fine-tuning* para detección de estafas
- Generación de predicción
- Técnica de juez con GPT-3.5-turbo para validación

Módulo Monitoreo:

- Registro de métricas de rendimiento
- *Logging* de errores

Almacenamiento:

- Redis para consultas rápidas temporales
- MongoDB para resultados persistentes y análisis posteriores

Consecuencias

Positivas:

- Tiempos de ejecución optimizados
- Fácil actualización de componentes
- Cumplimiento del RNF01

Negativas:

- Mayor complejidad inicial de diseño

ADR-005: Docker para Deployabilidad del Backend

Estado: Aceptado

Fecha: Abril 2025

Participantes:

- Suren Keushkerian
- Santiago Porro
- Andrés Montaño

Contexto

El *backend* necesita desplegarse de manera confiable en diferentes entornos (desarrollo, *staging*, producción). Se requiere consistencia en la ejecución y capacidad de *rollback* rápido ante problemas.

Opciones Consideradas

Despliegue directo en instancia EC2

- Pros: Sin *overhead* de contenedores
- Contras: Inconsistencias entre entornos, difícil *rollback*

Máquinas virtuales

- Pros: Aislamiento completo
- Contras: Mayor consumo de recursos

Contenedores Docker

- Pros: Portabilidad, consistencia, *rollback* rápido.
- Contras: Requiere orquestación para escalado.

Decisión

Desplegar el *backend* mediante contenedores Docker ejecutados en instancias EC2 de AWS (2 vCPUs, 8GB RAM).

Justificación

- Deployabilidad: Portabilidad entre entornos y consistencia en ejecución
- Deployabilidad: *Rollback* rápido a versiones previas ante problemas
- Performance: Configuración de hardware (2 vCPUs, 8GB RAM) suficiente para modelos ML sin costos excesivos

Implementación

- Contenedores Docker con la aplicación FastAPI
- Despliegue en EC2 de AWS
- Elastic IP para simplificar gestión de endpoints
- Pipeline CI/CD con GitHub Actions para despliegue automatizado
- Tiempo de despliegue: ~3 minutos por commit

Consecuencias

Positivas:

- Despliegues rápidos y confiables
- Consistencia entre entornos (dev, staging, prod)
- Facilita *rollback* ante errores

Negativas:

- Costos operativos recurrentes de infraestructura

En escenarios de producción masiva será necesario introducir balanceador de carga para distribuir solicitudes entre múltiples instancias EC2.

ADR-006: Pipeline CI/CD con GitHub Actions

Estado: Aceptado

Fecha: Mayo 2025

Participantes:

- Suren Keushkerian
- Santiago Porro
- Andrés Montaño

Contexto

El proyecto requiere frecuentes despliegues con calidad garantizada. Se necesita automatizar verificación de código y despliegue para minimizar errores y tiempos de inactividad.

Opciones Consideradas

Despliegues manuales

- Pros: Sin configuración inicial
- Contras: Propenso a errores, lento

Jenkins

- Pros: Altamente configurable
- Contras: Requiere mantenimiento de servidor

GitHub Actions

- Pros: Integrado con repositorio, fácil configuración
- Contras: Limitaciones en minutos gratuitos

Decisión

Implementar pipeline CI/CD con GitHub Actions para verificación, empaquetado y despliegue automatizado.

Justificación

- Deployabilidad: Despliegues automáticos con commits disponibles en producción en ~3 minutos
- Deployabilidad: Verificación de compilación en cada Pull Request garantiza calidad
- Performance: Minimiza tiempos de inactividad

Implementación

Pipeline incluye:

- Verificación de compilación en cada Pull Request
- Despliegue automático a ambiente de staging
- Empaquetado y despliegue de Docker

4.2 Diseño del Backend

Diagrama de despliegue

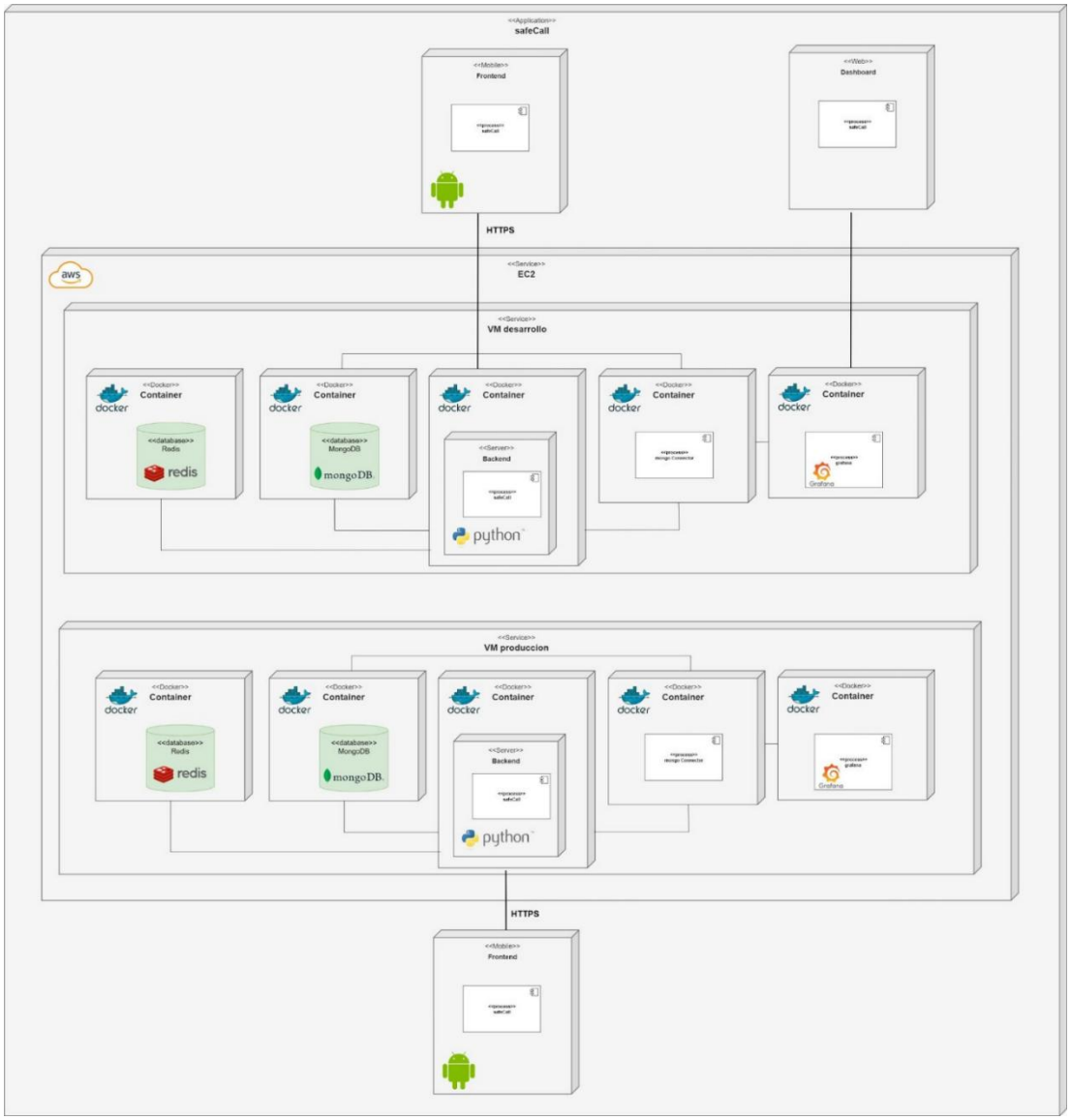


Ilustración 81 - Diagrama despliegue backend

Diagrama de flujo del backend

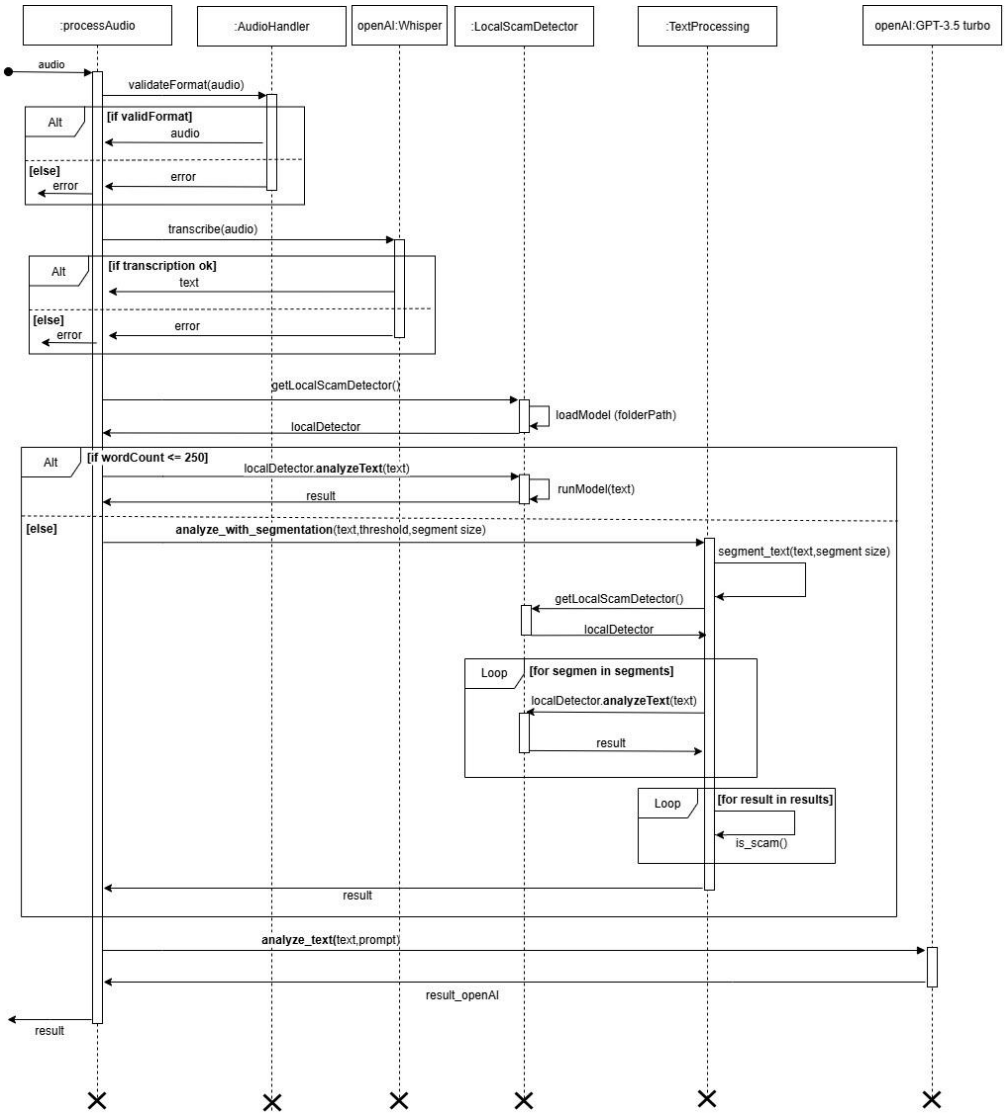


Ilustración 82 Flujo del Backend

Documentación de la API mediante Swagger.

Ilustración 83 - Api números sospechosos

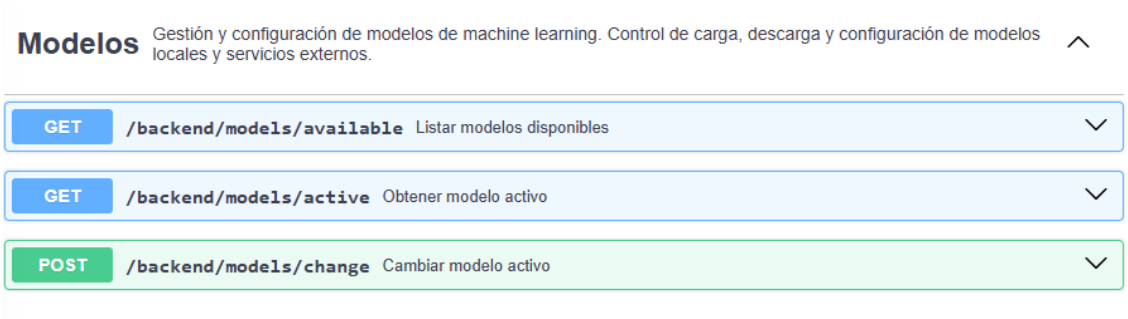


Ilustración 84 - API gestión de modelos

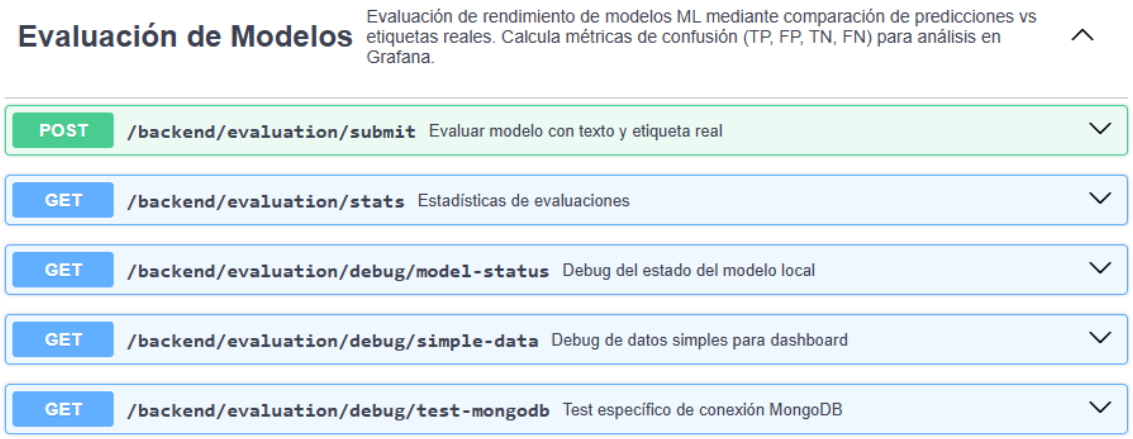


Ilustración 85 - API evaluación de modelos

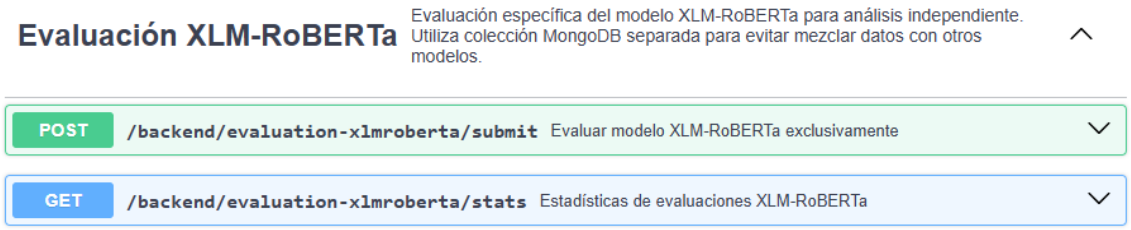


Ilustración 86 - API evaluación Roberta

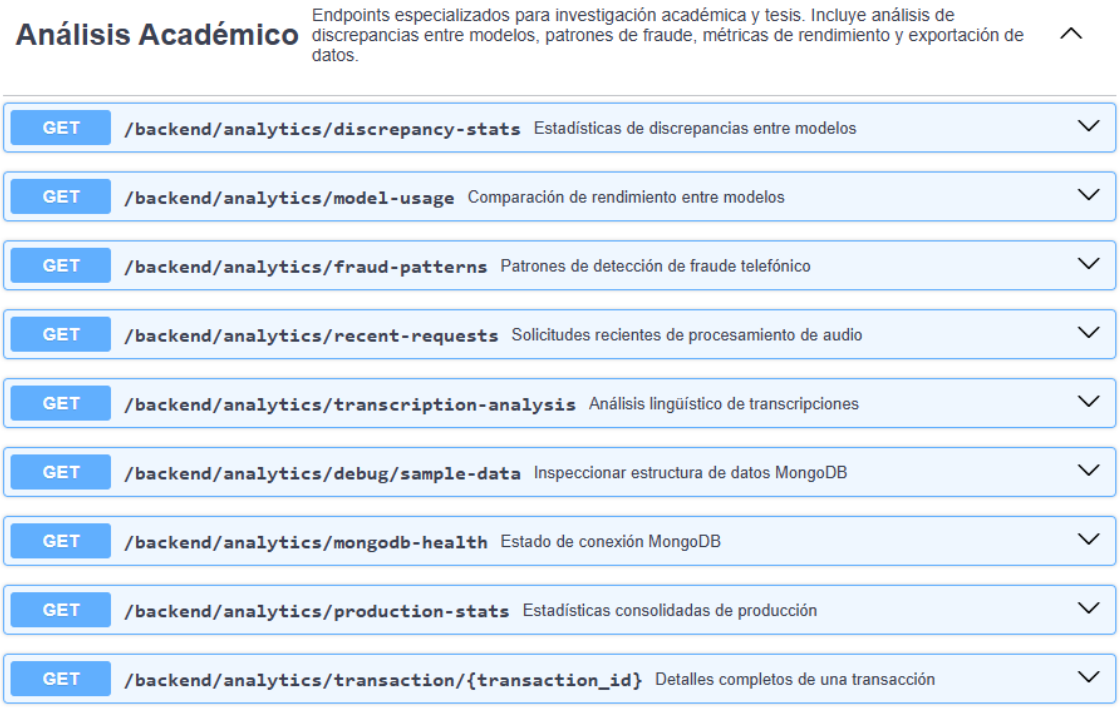


Ilustración 87 - API análisis académico

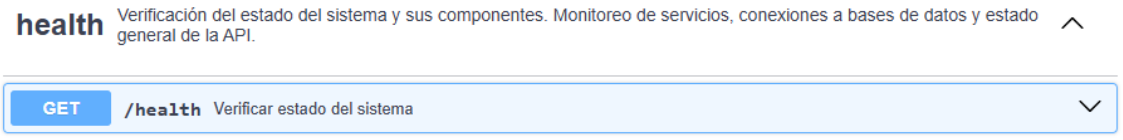


Ilustración 88 - API estado del sistema

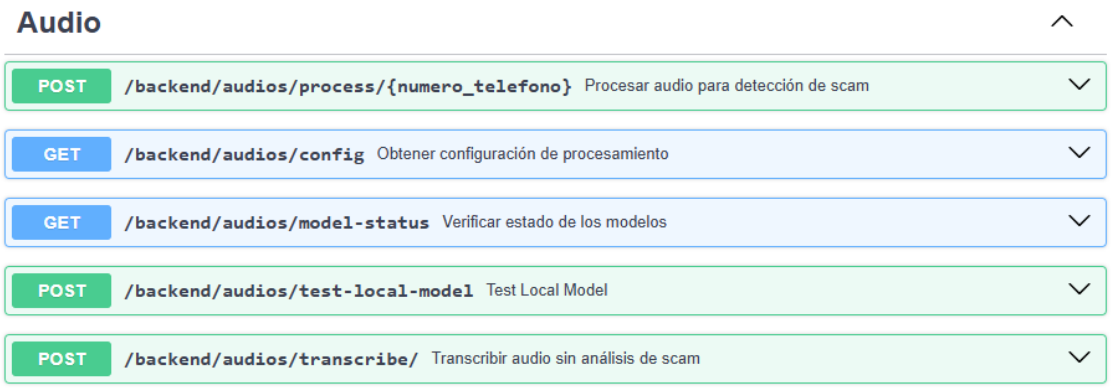


Ilustración 89 - API gestión audios

Resultado de Endpoint de Evaluación de modelo Roberta

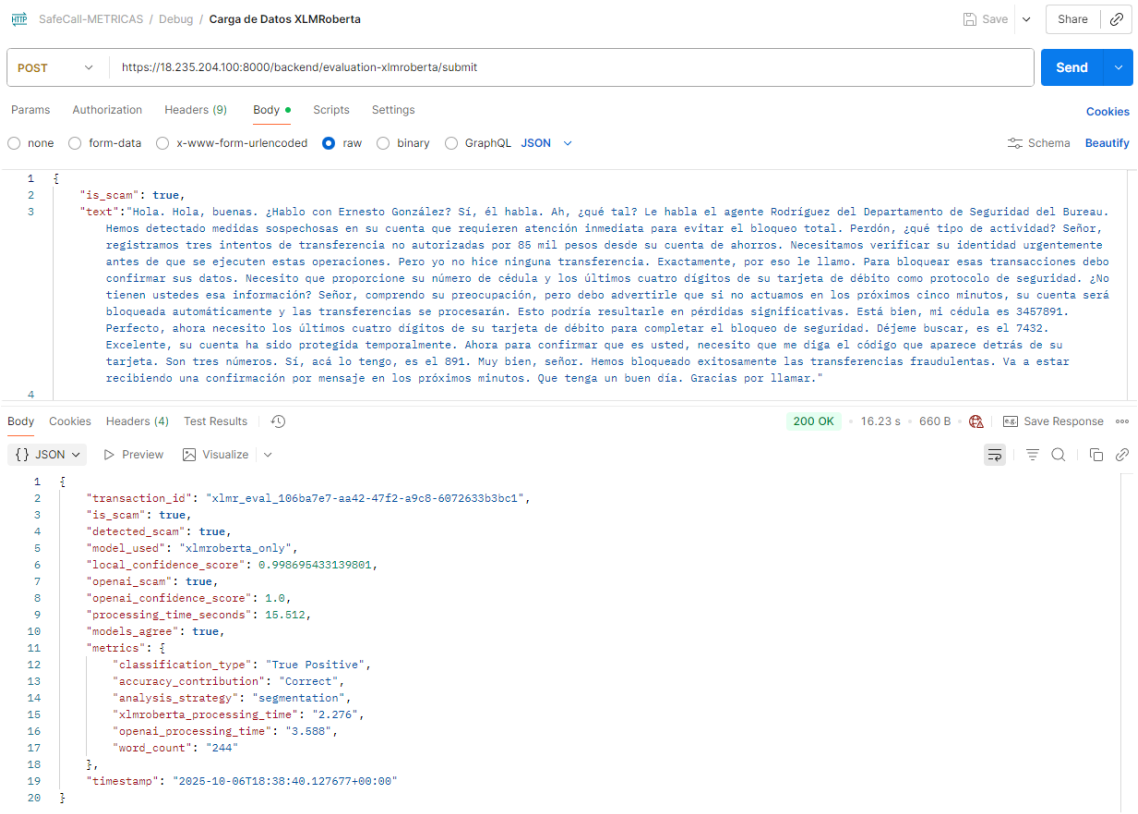


Ilustración 90 – Resultado de consulta a endpoint de Evaluacion de modelo Roberta

4.3 Diseño Aplicación Móvil

Diagrama Arquitectura de Alto nivel del *Frontend*

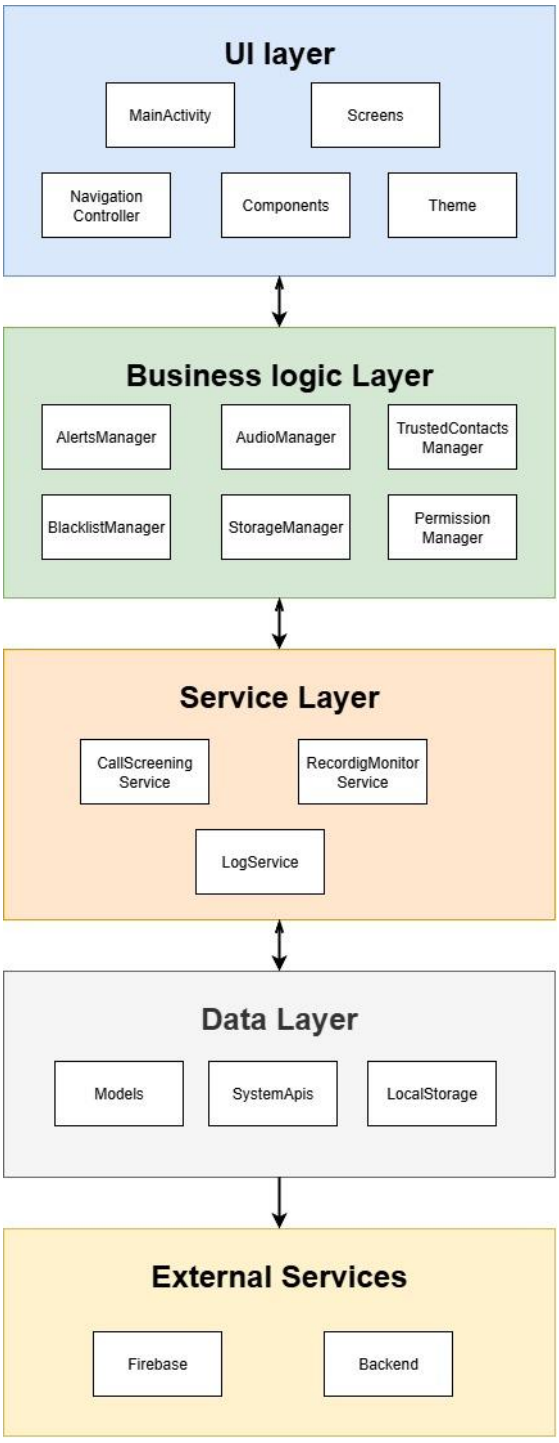


Ilustración 91 - Arquitectura de Alto nivel del Frontend

Diagrama de componentes del Frontend

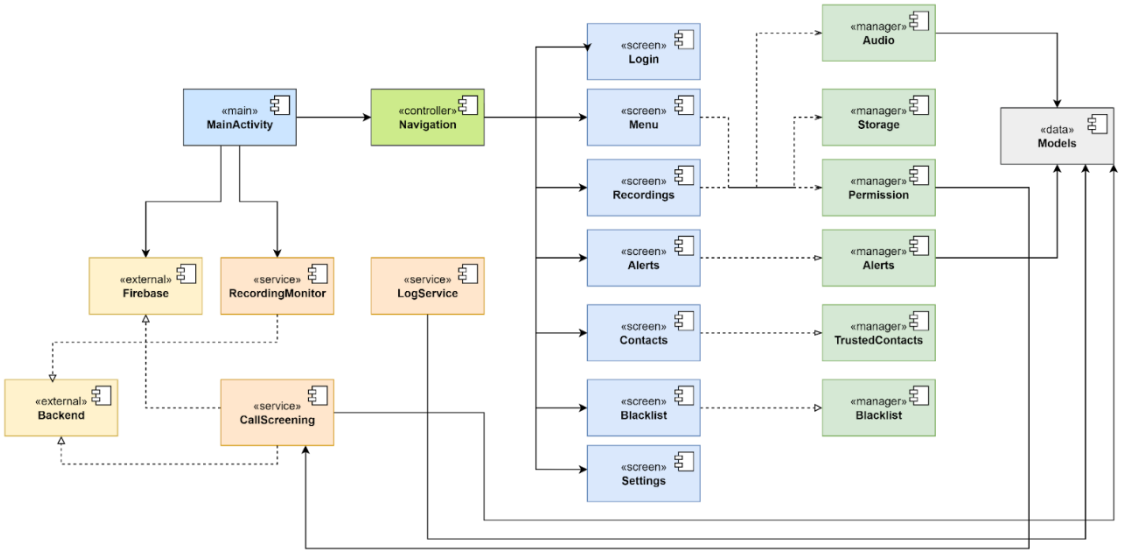


Ilustración 92 - Componentes del Frontend

Diagrama de secuencia de procesamiento de llamada entrante

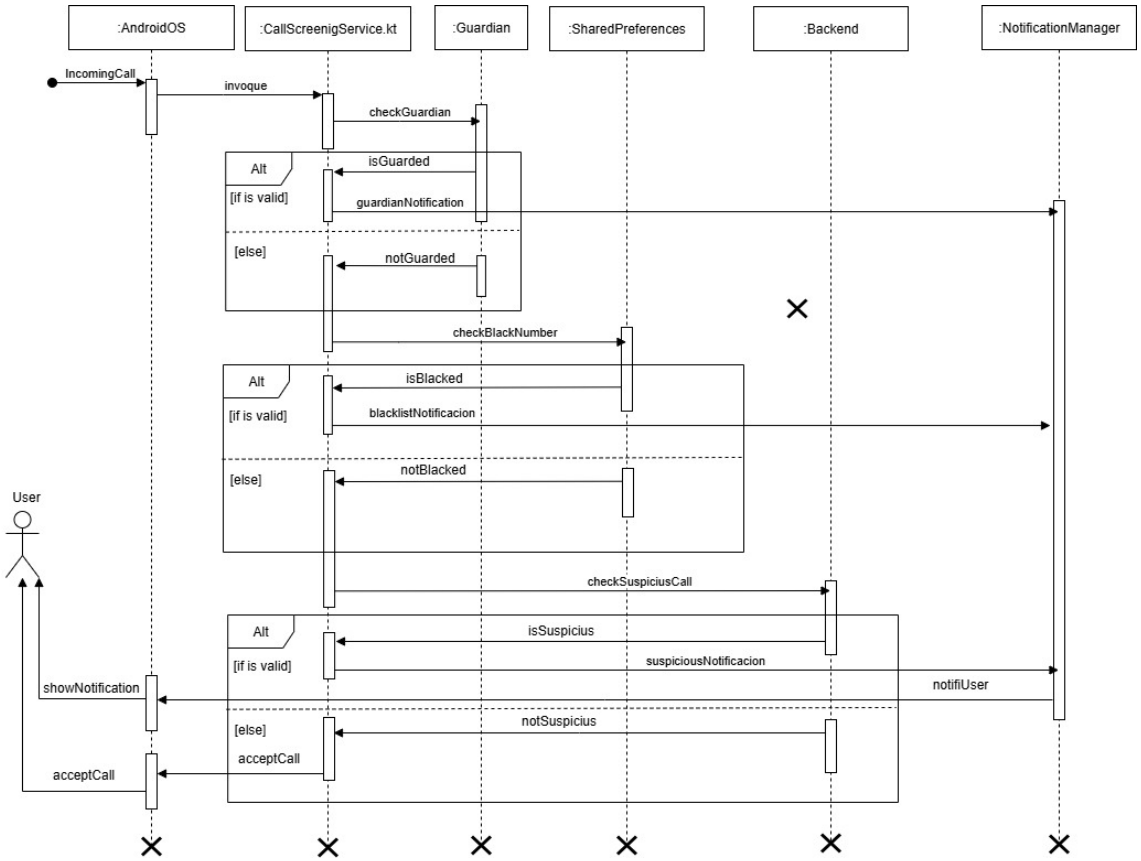


Ilustración 93 - Secuencia procesamiento llamada entrante

Diagrama de secuencia de envío de grabación al backend

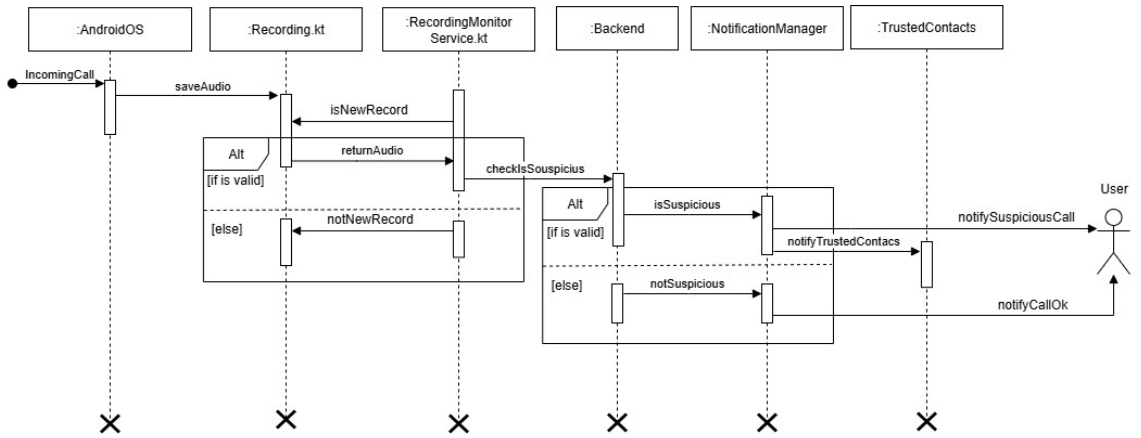


Ilustración 94 - Secuencia grabación backend

Firebase Analytics dashboard

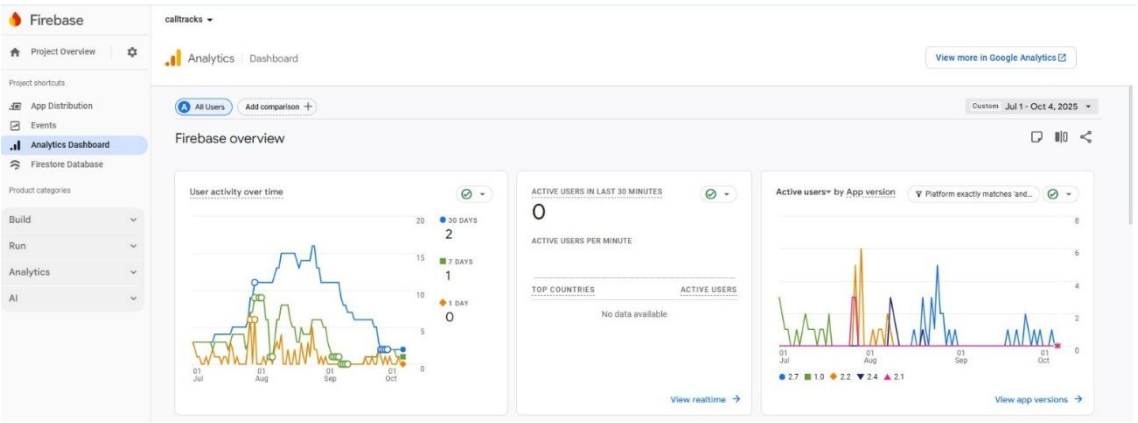


Ilustración 95 - Dashboard Firebase

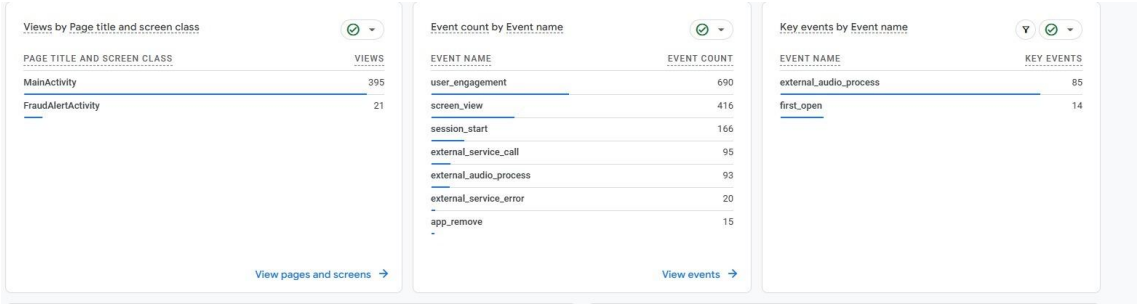


Ilustración 96 - Eventos registrados

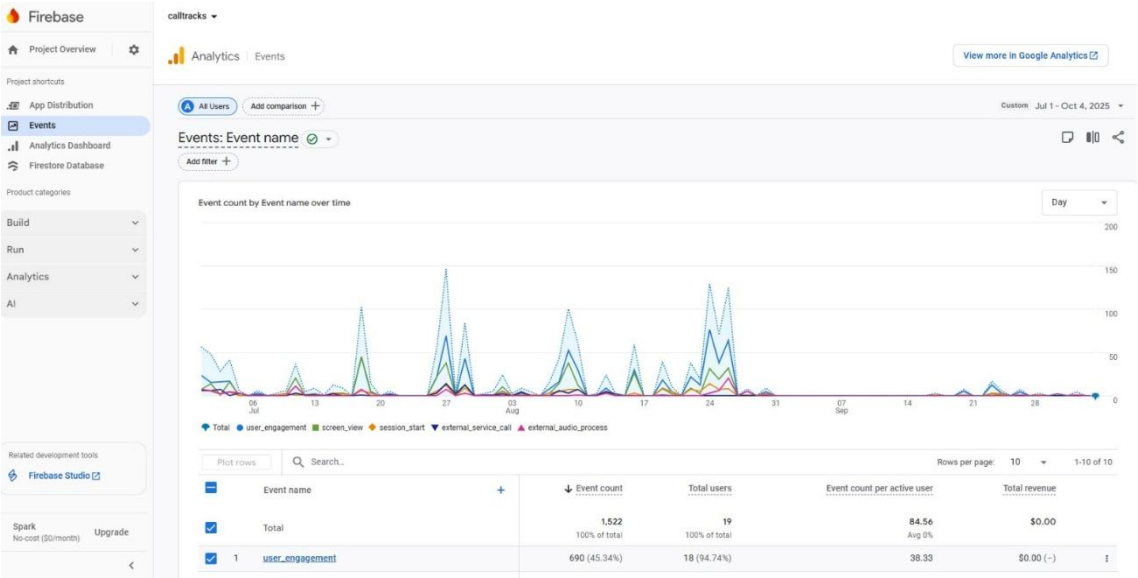


Ilustración 97 - Grafica de eventos

Heurísticas aplicadas por pantalla

Pantalla	Heurística aplicada	Justificación
Login	Reconocimiento antes que recuerdo	Elimina la necesidad de recordar credenciales, facilitando el acceso.
	Estética y diseño minimalista	Interfaz simple y clara, sin elementos innecesarios, adecuada para adultos mayores.
Menú Principal	Visibilidad del estado del sistema	El semáforo indica en tiempo real el estado del servicio.
	Prevención de errores	Botón de salida explícito evita confusión al cerrar la app.
	Estética y diseño minimalista	Presenta solo las opciones esenciales, reduciendo la carga cognitiva.
Contactos	Correspondencia entre el sistema y el mundo real	El uso de colores y etiquetas facilita la comprensión intuitiva.
Historial de Llamadas	Visibilidad del estado del sistema	Diferenciación por colores entre llamadas atendidas y pérdidas.
	Ayuda y documentación	Etiquetas que explican acciones disponibles en cada registro. Historial de versiones.
Grabaciones	Reconocimiento antes que recuerdo	Uso de controles de reproducción estándar comprensibles.
	Consistencia y estándares	Opciones de compartición a través de funciones nativas de Android.

	Ayuda a reconocer, diagnosticar y recuperarse de errores	Posibilidad de manejar errores al reproducir, eliminar o compartir archivos.
Configuraciones	Flexibilidad y eficiencia de uso	Modos accesibles y guardián adaptan la app a diferentes necesidades.
	Visibilidad del estado del sistema	Presenta la versión actual y el historial de cambios de la aplicación.
Blacklist	Control y libertad del usuario	Posibilidad de agregar o retirar números de la lista en cualquier momento.
Notificaciones	Visibilidad del estado del sistema	Mantienen informado al usuario sobre eventos relevantes.
	Consistencia y estándares	Uso del patrón estándar de notificaciones de Android, adaptado a la app.
	Flexibilidad y eficiencia de uso	Notificaciones personalizadas permiten responder o descartar rápidamente.
	Prevención de errores	Alertas inmediatas reducen el riesgo de que el usuario ignore eventos críticos.

Tabla 33 - Heurísticas aplicadas por pantalla

4.4 Performance

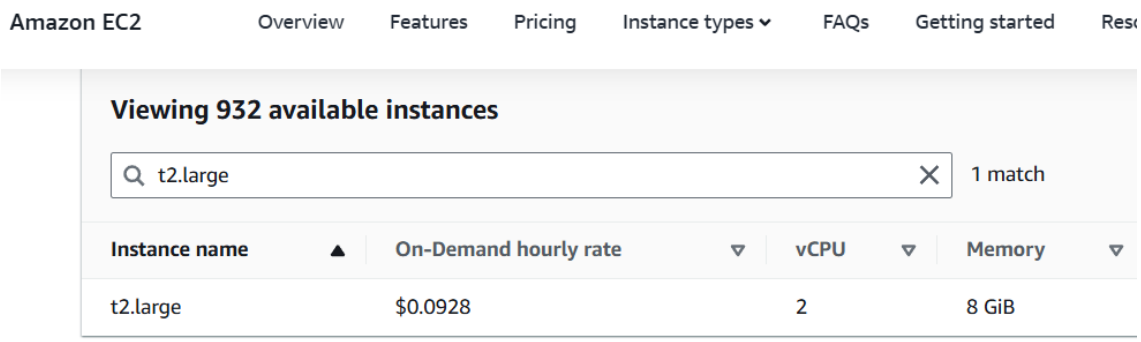


Ilustración 98 - Costo instancia t2.large

1. **Capacidad para contenedores Docker:**

    - Backend FastAPI (~1-2 GB RAM)
    - Redis (cache en memoria, ~1 GB RAM)
    - MongoDB (base de datos persistente, ~1-2 GB RAM)
    - Modelo RoBERTa (inferencia local, ~2-3 GB RAM)
    - Overhead del sistema operativo (~1 GB RAM)
    - **Total requerido**: ~6-7 GB → **8 GB suficiente**

2. **Decisión sobre Whisper:**

    - Whisper fue considerado inicialmente para transcripción de audio
    - Requiere GPU (CUDA) para rendimiento óptimo
    - Instancias con GPU (p3.2xlarge) cuestan ~$3.06/hora
    - Solución adoptada: API de transcripción externa

Alternativas evaluadas:

- **t2.medium** (2 vCPUs, 4GB): Insuficiente para todos los contenedores + modelo ML.
- **t2.xlarge** (4 vCPUs, 16GB): Sobredimensionada, costo $135/mes (24/7)
- **Instancias con GPU**(para Whisper): Costo muy alto para prototipo académico

ANEXO 5

5.1 Evaluación de Modelos

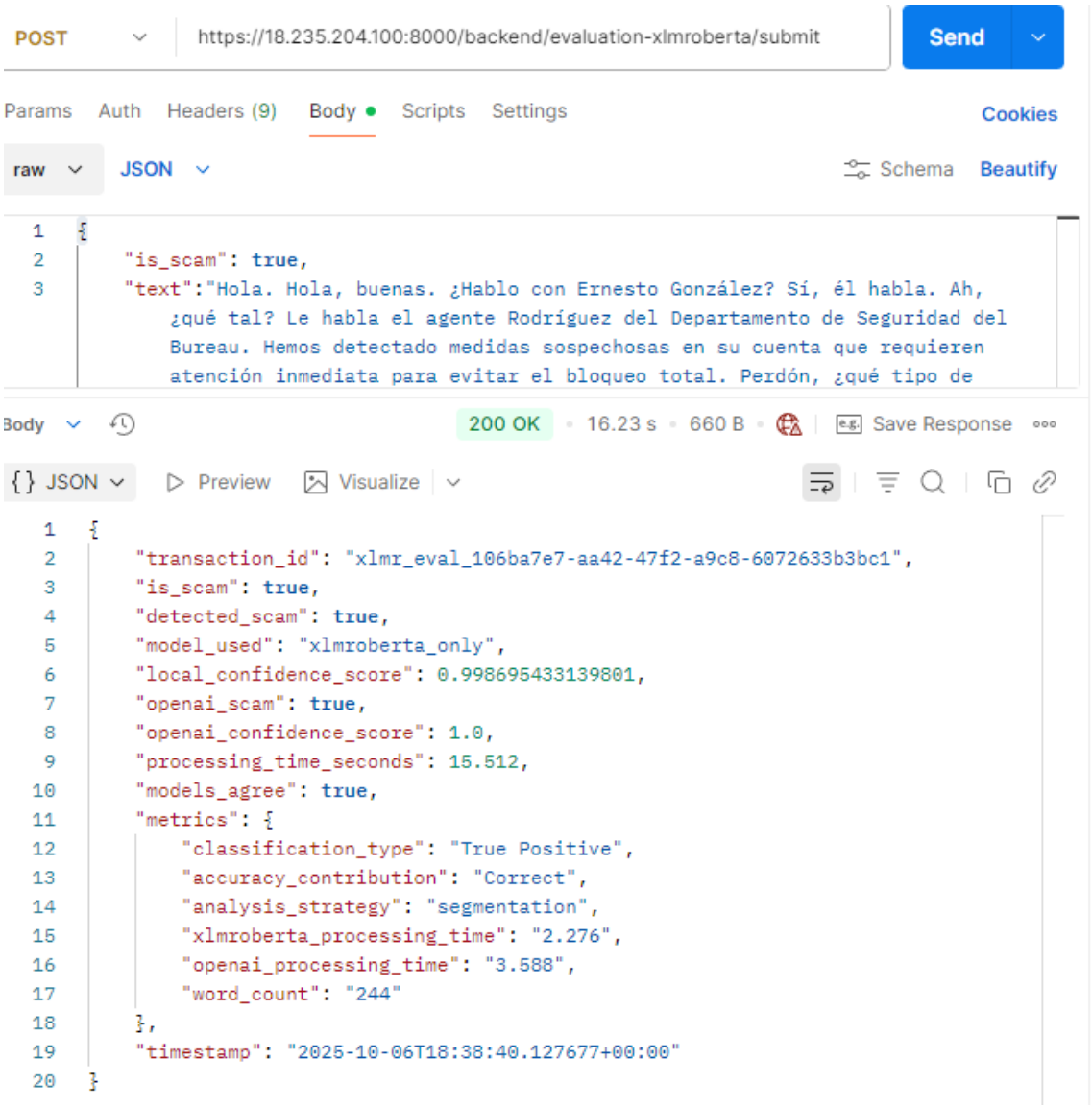


Ilustración 99 – Ejemplo de petición con envío de estafa de ejemplo para Evaluación de modelo Roberta

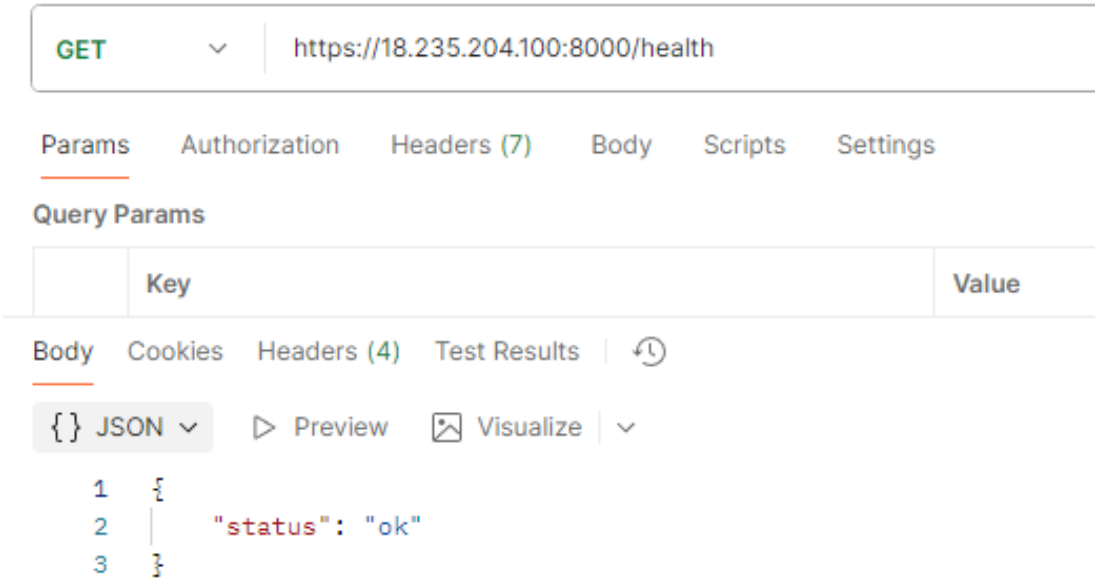


Ilustración 100 – Ejemplo de Respuesta de endpoint health

Consulta de número sospechoso

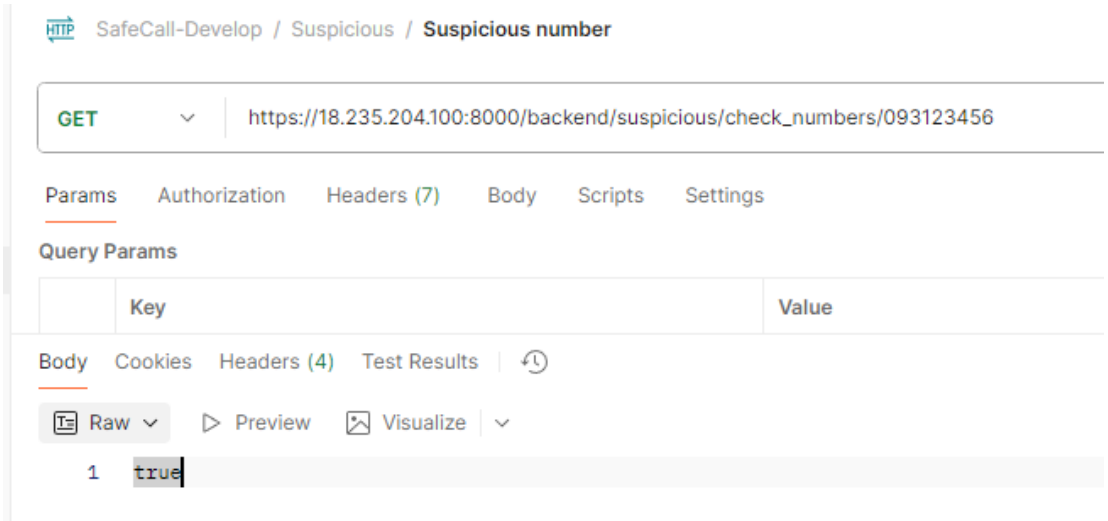


Ilustración 101 – Ejemplo de Respuesta de endpoint para verificación de número sospechoso

Consulta de datos de Modelos

GET https://18.235.204.100:8000/backend/evaluation/debug/model-status

Params Authorization Headers (7) Body Scripts Settings

Body Cookies Headers (4) Test Results

{ } JSON Preview Visualize


```
{
    "timestamp": "2025-10-07T22:20:40.726529+00:00",
    "environment": {
        "transformers_version": "4.36.2",
        "torch_version": "2.2.0+cpu",
        "torch_cuda_available": false
    },
    "model_paths": {
        "distilbert": {
            "path": "/opt/models/distilbert-scam-detector",
            "exists": true,
            "files": [
                "config.json",
                "metadata.json",
                "special_tokens_map.json",
                "tokenizer_config.json",
                "vocab.txt",
                "model.safetensors",
                "tokenizer.json",
                "training_args.bin"
            ],
            "has_config": true,
            "has_model": true
        },
        "xlmroberta": {
            "path": "/opt/models/xlmroberta-scam-detector",
            "exists": true,
            "files": [
                "config.json",
                "metadata.json",
                "model.safetensors",
                "special_tokens_map.json",
                "tokenizer.json",
                "tokenizer_config.json",
                "training_args.bin"
            ],
            "has_config": true,
            "has_model": true
        }
```


Ilustración 102 – Ejemplo de Respuesta de endpoint para verificación de status de modelos

Consulta de estadísticas de Modelo Roberta

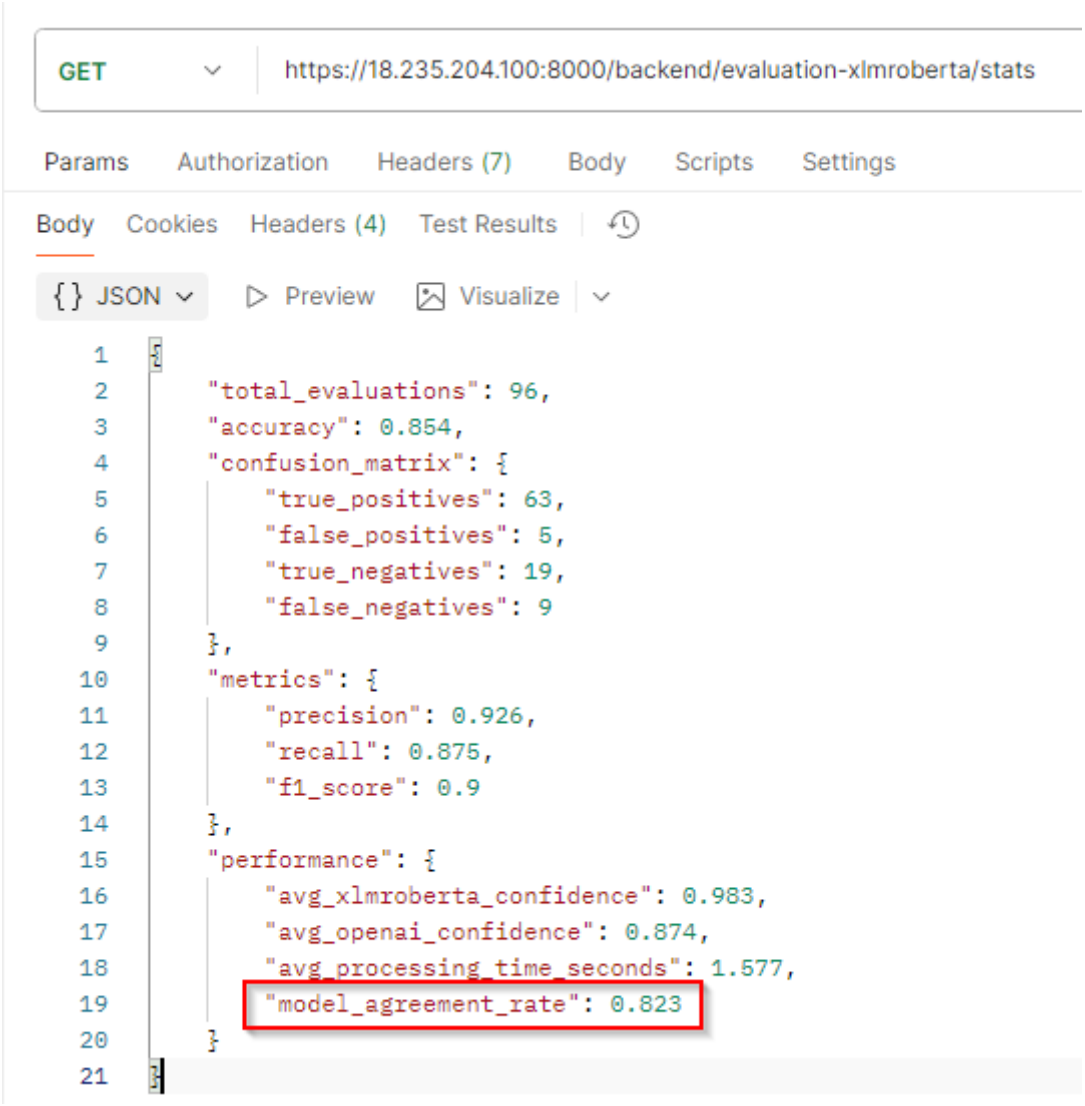


Ilustración 103 – Ejemplo de respuesta de endpoint para consulta de métricas de modelo Roberta

Estadísticas Distilbert


```
{
  "total_evaluations": 18,
  "accuracy": 0.333,
  "confusion_matrix": {
    "true_positives": 2,
    "false_positives": 0,
    "true_negatives": 4,
    "false_negatives": 12
  },
  "metrics": {
    "precision": 1.0,
    "recall": 0.143,
    "f1_score": 0.25
  },
  "performance": {
    "avg_distilbert_confidence": 0.87,
    "avg_openai_confidence": 0.917,
    "avg_processing_time_seconds": 2.366,
    "model_agreement_rate": 0.167
  }
}
```


Ilustración 104 – Ejemplo con resultado de métricas para modelo Distilbert

Estadísticas Roberta


```
{
  "total_evaluations": 10,
  "accuracy": 0.7,
  "confusion_matrix": {
    "true_positives": 5,
    "false_positives": 0,
    "true_negatives": 2,
    "false_negatives": 3
  },
  "metrics": {
    "precision": 1.0,
    "recall": 0.625,
    "f1_score": 0.769
  },
  "performance": {
    "avg_xlmroberta_confidence": 0.922,
    "avg_openai_confidence": 0.79,
    "avg_processing_time_seconds": 5.565,
    "model_agreement_rate": 0.7
  }
}
```


Ilustración 105 - Ejemplo con resultado de métricas para modelo Roberta

Dashboard Grafana

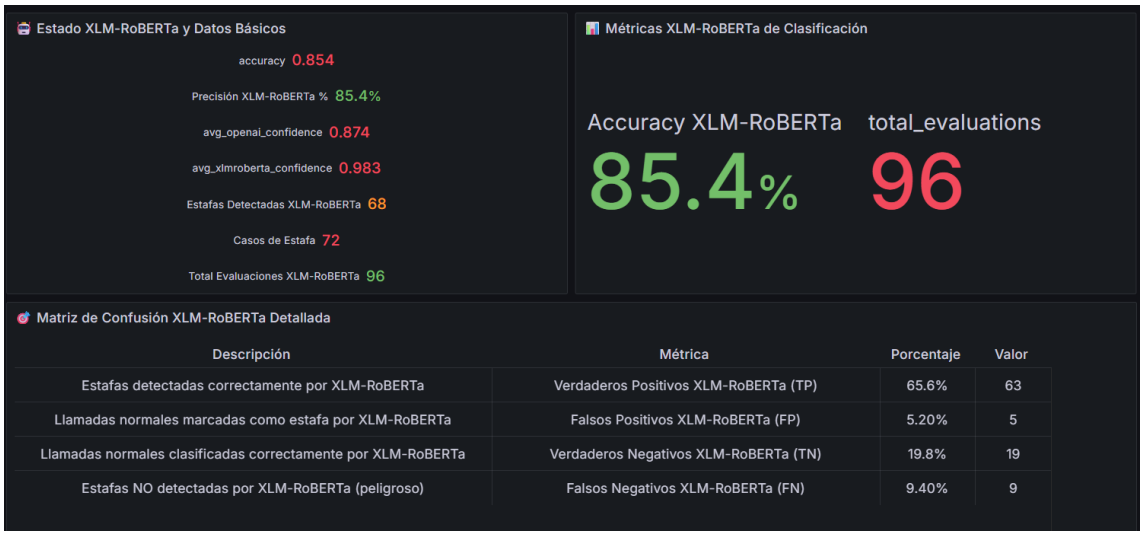


Ilustración 106 – Captura de dashboard en Grafana con métricas de modelo Roberta (en estado de evaluación)

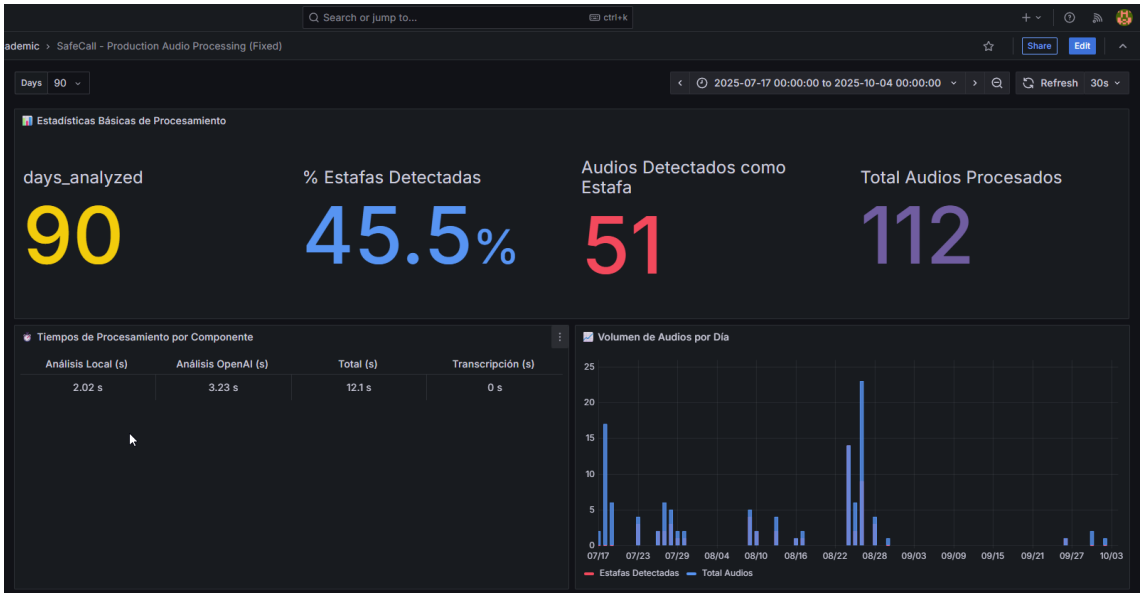


Ilustración 107 - Captura de Dashboard en Grafana con métricas de modelo Roberta (en estado de producción)

5.2 Pruebas

Pruebas Release 1 (06/06/2025 - 03/07/2025)

Usuario	Tarea	Completó la tarea	Errores	Satisfacción (1-5)	Comentarios
U1 (80 años, Bajo)	Tarea 1	Sí	Con muchos errores	3	Le pareció bien la alerta sonora, pero le parecía mejor un cartel más grande para la alerta. Necesitó ayuda para encontrar el detalle.
U1 (80 años, Bajo)	Tarea 2	Sí	Errores mínimos	4	Navegó intuitivamente, pero tardó en identificar el ícono del historial en el menú (muy chico)
U1 (80 años, Bajo)	Tarea 3	No	No la completó	2	No comprendió cómo agregar un número a la blacklist. El flujo de ingreso manual fue muy complejo para ella ya que no recordaba el número
U2 (47 años, Medio)	Tarea 1	Sí	Errores mínimos	4	Completó la tarea rápidamente, pero le pareció muy chico el cartel de alerta.
U2 (47 años,	Tarea 2	Sí	Sin errores	5	Navegación fluida,

Medio)					encontró todo de manera intuitiva.
U2 (47 años, Medio)	Tarea 3	Sí	Errores mínimos	4	Agregó el número, pero preguntó si podía seleccionar desde contactos en lugar de escribir manualmente.
U3 (19 años, Bajo-Medio)	Tarea 1	Sí	Sin errores	5	Completó sin dificultad, familiarizada con notificaciones de apps.
U3 (19 años, Bajo-Medio)	Tarea 2	Sí	Sin errores	5	Uso natural del menú de navegación y de movilidad en listado.
U3 (19 años, Bajo-Medio)	Tarea 3	Sí	Errores mínimos	4	Tardó en encontrar dónde agregar números, pero lo logró sin ayuda. Cambiaria la apariencia
U4 (62 años, Bajo)	Tarea 1	Sí	Con muchos errores	3	Le costó ver bien la alerta por el tamaño. Necesitó explicación verbal del equipo.
U4 (62 años, Bajo)	Tarea 2	Sí	Con muchos errores	3	Se confundió con los íconos del menu. No los veia bien y apretó en lugares equivocados varias veces. No está familiarizado con el

					estilo de menú
U4 (62 años, Bajo)	Tarea 3	No	No la completó	2	No entendió el concepto de "agregar a blacklist". El teclado numérico lo frustró. Pidió hacerlo “más fácil”.
U5 (58 años, Bajo-Medio)	Tarea 1	Sí	Errores mínimos	4	Completó la tarea con errores minimos pero comentó que la pantalla de detalle tenía demasiada información.
U5 (58 años, Bajo-Medio)	Tarea 2	Sí	Sin errores	5	Sin dificultades, navegación clara.
U5 (58 años, Bajo-Medio)	Tarea 3	Sí	Errores mínimos	3	Logró agregar el número, pero preguntó si se podían elegir sin tener que acordarse del número de memoria.

Tabla 34 - Pruebas de release 1

Pruebas Release 2 (04/07/2025 - 31/07/2025)

Usuario	Tarea	Completó la tarea	Errores	Satisfacción (1-5)	Comentarios
U1 (80 años, Bajo)	Tarea 1	Sí	Errores mínimos	4	Buscó contacto con la barra de búsqueda tras indicarle dónde estaba. Comentó que es "muy útil" para avisar a su hija.
U1 (80 años, Bajo)	Tarea 2	Sí	Sin errores	5	Recibió el mensaje y lo mostró emocionada. Le gustó la tranquilidad de que su familia esté al tanto.
U1 (80 años, Bajo)	Tarea 3	Sí	Con muchos errores	3	Le costó encontrar desde donde hacerlo. Una vez encontrado le pareció bueno tener un solo botón
U1 (80 años, Bajo)	Tarea 4	Sí	Errores mínimos	4	Reprodujo la grabación sin problemas, pero necesitó minima ayuda para compartir por WhatsApp.
U2 (47 años, Medio)	Tarea 1	Sí	Sin errores	5	Completó rápidamente. Sugirió poder agregar varios contactos de una vez.

U2 (47 años, Medio)	Tarea 2	Sí	Sin errores	5	Preguntó si el contacto podía responder al mensaje.
U2 (47 años, Medio)	Tarea 3	Sí	Sin errores	5	Denunció y bloqueó sin dificultades. Le gustó que se llegue desde los detalles y se haga en un solo paso
U2 (47 años, Medio)	Tarea 4	Sí	Sin errores	5	Compartió por email. Preguntó si se podía recortar el audio antes de compartir.
U3 (19 años, Bajo-Medio)	Tarea 1	Sí	Sin errores	5	Agregó múltiples contactos sin dificultad. Pidió la opción de seleccionar varios a la vez.
U3 (19 años, Bajo-Medio)	Tarea 2	Sí	Sin errores	5	Le pareció una buena opción que le llegue por SMS y con el número de quien llamó.
U3 (19 años, Bajo-Medio)	Tarea 3	Sí	Sin errores	5	Ejecución fluida, ninguna dificultad.
U3 (19 años, Bajo-Medio)	Tarea 4	Sí	Sin errores	5	Compartió rápidamente. Sugirió botones más simples y tambien poder eliminar

					audios.
U4 (62 años, Bajo)	Tarea 1	Sí	Sin errores	5	Ejecucion fluida, ninguna dificultad
U4 (62 años, Bajo)	Tarea 2	Sí	Errores mínimos	4	Pidió feedback visual de que el mensaje fue enviado.
U4 (62 años, Bajo)	Tarea 3	Sí	Errores minimos	3	Le fue facil acceder y le gusto que se usara solo un botón. Preguntó cuál era la diferencia entre ambas opciones
U4 (62 años, Bajo)	Tarea 4	No	No la completó	2	No supo cómo compartir. No está familiarizado con la opción y el menú de Android lo confundió completamente.
U5 (58 años, Bajo-Medio)	Tarea 1	Sí	Sin errores	5	Encontró la búsqueda por nombre parcial muy útil. Lo consideró rápido y sencillo.
U5 (58 años, Bajo-Medio)	Tarea 2	Sí	Errores mínimos	4	Sugirió que el mensaje incluya fecha y hora de la llamada.

U5 (58 años, Bajo-Medio)	Tarea 3	Sí	Errores mínimos	4	Logró hacerlo, pero pidió confirmación visual más clara de la acción realizada.

Tabla 35 - Pruebas de release 2

Pruebas Release 3 (01/08/2025 - 29/08/2025)

Usuario	Tarea	Completó la tarea	Errores	Satisfacción (1-5)	Comentarios
U1 (80 años, Bajo)	Tarea 1	Sí	Errores mínimos	4	Ya conocía el proceso del R2. Solo necesitó recordar como agregar.
U1 (80 años, Bajo)	Tarea 2	Sí	Con muchos errores	3	No entendió qué era "Modo Guardian". Necesitó explicación del concepto y dónde activarlo.
U1 (80 años, Bajo)	Tarea 3	Sí	Sin errores	5	Botón de eliminar fue claro y directo. Le gustó el mensaje de confirmación.
U1 (80 años, Bajo)	Tarea 4	Sí	Sin errores	5	Cerrar la app fue intuitivo. Comentó que ahora se siente más cómoda usando la aplicación.

U2 (47 años, Medio)	Tarea 1	Sí	Sin errores	5	Ejecución rápida, sin errores
U2 (47 años, Medio)	Tarea 2	Sí	Sin errores	5	Activó el modo y recibió la alerta al recibir llamada sin problemas. Valoró la opción como útil para no ser molestado.
U2 (47 años, Medio)	Tarea 3	Sí	Sin errores	5	Eliminó grabación sin dificultad.
U2 (47 años, Medio)	Tarea 4	Sí	Sin errores	5	Sin comentarios, ejecución fluida.
U3 (19 años, Bajo-Medio)	Tarea 1	Sí	Sin errores	5	Completó en segundos, proceso ya internalizado.
U3 (19 años, Bajo-Medio)	Tarea 2	Sí	Sin errores	5	Probó el modo sin dificultades. Sugirió notificación más destacada cuando está activo y posible nombre mas explicativo
U3 (19 años, Bajo-Medio)	Tarea 3	Sí	Sin errores	5	Eliminó varias grabaciones rápidamente.
U3 (19 años, Bajo-Medio)	Tarea 4	Sí	Sin errores	5	Sin dificultades.

U4 (62 años, Bajo)	Tarea 1	Sí	Errores mínimos	4	Mejoró respecto a R2, pero aún necesitó un momento para recordar cómo hacerlo.
U4 (62 años, Bajo)	Tarea 2	Sí	Con muchos errores	3	Confundió "Modo Guardian" con la configuración de blacklist. Tardó en encontrar dónde activarlo.
U4 (62 años, Bajo)	Tarea 3	Sí	Sin errores	5	Botón claro y visible. No tuvo problemas.
U4 (62 años, Bajo)	Tarea 4	Sí	Sin errores	5	Salió de la app correctamente. Sin problemas
U5 (58 años, Bajo-Medio)	Tarea 1	Sí	Sin errores	5	Proceso ya conocido, ejecución sin errores.
U5 (58 años, Bajo-Medio)	Tarea 2	Sí	Errores mínimos	4	Activó el modo correctamente. Preguntó si podía cambiarse el nombre y estar en un lugar mas accesible
U5 (58 años, Bajo-Medio)	Tarea 3	Sí	Sin errores	5	Eliminó grabación sin inconvenientes.

Tabla 36 - Pruebas de release 3

5.3 Calidad de Código

SonarQube

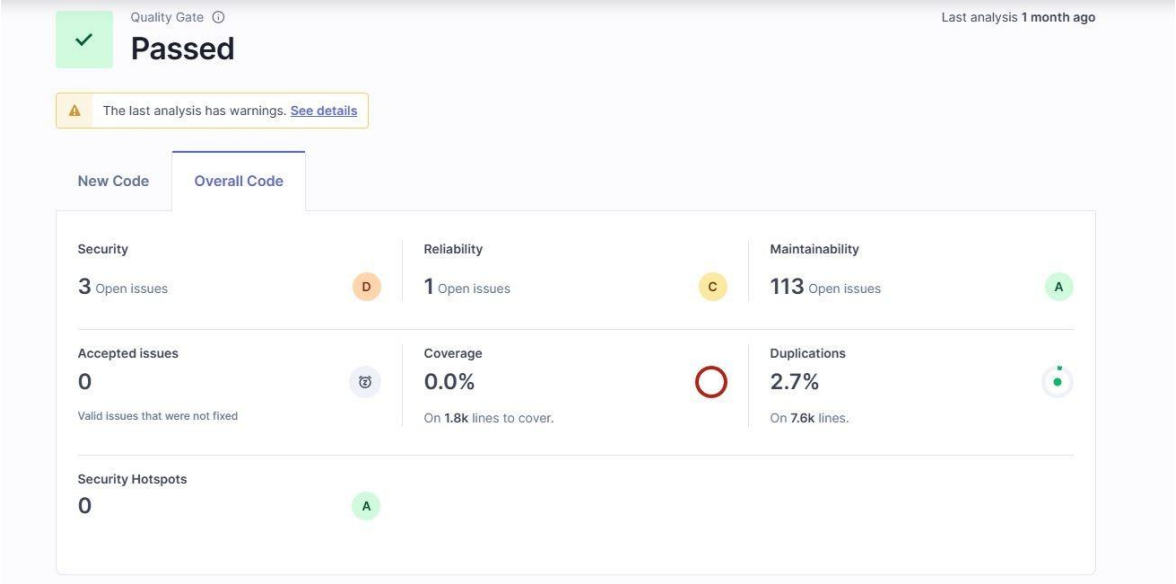


Ilustración 108 108 - Vista general de calidad del Frontend en SonarQube

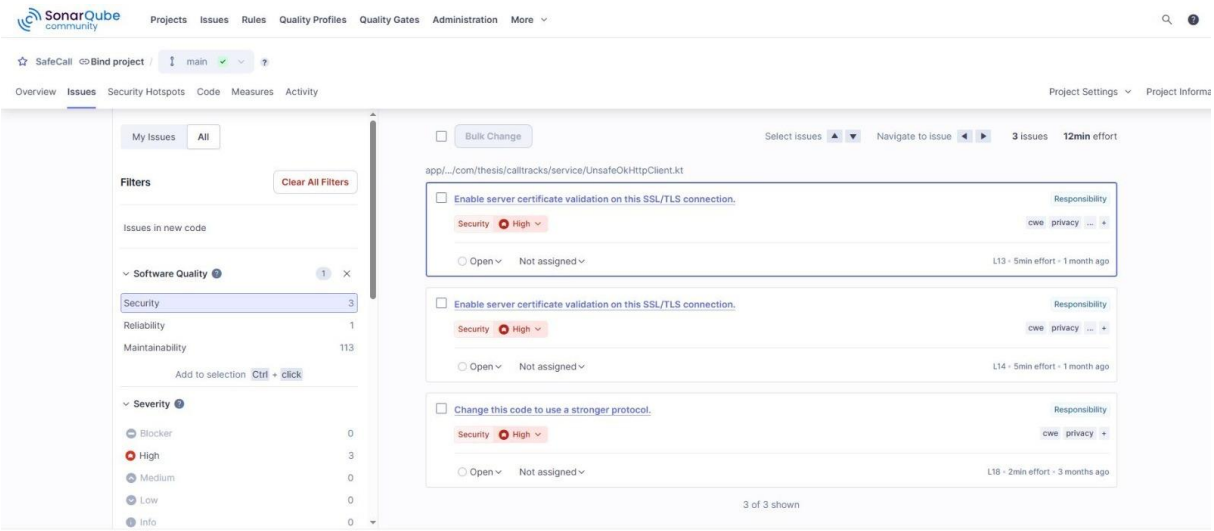


Ilustración 109 109 - Vulnerabilidades detectadas por SonarQube en Frontend

5.4 Gestión de la Configuración

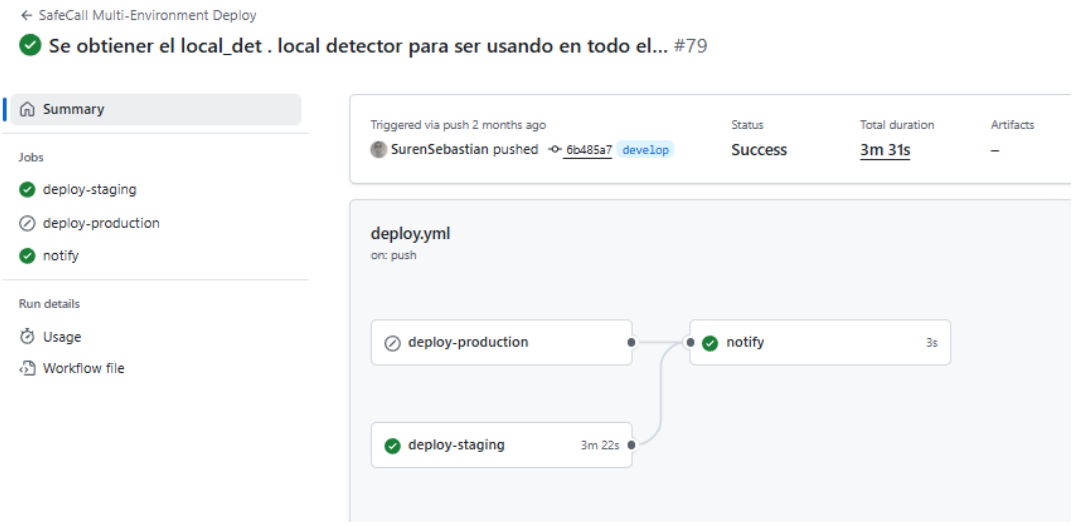


Ilustración 110 110 - Ci/Cd de push a Develop

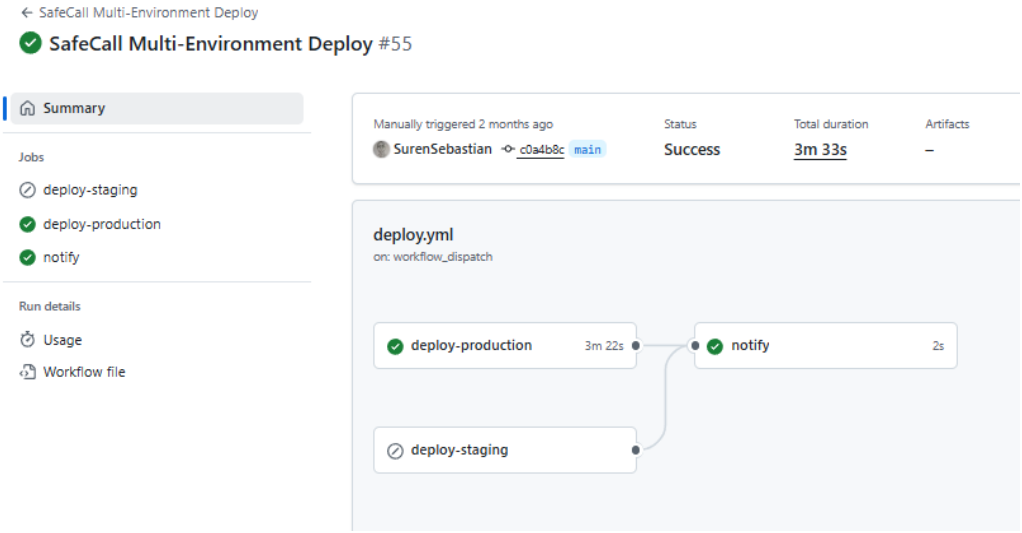


Ilustración 111 111 - Ci/Cd de push a Main

ANEXO 6

6.1 Metodología de Trabajo

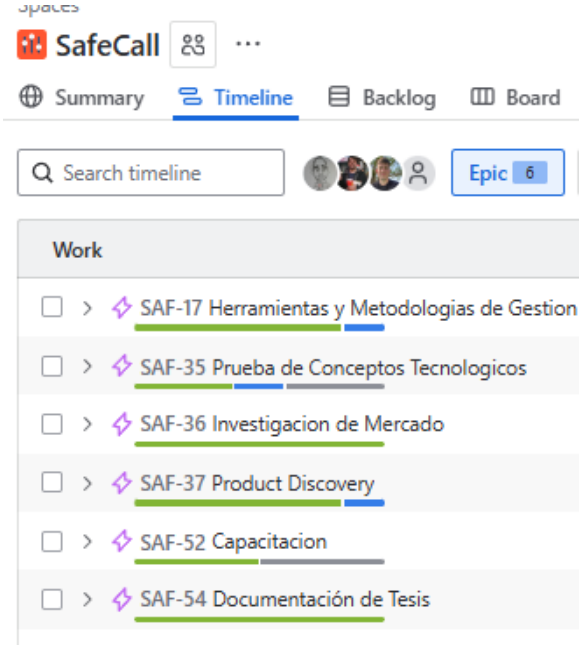


Ilustración 112112 - Épicas Kanban

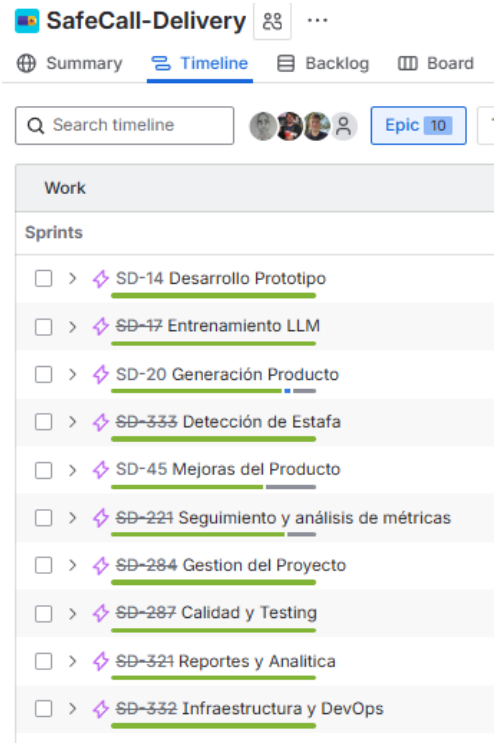


Ilustración 113 113 - Épicas Scrum

Historias de Usuario

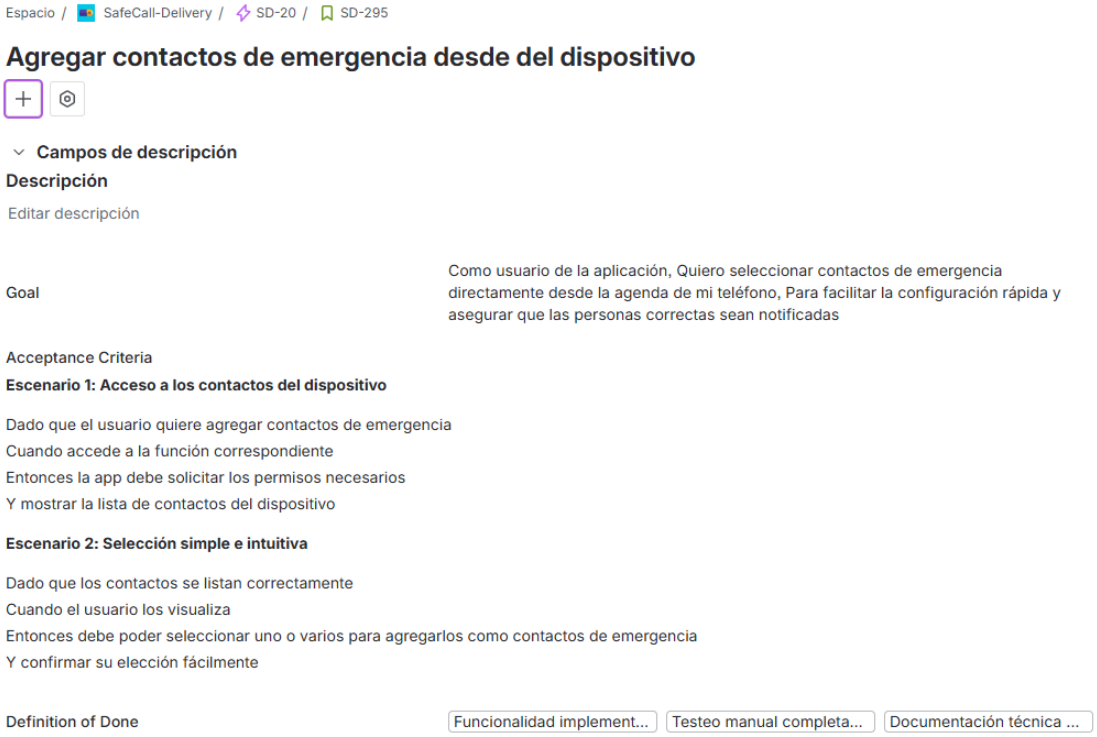


Ilustración 114 114 - Ejemplo Historia de Usuario en Scrum

6.2 Gestión de Sprints

Sprint	Estado	Fecha Inicio	Fecha Fin	Duración (días)
Sprint 1	Cerrado	20-feb-2025	06-mar-2025	14
Sprint 2	**Anómalo**	-	-	0
Sprint 3	Cerrado	06-mar-2025	20-mar-2025	14
Sprint 4	Cerrado	20-mar-2025	03-abr-2025	14
Sprint 5	Cerrado	03-abr-2025	17-abr-2025	14
Sprint 6	Cerrado	18-abr-2025	30-abr-2025	12
Sprint 7	Cerrado	01-may-2025	08-may-2025	7
Sprint 8	Cerrado	08-may-2025	15-may-2025	7
Sprint 9	Cerrado	15-may-2025	22-may-2025	7
Sprint 10	Cerrado	23-may-2025	30-may-2025	7
Sprint 11	Cerrado	30-may-2025	06-jun-2025-	7
Sprint 12	Cerrado	06-jun-2025	13-jun-2025	7
Sprint 13	Cerrado	13-jun-2025	20-jun-2025	7
Sprint 14	Cerrado	20-jun-2025	27-jun-2025	7
Sprint 15	Cerrado	27-jun-2025	4-jul-2025	7
Sprint 16	Cerrado	04-jul-2025	11-jul-2025	7
Sprint 17	Cerrado	11-jul-2025	18-jul-2025	7
Sprint 18	Cerrado	18-jul-2025	25-jul-2025	7
Sprint 19	Cerrado	25-jul-2025	01-ago-2025	7
Sprint 20	Cerrado	01-ago-2025	08-ago-2025	7
Sprint 21	Cerrado	08-ago-2025	15-ago-2025	7
Sprint 22	Cerrado	15-ago-2025	22-ago-2025	7

Tabla 37 - Cronograma de Sprints

Nota: El Sprint 2 fue anómalo y quedó con duración 0 por error en el uso de la herramienta Jira.

Retrospectivas

Ilustración 115115 - Listado de Retrospectivas

Retro 06-13

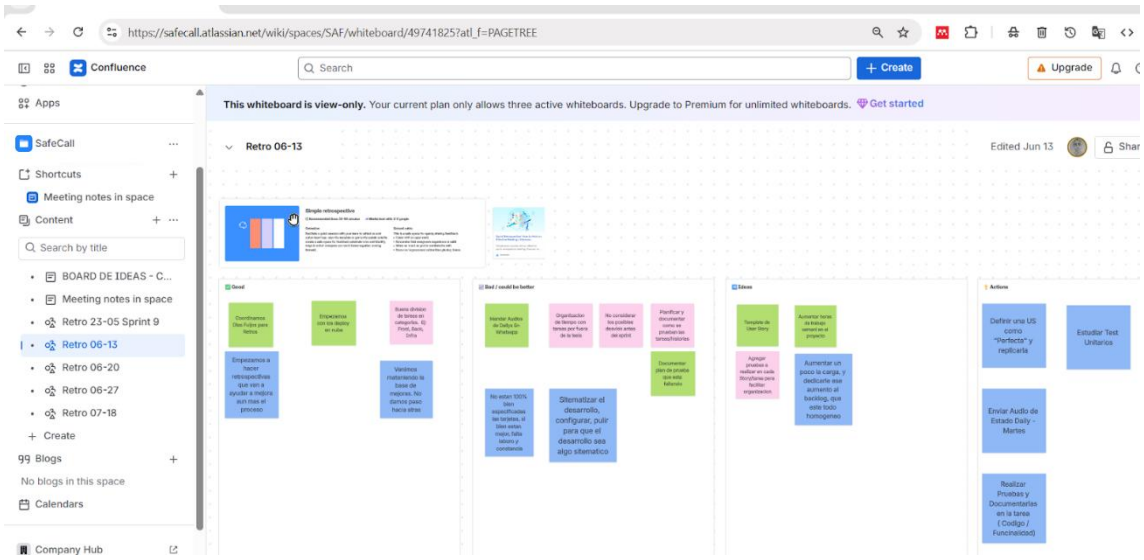


Ilustración 116 116 - Tablero Retrospectiva 13 de Junio 2025

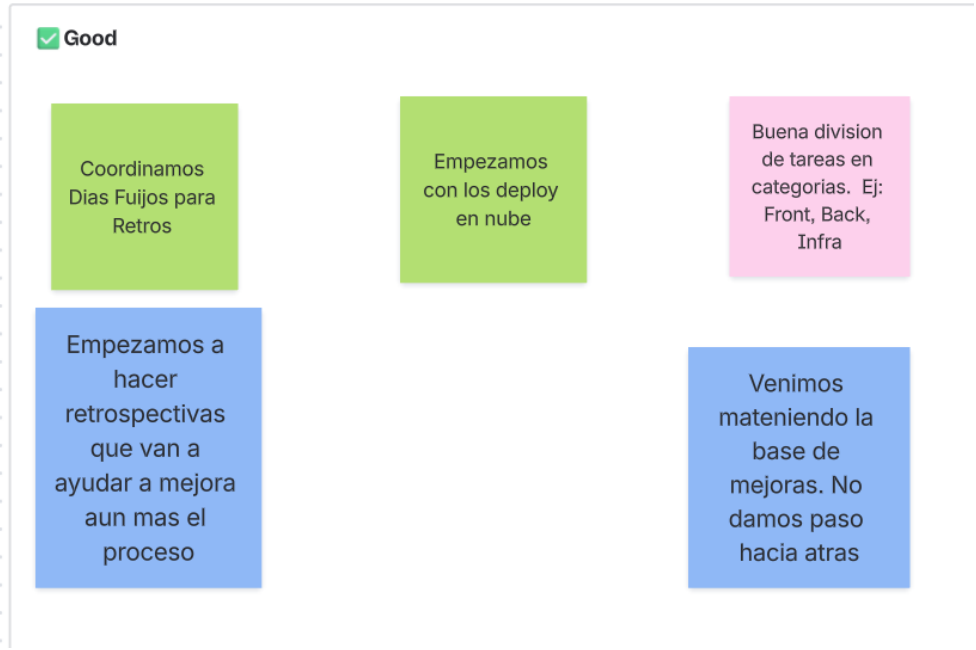


Ilustración 117 117 - Notas "¿Que salió bien?"

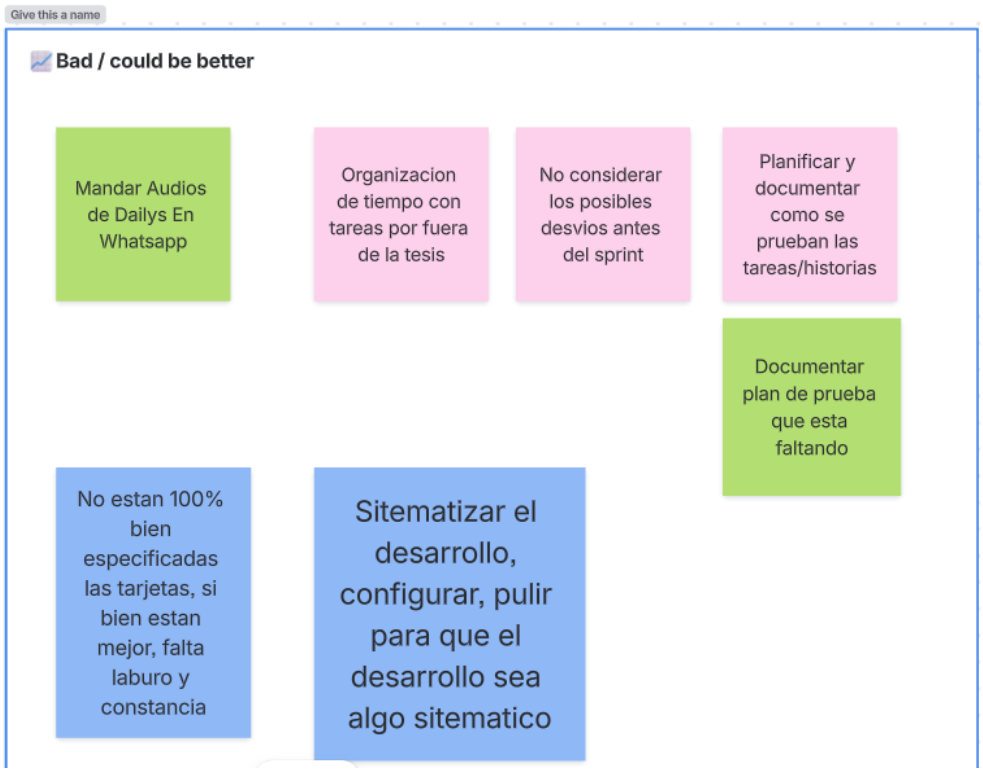


Ilustración 118 118 - Notas "¿Que se puede mejorar?"

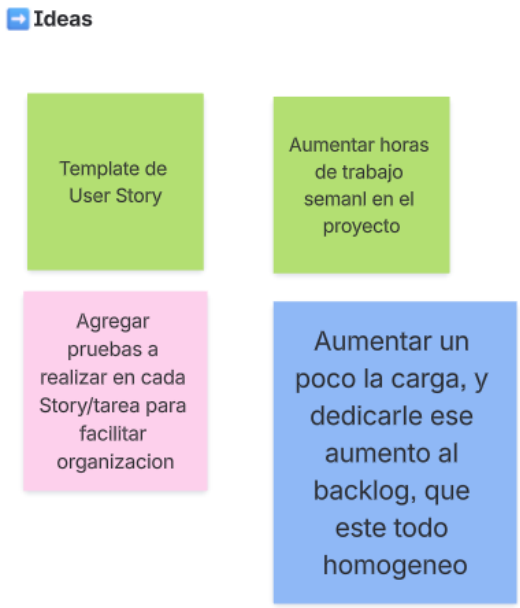


Ilustración 119 - Notas "Ideas"

Ilustración 120 119 - Notas "Acciones"

Retro 6-20

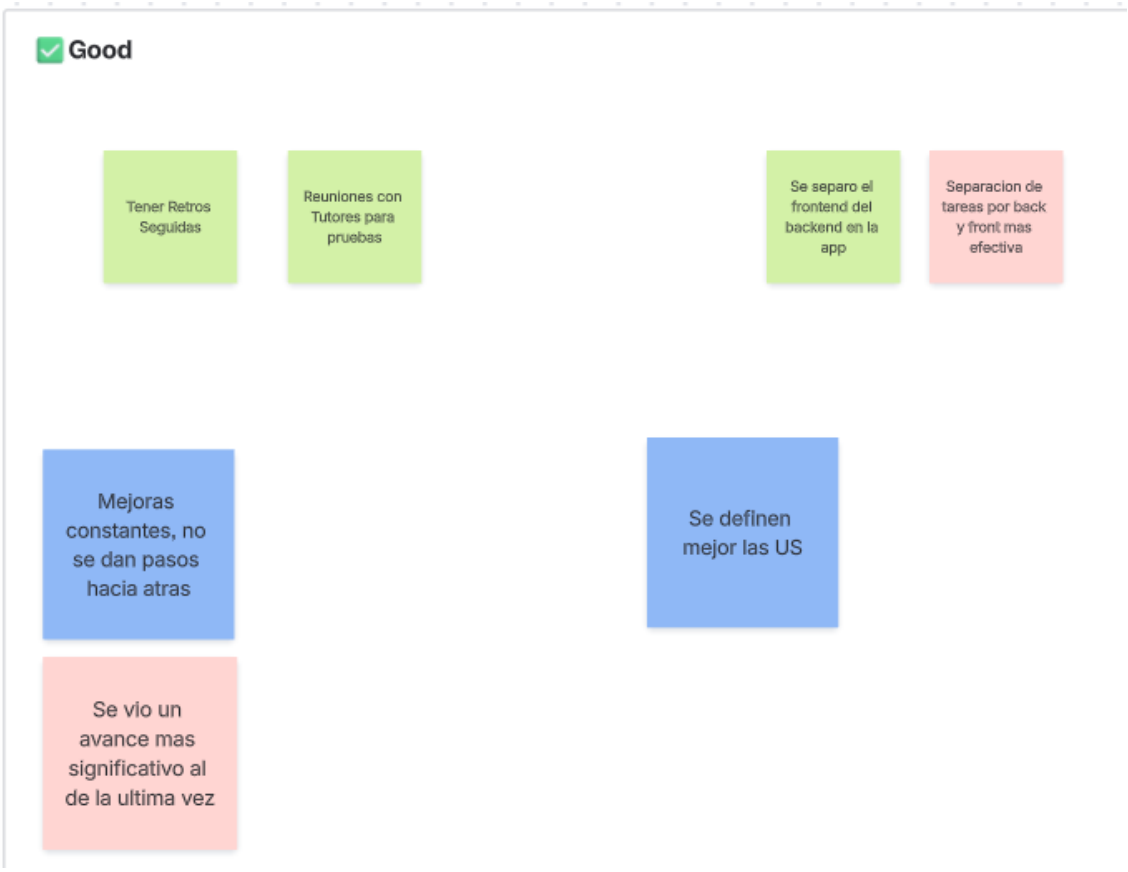


Ilustración 121 120 - Notas "¿Que salió bien?"

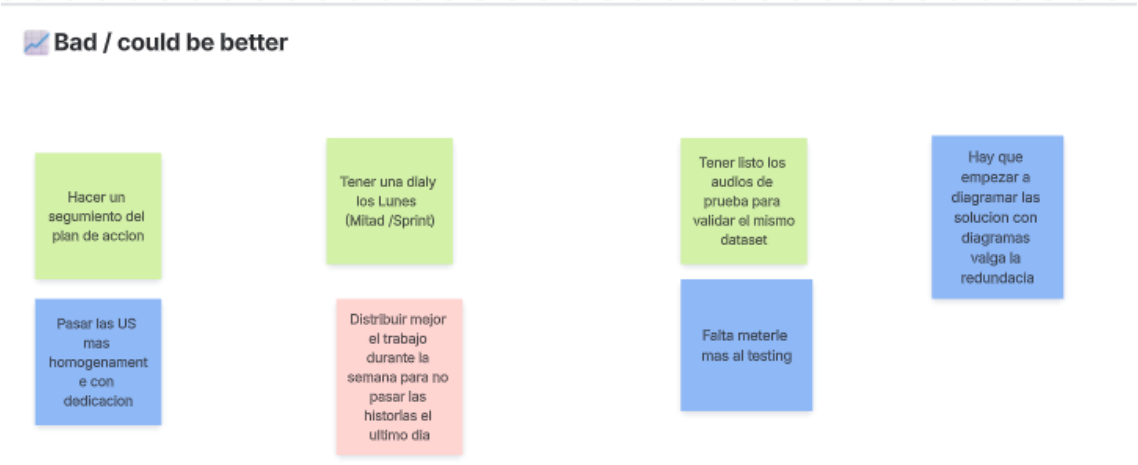


Ilustración 121 122 - Notas "¿Que se puede mejorar?"

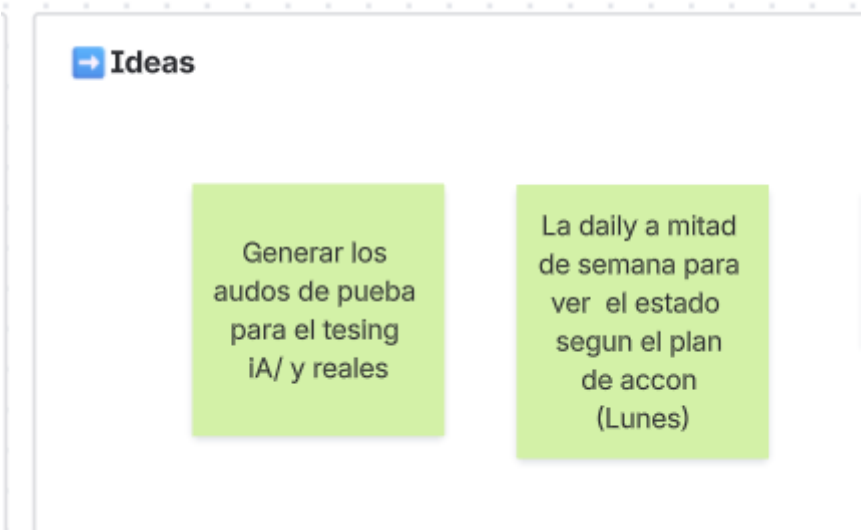


Ilustración 122123 - "Ideas"

6.3 Riesgos de Atributos de calidad

Las siguientes tablas presentan los riesgos identificados por atributo de calidad, priorizando aquellos de mayor impacto para la población objetivo.

Leyenda de Estados

- Mitigado/Controlado/Eliminado: Riesgo actualmente bajo control con estrategias implementadas y validadas
- Identificado/Aceptado: Riesgo conocido y aceptado conscientemente con justificación documentada en ADRs
- Planificado/Monitoreado: Riesgo con plan de acción futuro documentado o bajo monitoreo continuo

Riesgos de Usabilidad

Código	Descripción del Riesgo	Probabilidad	Impacto	Estado
R-U-01	Interfaz demasiado compleja para adultos mayores	Alta	Alto	Mitigado
R-U-02	Flujo de configuración inicial confuso	Alta	Alto	Mitigado
R-U-03	Terminología técnica no comprensible por el usuario objetivo	Media	Medio	Mitigado
R-U-04	Tamaño de texto e iconos inadecuado para visibilidad	Media	Alto	Mitigado
R-U-05	Notificaciones no visibles en situaciones de estrés	Media	Alto	Mitigado
R-U-06	Dificultad de entendimiento por exceso de información en interfaz	Media	Alto	Mitigado

Tabla 38 - Riesgos de Usabilidad

Estrategias de Mitigación - Usabilidad

Código	Estrategia de Mitigación
R-U-01	Validación con 2 usuarios de 60-80 años. Diseño específico con texto grande y confirmación audible.
R-U-02	Proceso guiado paso a paso con íconos simples y autodescriptivos. Iteraciones basadas en feedback de usuarios objetivo.

R-U-03	Uso de lenguaje coloquial y familiar. Eliminación de términos técnicos de ML/IA en la interfaz.
R-U-04	Tipografías legibles y escalables. Colores contrastantes. Funcionalidad de ajuste de tamaño (RF11).
R-U-05	Notificaciones con colores contrastantes y alertas audibles. Diseño para contextos de vulnerabilidad.
R-U-06	Interfaz minimalista. Reducción de opciones por pantalla. Guías contextuales de ayuda.

Tabla 39 - Estrategia de Mitigación de Usabilidad.

Riesgos de Performance:

Código	**Descripción del Riesgo**	**Probabilidad**	**Impacto**	**Estado**
R-PE-01	Tiempo de análisis post-llamada superior a 30 segundos	Media	Alto	Superado
R-PE-02	Precisión del modelo de detección inferior al 80%	Media	Crítico	Superado
R-PE-03	Uso excesivo de memoria o almacenamiento del dispositivo	Baja	Medio	Controlado
R-PE-04	Degradación de performance con audios largos (>5 min)	Media	Medio	Mitigado
R-PE-05	Falsos positivos que destruyen confianza del usuario	Media	Crítico	Controlado
R-PE-06	Falsos negativos que permiten estafas exitosas	Media	Crítico	Controlado

Tabla 40 - Riesgos de Performance

Estrategias de Mitigación – Performance

Código	Estrategia de Mitigación
R-PE-01	Optimización de modelo ML para móviles.
R-PE-02	Dataset especializado en estafas.
R-PE-03	Modelo no almacenado en dispositivo, ahorrando considerablemente espacio de almacenamiento
R-PE-04	Optimización de procesamiento por segmentos. Testing con audios de diferentes duraciones.
R-PE-05	Refinamiento del Modelo
R-PE-06	Modelo entrenado con patrones específicos de estafas.

Tabla 41 - Estrategias de Mitigación de Performance

Riesgos de Portabilidad

Código	Descripción del Riesgo	Probabilidad	Impacto	Estado
R-PO-01	Dependencia exclusiva de dispositivos Samsung	Alta	Alto	Aceptado
R-PO-02	Limitación a Android 11 o versiones superiores	Media	Medio	Aceptado
R-PO-03	Incompatibilidad de grabación con otros fabricantes	Alta	Alto	Aceptado
R-PO-04	Actualizaciones genéricas constantes de Android	Alta	Alto	Aceptado

Tabla 42 - Riesgos de Portabilidad

Estrategias de Mitigación - Portabilidad

Código	Estrategia de Mitigación
R-PO-01	Evaluar desarrollo de aplicación de Dialer propia
R-PO-02	Decisión consciente basada en APIs de grabación necesarias.
R-PO-03	Analizar APIs alternativas por fabricante (Xiaomi, Motorola, etc)
R-PO-04	No se contaba con estrategia de mitigación para este riesgo

Tabla 43 - Estrategias de Mitigación de Portabilidad

Riesgos de Deployabilidad

Código	Descripción del Riesgo	Probabilidad	Impacto	Estado
R-D-01	Tiempos de despliegue prolongados que retrasan entregas	Media	Alto	Eliminado
R-D-02	Inconsistencias entre entornos de desarrollo, prueba y producción	Media	Alto	Mitigado
R-D-03	Dificultad para realizar rollbacks ante fallas en producción	Media	Alto	Mitigado
R-D-04	Complejidad en gestión de dependencias y configuraciones	Media	Medio	Controlado
R-D-05	Costos operativos elevados de infraestructura en nube	Media	Medio	Controlado
R-D-06	Falta de observabilidad y monitoreo del sistema en producción	Media	Alto	Mitigado

Tabla 44 - Riesgos de Deployabilidad

Estrategias de Mitigación - Deployabilidad

Código	Estrategia de Mitigación
R-D-01	Contenedores Docker para despliegues reproducibles. Pipeline CI/CD automatizado.
R-D-02	Uso de Docker garantiza consistencia entre entornos. Configuración como código.
R-D-03	Versionado de imágenes Docker.
R-D-04	Docker Compose para orquestación. Dependencias declaradas explícitamente.
R-D-05	Despliegue en AWS EC2 optimizado.
R-D-06	Integración con Grafana para métricas en tiempo real. Dashboard de CPU, memoria y tiempos de inferencia.

Tabla 45 - Estrategias Mitigación Deployabilidad

6.4 Riesgos de Gestión de Proyecto

La siguiente tabla presenta los riesgos organizacionales y de gestión identificados durante el proyecto.

Categoría	**Riesgo**	**Impacto**	**Probabilidad**
Alto	Falta de compromiso de un miembro	Alto	Alta
Alto	Falta de organización del equipo	Alto	Alta
Alto	Dificultad de acceso a llamadas	Alto	Alta
Alto	Dificultad en detección del fraude	Alto	Alta
Medio	Conflictos de comunicación	Medio	Media
Medio	Desconocimiento en ML	Medio	Alta
Medio	Dificultad para entrenar modelo	Medio	Media
Medio	Dificultad para fijar fechas	Medio	Alta
Medio	Registro real de horas	Medio	Media
Medio	Tareas universitarias extra tesis	Medio	Alta
Bajo	Diferencias de horario	Bajo	Media
Bajo	Falta de interés del público	Bajo	Media

Tabla 46 - Riesgos de Gestión de Proyecto