



UNIVERSIDAD LAICA “ELOY ALFARO” DE MANABÍ

FACULTAD DE CIENCIAS DE LA VIDA Y TECNOLOGÍAS

CARRERA:

TECNOLOGÍAS DE LA INFORMACIÓN

TEMA:

**DESARROLLO DE UNA APLICACIÓN MÓVIL DE CONTROL DE EJERCICIOS FÍSICOS
PARA JÓVENES PRINCIPIANTES ENTRE 18 -30 AÑOS UTILIZANDO TECNOLOGÍA DE
TRACKING.**

Trabajo de titulación presentado en conformidad con los requisitos establecidos para optar por el
título de Ingenieros en Tecnologías de la Información

AUTORES:

BAQUE BARCIA JOSTHIN ARIEL

ORTEGA GILER MIGUEL ÁNGEL

TUTOR:

ING. PINCAY PONCE JORGE IVAN, PHD.

MANTA-MANABÍ-ECUADOR

2026

Certificación del director de trabajo de graduación

	NOMBRE DEL DOCUMENTO: CERTIFICADO DE TUTOR(A).	CÓDIGO: PAT-04-F-004
	PROCEDIMIENTO: TITULACIÓN DE ESTUDIANTES DE GRADO BAJO LA UNIDAD DE INTEGRACIÓN CURRICULAR	REVISIÓN: 1
		Página 1 de 1

CERTIFICACIÓN

En calidad de docente tutor de la Facultad de Ciencias de la Vida y Tecnología de la Universidad Laica "Eloy Alfaro" de Manabí, CERTIFICO:

Haber dirigido, revisado y aprobado preliminarmente el Trabajo de Integración Curricular bajo la autoría del estudiante **ORTEGA GILER MIGUEL ÁNGEL**, legalmente matriculado en la carrera de tecnologías de la información, período académico 2025 (2), cumpliendo el total de 384 horas, cuyo tema del proyecto es ***"Desarrollo de una aplicación móvil de control de ejercicios físicos para jóvenes principiantes entre 18 -30 años utilizando tecnología de tracking"***.

La presente investigación ha sido desarrollada en apego al cumplimiento de los requisitos académicos exigidos por el Reglamento de Régimen Académico y en concurrencia con los lineamientos internos de la opción de titulación en mención, reuniendo y cumpliendo con los méritos académicos, científicos y formales, y la originalidad de este, requisitos suficientes para ser sometida a la evaluación del tribunal de titulación que designe la autoridad competente.

Particular que certifico para los fines consiguientes, salvo disposición de Ley en contrario.

Manta, 22 de enero de 2026.

Lo certifico,



Ing. Pincay Ponce Jorge Ivan, PhD.

Docente Tutor

**Facultad de Ciencias de la Vida y Tecnología
Carrera Tecnologías de la Información**

Certificación del director de trabajo de graduación

	NOMBRE DEL DOCUMENTO: CERTIFICADO DE TUTOR(A).	CÓDIGO: PAT-04-F-004
	PROCEDIMIENTO: TITULACIÓN DE ESTUDIANTES DE GRADO BAJO LA UNIDAD DE INTEGRACIÓN CURRICULAR	REVISIÓN: 1
		Página 1 de 1

CERTIFICACIÓN

En calidad de docente tutor de la Facultad de Ciencias de la Vida y Tecnología de la Universidad Laica "Eloy Alfaro" de Manabí, CERTIFICO:

Haber dirigido, revisado y aprobado preliminarmente el Trabajo de Integración Curricular bajo la autoría del estudiante **BAQUE BARCIA JOSTHIN ARIEL**, legalmente matriculado en la carrera de tecnologías de la información, período académico 2025 (2), cumpliendo el total de 384 horas, cuyo tema del proyecto es ***"Desarrollo de una aplicación móvil de control de ejercicios físicos para jóvenes principiantes entre 18 -30 años utilizando tecnología de tracking"***.

La presente investigación ha sido desarrollada en apego al cumplimiento de los requisitos académicos exigidos por el Reglamento de Régimen Académico y en concórdancia con los lineamientos internos de la opción de titulación en mención, reuniendo y cumpliendo con los méritos académicos, científicos y formales, y la originalidad de este, requisitos suficientes para ser sometida a la evaluación del tribunal de titulación que designe la autoridad competente.

Particular que certifico para los fines consiguientes, salvo disposición de Ley en contrario.

Manta, 22 de enero de 2026.

Lo certifico,



Ing. Pincay Ponce Jorge Ivan, PhD.

Docente Tutor

**Facultad de Ciencias de la Vida y Tecnología
Carrera Tecnologías de la Información**

Declaración expresa de Autoría

Yo, Miguel Ángel Ortega Giler, con cédula de ciudadanía 0928942390 autor del presente trabajo de titulación, “Desarrollo de una aplicación móvil de control de ejercicios físicos para jóvenes principiantes entre 18 -30 años utilizando tecnología de tracking.”, autorizo a la Universidad Laica Eloy Alfaro de Manabí, hacer el uso total o parcial del presente trabajo de titulación, con fines estrictamente académicos o investigativos.

Los derechos que como autor me corresponden, con la excepción de la presente autorización, seguirán vigente a mi favor, de conformidad con lo establecido en los artículos 5, 6, 8, 19 y demás artículos pertenecientes a la Ley de Propiedad Intelectual y su reglamento.

Así mismo autorizo a la Universidad Laica Eloy Alfaro de Manabí para que realice la digitación y publicación de mi trabajo de titulación en el repositorio virtual, en conformidad a lo establecido en el Art. 144 de la Ley Orgánica de Educación Superior.

Lo certifica



Ortega Giler Miguel Ángel
0928942390

Declaración expresa de Autoría

Yo, Baque Barcia Josthin Ariel, con cédula de ciudadanía 1313512764 autor del presente trabajo de titulación, “Desarrollo de una aplicación móvil de control de ejercicios físicos para jóvenes principiantes entre 18 -30 años utilizando tecnología de tracking.”, autorizo a la Universidad Laica Eloy Alfaro de Manabí, hacer el uso total o parcial del presente trabajo de titulación, con fines estrictamente académicos o investigativos.

Los derechos que como autor me corresponden, con la excepción de la presente autorización, seguirán vigente a mi favor, de conformidad con lo establecido en los artículos 5, 6, 8, 19 y demás artículos pertenecientes a la Ley de Propiedad Intelectual y su reglamento.

Así mismo autorizo a la Universidad Laica Eloy Alfaro de Manabí para que realice la digitación y publicación de mi trabajo de titulación en el repositorio virtual, en conformidad a lo establecido en el Art. 144 de la Ley Orgánica de Educación Superior.

Lo certifica



Baque Barcia Josthin Ariel
1313512764

Dedicatoria

Con profunda gratitud y respeto quiero dedicar este logro a mis padres, Miguel Ángel Ortega Saona e Isabel Marina Giler Cuzco, por haberme apoyado con mi decisión de estudiar esta carrera y por su apoyo que nunca faltó durante todo mi proceso universitario.

A mis hermanos Emilio y Arianna, por tolerar las incontables amanecidas y todas las veces que necesite de su ayuda para probar o evaluar los distintos proyectos que desarrolle a lo largo de la carrera.

A mi compañero de tesis Josthin Ariel Baque Barcia por haber aceptado ser mi compañero durante esta etapa final de nuestro proceso universitario y por ser en general un gran amigo durante todo este trayecto.

También a todos mis compañeros y amigos que conocí durante mi formación, pues gracias a ellos también pude cambiar como persona y por los buenos momentos que pasé con ellos.

Miguel Ángel Ortega Giler

Dedicatoria

Con gran aprecio, gratitud y respeto dedico este gran logro a mis padres, Bolívar Baque Pilay y Mercedes Barcia Menéndez quienes estuvieron conmigo durante todo este viaje apoyándome de todas las maneras posibles, este logro profesional es el resultado de todo lo que han invertido en mí.

A mis diferentes maestros durante toda esta etapa, ya que mis conocimientos son derivados de sus enseñanzas y en general a todos aquellos que me han apoyado para seguir adelante.

Josthin Ariel Baque Barcia

Agradecimiento

La culminación de esta etapa académica es más que la entrega de un proyecto final, es el reflejo de un camino donde perseverar fue la clave para abrir un camino que al inicio parecía difícil de alcanzar.

Mirando hacia atrás, siento satisfacción al saber que pude superar todos los desafíos que se me presentaron a lo largo de la carrera y que cada uno de ellos fue una oportunidad, no sólo para mejorar mis aptitudes profesionales sino también mejorar mi carácter como persona. Este proceso de transformación es el que me permite ver realizado mi sueño de graduarme como profesional de la carrera.

Por ello, mi gratitud enorme a todos los docentes de la Facultad de Ciencias Informáticas, quienes aparte de ser buenos docentes de sus materias respectivas, siempre me ayudaron con feedback con algún proyecto o con ayuda y recomendaciones en general.

Mención especial para el tutor de nuestra tesis el Ing. Pincay Ponce Jorge Iván, PhD, por su conocimiento y la ayuda que nos brindó para realizar nuestro proyecto final.

Y a la Universidad Laica Eloy Alfaro de Manabí, por ser mi segundo hogar durante toda mi formación y haberme proporcionado las herramientas para convertirme en profesional.

Miguel Ángel Ortega Giler

Agradecimiento

El camino hacia la culminación de esta etapa universitaria más que un requisito académico representa un logro más en mi vida y el cierre de una etapa de crecimiento personal.

Soy lo que soy gracias a las personas que han estado a mi lado a lo largo de este trayecto, es por eso que agradezco en primer lugar a Dios por la vida y las oportunidades que se me han presentado, a mis padres por el pilar fundamental de mi existencia que me ha ayudado a seguir adelante en muchas ocasiones.

Asimismo, expreso mi gratitud hacia los docentes de la Facultad de Ciencias de la Vida y Tecnología, y de manera especial a mi tutor, el Ing. Jorge Pincay, PhD. Quien aportó su conocimiento y voluntad para así orientarme en este proyecto.

Por último, agradezco a mis amigos y compañeros con los cuales pasé una gran etapa de mi vida y gracias a esos momentos que hicieron mis días en esta institución bastante agradables.

Josthin Ariel Baque Barcia

Índice de Contenido

Certificación del director de trabajo de graduación.....	V
Certificación del director de trabajo de graduación.....	VI
Tribunal de Sustentación.....	¡Error! Marcador no definido.
Declaración expresa de Autoría.....	VII
Declaración expresa de Autoría.....	VIII
Dedicatoria.....	IX
Dedicatoria.....	X
Agradecimiento.....	XI
Agradecimiento.....	XII
Índice de Contenido.....	XIII
Índice de Tablas.....	XVII
Índice de ilustraciones.....	XIX
Resumen.....	20
Abstract.....	21
CAPITULO 1: MARCO INTRODUCTORIO.....	23
1.1 Introducción.....	23
1.2 Ubicación y Contextualización del Problema.....	24
1.3 Planteamiento del Problema.....	25
1.4 Objetivos.....	26
1.4.1 Objetivo General.....	26
1.4.2 Objetivos Específicos.....	26
1.5 Justificación.....	27
CAPÍTULO 2: MARCO TEÓRICO.....	29
2.1. Antecedentes de investigaciones relacionadas.....	29

2.2. Definiciones conceptuales.....	30
2.2.1. Actividad Física	30
2.2.2. Ejercicio Físico	30
2.2.3. Visión por Computadora (Computer Vision)	31
2.2.4. Estimación de Pose Humana, Pose estimación.....	32
2.2.5 Seguimiento de Movimiento, Motion Tracking.....	32
2.2.6. MediaPipe	33
2.2.7 Aplicaciones de Fitness.....	34
2.2.8 Kotlin	34
2.2.9 Android Studio.....	35
2.2.10 Jetpack Navigation.....	36
2.2.11 Text to Speech (TTS).....	37
2.2.12 Firebase	37
2.2.13 Firebase Authentication	37
2.2.14 Cloud Firestore.....	38
2.2.15 Cloud Storage for Firebase	38
2.2.16 Glide.....	39
2.2.17 AndroidChart	39
2.2.18 Material Calendar Design	39
2.2.19 Ucrop.....	39
2.3. Conclusiones relacionadas al marco teórico	39
CAPÍTULO 3: MARCO INVESTIGATIVO	42
3.1 Tipos de Investigación	42
3.1.1 Investigación Aplicada.....	42
3.2 Métodos de investigación.....	42

3.2.1 Método deductivo	42
3.2.2 Método Inductivo.....	43
3.3 Herramientas de recolección de datos	43
3.4 Fuentes de Información.....	43
3.4.1 Fuentes Primarias.....	43
3.4.2 Fuentes secundarias	44
3.5 Instrumental Operacional	44
3.5.1 Estructura y características de los instrumentos de recolección de datos	44
3.6 Caracterización del Usuario Objetivo	44
3.7 Estrategia Operacional para la recolección y tabulación de datos	45
CAPÍTULO 4: MARCO PROPOSITIVO	49
4.1 Introducción	49
4.2 Descripción de la propuesta	50
4.3 Determinación de Recursos.....	50
3.3.1 Recursos Humanos.....	50
3.3.2 Recursos Tecnológicos	50
3.3.3 Recursos Económicos	52
4.4 Etapas de acción para el desarrollo de la propuesta	52
4.4.1 Análisis	52
4.4.2 Diseño	56
4.4.3 Implementación.....	76
4.4.4 Pruebas.....	86
CAPÍTULO 5: EVALUACIÓN DE RESULTADOS	88
5.1. Introducción	88
5.2. Evaluación y monitoreo de resultados	88

5.2.1. Resultados con respecto a la funcionalidad	88
5.2.2. Resultados con respecto a la usabilidad.....	93
5.2.3. Resultados con respecto a la eficiencia.....	93
5.2.4. Resultados con respecto a la portabilidad.....	94
5.3 Análisis de Impacto y Validación frente a la Entrevista inicial	94
Conclusiones	95
Recomendaciones	96
Trabajos Futuros	97
Referencias.....	98
Anexos	101

Índice de Tablas

Tabla 1 <i>Matriz de Hallazgos y Soluciones de la entrevista realizada</i>	45
Tabla 2 <i>Recursos Humanos necesarios</i>	50
Tabla 3 <i>Recursos Tecnológicos necesarios</i>	50
Tabla 4 <i>Recursos Economicos necesarios</i>	52
Tabla 5 <i>Requerimiento de Información IQR_1</i>	54
Tabla 6 <i>Requerimiento de Información IQR_2</i>	55
Tabla 7 <i>Requerimiento de Información IQR_3</i>	55
Tabla 8 <i>Componentes principales del sistema</i>	59
Tabla 9 <i>Descripción de Colecciones</i>	61
Tabla 10 <i>Caso de uso de usuarios del sistema</i>	63
Tabla 11 <i>Caso de uso de Registro de Usuarios</i>	64
Tabla 12 <i>Caso de uso de Ingreso del Usuario</i>	65
Tabla 13 <i>Caso de Uso de creación de ejercicio</i>	66
Tabla 14 <i>Caso de Uso de edición de ejercicio</i>	67
Tabla 15 <i>Caso de Uso de eliminación de ejercicio</i>	67
Tabla 16 <i>Caso de Uso de consulta de ejercicio</i>	70
Tabla 17 <i>Caso de Uso de creación de rutina</i>	70
Tabla 18 <i>Caso de Uso de edición de rutina</i>	71
Tabla 19 <i>Caso de Uso de eliminación de rutina</i>	71
Tabla 20 <i>Caso de Uso de consulta de rutina</i>	72
Tabla 21 <i>Matriz de Trazabilidad de Requisitos</i>	74
Tabla 22 <i>Prueba de aceptación crear estudiante</i>	88
Tabla 23 <i>Prueba de aceptación de ingreso de usuario</i>	89
Tabla 24 <i>Prueba de aceptación de creación de ejercicios</i>	89
Tabla 25 <i>Prueba de aceptación edición de ejercicios</i>	90
Tabla 26 <i>Prueba de aceptación eliminación de ejercicio</i>	90
Tabla 27 <i>Prueba de aceptación consultar ejercicios</i>	90
Tabla 28 <i>Prueba de aceptación crear rutinas</i>	91
Tabla 29 <i>Prueba de aceptación edición de rutina</i>	91
Tabla 30 <i>Prueba de aceptación de eliminación de rutina</i>	92

Tabla 31 <i>Prueba de aceptación visualización de rutina</i>	92
---	----

Índice de ilustraciones

Ilustración 1 <i>Modelo de Marcador de Posición</i>	33
Ilustración 2 <i>Estructura de Cascada Modificada</i>	49
Ilustración 3 <i>Diagrama de Arquitectura del Sistema</i>	58
Ilustración 4 <i>Diseño lógico de base de datos</i>	60
Ilustración 5 <i>Caso de uso de usuarios del sistema</i>	62
Ilustración 6 <i>Caso de uso de Registro de Usuarios</i>	63
Ilustración 7 <i>Caso de uso de Ingreso del Usuario</i>	65
Ilustración 8 <i>Caso de Uso de operaciones del Admin</i>	66
Ilustración 9 <i>Caso de Uso de Operaciones del Usuario</i>	69
Ilustración 10 <i>Secuencia lógica del proceso de Tracking</i>	73
Ilustración 11 <i>Interfaz de Inicio de Sesión</i>	76
Ilustración 12 <i>Interfaz de Registro de cuenta</i>	77
Ilustración 13 <i>Interfaz de perfil del usuario</i>	78
Ilustración 14 <i>Interfaz de las rutinas del usuario</i>	79
Ilustración 15 <i>Interfaz de creación de rutinas del usuario</i>	80
Ilustración 16 <i>Interfaz de ejercicios</i>	81
Ilustración 17 <i>Interfaz del ejercicio asistido</i>	82
Ilustración 18 <i>Interfaz de progreso de ejercicios</i>	83
Ilustración 19 <i>Interfaz de Gestión de ejercicios del Administrador</i>	84
Ilustración 20 <i>Interfaz de gestión de usuarios del Administrador</i>	85

Resumen

Contexto: El entrenamiento físico para principiantes suele carecer de supervisión técnica, lo que incrementa el riesgo de lesiones y desmotivación. Este proyecto integrador aborda el problema mediante el desarrollo de una aplicación móvil de fácil acceso, que democratiza la orientación de la técnica del ejercicio, especialmente en personas de nivel principiante.

Métodos: Se empleó una investigación descriptiva y aplicada junto con metodología de cascada modificada. El proyecto se desarrolló en Kotlin usando Android Studio, integrando la biblioteca de MediaPipe para rastreo de 33 puntos corporales y Firebase, junto con sus recursos Auth, Firestore y Storage.

Resultados: Las pruebas controladas determinaron que el procesamiento local de visión por computadora garantiza latencia mínima durante la detección de poses. La interfaz con base en Material Design hizo más fácil la interacción a manos libres y la implementación de texto a voz permitió correcciones de manera auditiva sin interrumpir el flujo del ejercicio.

Conclusiones: Es posible usar sistemas de inteligencia artificial de tracking en dispositivos móviles para que funcionen como guías virtuales. La arquitectura Model-View-ViewModel, que se implementó, se logró una solución escalable que mejora la seguridad y autonomía del joven principiante.

Abstract

Context: Physical training for beginners frequently lacks professional technical supervision, which increases the risk of injury and lack of motivation. This project addresses this gap through the development of an easy to access mobile application designed to democratize technical guidance during exercise, specifically targeting those at an introductory level.

Methods: This study adopted a descriptive and applied research design, following a modified cascade methodology. The system was developed using Kotlin with Android Studio, leveraging the MediaPipe library to track 33 body landmarks. For backend management, the platform integrates Firebase services, including Authentication, Cloud Firestore, and Cloud Storage.

Results: Controlled testing demonstrated that local computer vision processing ensures minimal latency during pose detection, which is important for real time feedback. The interface based on Material Design principles, facilitated a hands-free user experience, while the Text to Speech engine provided auditory corrections without breaking the user's focus during the exercise.

Conclusions: The project validates the feasibility of integrating AI-powered motion tracking into standard mobile devices to serve as virtual supervisors. The implemented Model-View-ViewModel architecture provided a scalable and robust solution that enhances the safety and autonomy of beginner athletes.

C APITULO 1:

MARCO INTRODUCTORIO

CAPITULO 1: MARCO INTRODUCTORIO

1.1 Introducción

En la actualidad la adopción de estilos de vida saludables ha impulsado un crecimiento significativo en la práctica de actividad física autónoma. Este fenómeno potenciado por la disponibilidad de dispositivos móviles genera demanda de herramientas que no solo registre actividad, buscando que actúen como asistentes inteligentes en tiempo real. Sin embargo, uno de los desafíos críticos del entrenamiento autónomo es la ausencia de supervisión técnica, esto incrementa el riesgo de lesiones debido a la mala ejecución de los ejercicios.

El presente proyecto integrador surge como una respuesta técnica a esta problemática, mediante el desarrollo de un prototipo de aplicación móvil diseñado para el seguimiento y corrección de ejercicios de fuerza. La solución propuesta integra algoritmos de visión artificial a través de la biblioteca MediaPipe Pose Landmarker, permitiendo el rastreo de 33 puntos de referencia corporales de manera local en el dispositivo. Esta arquitectura garantiza una retroalimentación inmediata y protege la privacidad del usuario, eliminando la dependencia de procesamiento en la nube para la detección de posturas.

La investigación se aborda bajo un enfoque de investigación aplicada, orientado a la resolución de un problema práctico mediante el desarrollo de una solución tecnológica. El estudio asume un alcance descriptivo, centrándose en la caracterización precisa de las posturas biomecánicas para la validación de ejercicios. Metodológicamente, se emplea el método inductivo para identificar patrones de movimiento y errores recurrentes a través de la observación del usuario fuente, y el método deductivo para establecer las condiciones lógicas y umbrales específicos que permiten al algoritmo determinar la corrección del ejercicio.

Finalmente, el documento se estructura en cinco capítulos: el primero establece el problema y objetivos, el segundo desarrolla el marco teórico, el tercero detalla la metodología y caracterización del usuario, el cuarto describe el diseño e implementación del software, y el quinto presenta la evaluación de resultados y el análisis de impacto frente a las necesidades iniciales identificadas.

1.2 Ubicación y Contextualización del Problema

El problema se ubica en la parte del entrenamiento físico, dirigido a jóvenes principiantes que asisten a gimnasios pequeños o quieren empezar por sí mismos. Estos espacios suelen contar con pocos entrenadores y sin herramientas digitales que les permita tener una supervisión personalizada del ejercicio.

Esta limitación impacta a usuarios sin experiencia previa quienes necesitan orientación constante para evitar lesiones y hacer de manera correcta los ejercicios. La falta de seguimiento adecuado reduce la efectividad del entrenamiento y puede llevar al abandono de actividades física.

Por ello se plantea una propuesta académica para el desarrollo de una aplicación móvil que utilice tecnología de tracking de movimiento con la cámara del celular Android. Esa aplicación busca ofrecer retroalimentación como apoyo técnico en espacios deportivos con escasos recursos o para jóvenes principiantes que comienzan sin asistir a gimnasios, lo que aporta una solución al democratizar el proceso de supervisión técnica.

1.3 Planteamiento del Problema

Los jóvenes usuarios principiantes frecuentemente hacen ejercicios sin supervisión adecuada, lo que puede generar errores en la técnica, lesiones y desmotivación. Esta situación es más fuerte por la falta de herramientas tecnológicas de fácil acceso que ofrezcan asistencia personalizada en tiempo real. Frente a esto, nace la siguiente pregunta de investigación.

¿Cómo una aplicación móvil con tecnología de tracking de movimiento podría asistir a jóvenes principiantes en la correcta ejecución de ejercicios físicos en contextos con recursos limitados?

Los autores de este proyecto integrador plantean una propuesta técnica soluble a esta problemática, enfocándose en el desarrollo conceptual de una herramienta fácilmente disponible, que brinde retroalimentación mediante la cámara del dispositivo móvil.

1.4 Objetivos

1.4.1 Objetivo General

- Desarrollar una aplicación móvil para Android que permita a usuarios principiantes realizar ejercicios físicos con asistencia técnica mediante tecnología de tracking de movimiento usando la cámara del dispositivo

1.4.2 Objetivos Específicos

1. Caracterizar y seleccionar tecnologías de tracking de movimiento adecuadas para el reconocimiento de posturas corporales mediante la cámara del dispositivo móvil.
2. Diseñar una arquitectura móvil basada en el patrón Model-View-ViewModel (MVVM) para que ayude a separar claramente la lógica y la presentación de la aplicación de su interfaz de usuario (UI). Además, de que integre servicios en la nube para la gestión de datos y procesamiento local para el análisis de video.
3. Implementar algoritmos de visión por computadora para la detección de inconsistencias de postura y generar retroalimentación visual y auditiva en tiempo real mediante síntesis de voz.
4. Verificar funcionalmente el prototipo mediante escenarios controlados para evaluar el cumplimiento de los requerimientos técnicos establecidos y la fluidez de la interacción con el usuario.

1.5 Justificación

El desarrollo de una aplicación móvil que utilice tecnología de tracking de movimiento representa una solución innovadora ante la falta de supervisión técnica en el entrenamiento físico de usuarios principiantes. Esta necesidad es especialmente visible en gimnasios pequeños, medianos o en personas que entrenan por su cuenta, donde el acceso a entrenadores personales o tecnologías avanzadas es limitado o inexistente.

Desde el punto de vista social, esta propuesta contribuye a democratizar el acceso a una orientación técnica básica, reduciendo el riesgo de lesiones y aumentando la motivación de los usuarios. Para conseguirlo se aprovecharon tecnologías como la visión por computadora y el tracking de movimiento lo que permite ofrecer una solución práctica y económica. En el ámbito académico, este proyecto fue ideal para aplicar conocimientos de desarrollo móvil lo que dando como resultado un producto con aplicación en deporte y educación.

La propuesta es clave por que mejora la calidad del ejercicio mediante la retroalimentación inmediata. Como funciona desde el celular con la cámara del mismo elimina la necesidad de dispositivos externos lo que hace que la aplicación sea útil y fácil de usar en situaciones reales.

C

APITULO 2:

MARCO TEÓRICO

CAPÍTULO 2: MARCO TEÓRICO

2.1. Antecedentes de investigaciones relacionadas

En junio de 2023, el desarrollador Yosuke Hanaoka publicó en la plataforma Dev.to el artículo titulado “Fitness App Development with Real-Time Posture Detection using MediaPipe and React”, donde detalla el desarrollo de una aplicación web que permite detectar posturas humanas en tiempo real con el objetivo de mejorar la ejecución de ejercicios físicos. Esta herramienta, denominada BodyBuddy, hace uso de la biblioteca MediaPipe para identificar los puntos clave del cuerpo mediante la cámara del dispositivo del usuario, permitiendo analizar la postura durante actividades como sentadillas o flexiones.

La investigación técnica y desarrollo práctico descritos en este artículo sirvieron como base para comprender el funcionamiento de la visión por computadora aplicada al monitoreo postural en tiempo real. Esto resultó útil como referencia para el diseño del módulo de detección y corrección de postura en la presente investigación ya que demuestra cómo integrar tecnologías modernas como MediaPipe y procesamiento local para ofrecer asistencia automatizada sin necesidad de dispositivos adicionales. Este antecedente fortalece el enfoque de aprovechar el potencial de los dispositivos móviles y las bibliotecas de inteligencia artificial para el seguimiento físico de los usuarios en aplicaciones orientadas al bienestar (Hanaoka, 2023).

El artículo titulado "Build an AI Fitness Trainer Using MediaPipe For Squat Analysis", publicado en LearnOpenCV por (Dawn, 2022), habla acerca del desarrollo de una aplicación de ejercicio que utiliza inteligencia artificial para analizar y mejorar la ejecución de sentadillas. La aplicación usa Google MediaPipe, una solución de tracking de movimiento que estima 33 puntos clave del cuerpo humano en tiempo real a partir de imágenes de video RGB.

El sistema ha sido desarrollado para brindar dos métodos de entrenamiento: "Principiante" y "Profesional", de esta manera ajustándose al nivel de experiencia del usuario. La aplicación determina la ejecución adecuada de las sentadillas mediante la medición de ángulos concretos entre articulaciones, como la cadera, el tobillo y la rodilla, en función de su posición vertical. Esta perspectiva muestra que es posible mejorar la calidad del entrenamiento físico a través de software al combinar métodos de visión por computadora y aprendizaje automático (Dawn, 2022).

El artículo titulado "Muscle Vision: Real Time Key point Based Pose Classification of Physical Exercises", presentado por Alex Moran y colaboradores en marzo de 2022, habla sobre un sistema de clasificación de ejercicios en tiempo real usando puntos de referencia del cuerpo humano. El sistema emplea la solución BlazePose de MediaPipe para extraer 33 puntos clave del cuerpo a partir de video en vivo. Estos puntos se procesan mediante una red neuronal Long Short-Term Memory para clasificar ejercicios como flexiones, estocadas, sentadillas y lanzamientos de disco. El modelo muestra los resultados en una página web lo que facilita el análisis de los resultados obtenidos.

Lo valioso de este proyecto es su diseño liviano hecho para que funcione en celulares comunes sin necesidad de dispositivos externos. Al enfocarse en los puntos de referencia en vez de analizar la imagen completa hace que el sistema consuma menos recursos y reduzca la carga del procesamiento. En consecuencia, se proporciona una base para el desarrollo de aplicaciones de entrenamiento físico asistidas por inteligencia artificial, que clasifica y diferencia ejercicios en tiempo real (Alex Moran, 2022).

2.2. Definiciones conceptuales

2.2.1. Actividad Física

La OMS define la actividad física como todo movimiento corporal producido por los músculos esqueléticos que requiere consumir energía. En la práctica, consiste en cualquier movimiento, realizado incluso durante el tiempo de ocio, que se efectúa para desplazarse a determinados lugares y desde ellos, para trabajar o para llevar a cabo las actividades domésticas (Organización Mundial de la Salud, 2024).

La práctica de actividad física influye en la mejora de la condición física relacionada con la salud y en mayor medida lo hace el ejercicio físico (Escalante, 2011).

2.2.2. Ejercicio Físico

Se define como la actividad física planificada, estructurada y repetida, cuyo objetivo es adquirir, mantener o mejorar la condición física. En tanto que, un programa de ejercicio físico

requiere la planificación y estructuración de la intensidad, volumen y tipo de actividad física que se desarrolla (Escalante, 2011).

2.2.3. Visión por Computadora (Computer Vision)

Según (IBM, 2021), es un campo de la inteligencia artificial que utiliza el aprendizaje automático y las redes neuronales para enseñar a los ordenadores y sistemas a obtener información significativa de imágenes digitales, vídeos y otros elementos visuales y a hacer recomendaciones o tomar medidas cuando ven defectos o problemas.

La visión por computadora se realiza mediante el uso de cámaras, datos y algoritmos en lugar de retinas, nervios ópticos y una corteza visual. Para lograrlo se utilizan dos tecnologías esenciales: un tipo de aprendizaje automático llamado aprendizaje profundo y una red neuronal convolucional.

El aprendizaje automático utiliza modelos algorítmicos que permiten a una computadora aprender sobre el contexto de los datos visuales. Si se introducen suficientes datos a través del modelo, la computadora "mirará" los datos y se enseñará a sí misma a distinguir una imagen de otra. Los algoritmos permiten que la máquina aprenda por sí misma, en lugar de que alguien la programe para que reconozca una imagen.

Una CNN ayuda a un modelo de aprendizaje automático o aprendizaje profundo a "mirar" dividiendo las imágenes en píxeles a los que se les asignan etiquetas o rótulos. Utiliza las etiquetas para realizar convoluciones, que es una operación matemática en dos funciones para producir una tercera función, y, hace predicciones sobre lo que está "viendo". La red neuronal ejecuta convoluciones y comprueba la precisión de sus predicciones en una serie de iteraciones hasta que las predicciones comienzan a hacerse realidad. Entonces se trata de reconocer o ver imágenes de una manera similar a los humanos (IBM, 2021).

Esta tecnología se implementa para que el dispositivo pueda interpretar las imágenes de la cámara y pueda decidir si el ejercicio ha sido realizado correctamente.

2.2.4. Estimación de Pose Humana, Pose estimación

Es una tarea de la visión por computadora que consiste en detectar y estimar las posiciones de varias partes del cuerpo en una imagen o vídeo de una persona. Esto se puede utilizar para comprender la postura, el movimiento y el comportamiento de la persona, así como para habilitar aplicaciones como la interacción humano-computadora, la videovigilancia y el análisis deportivo.

La estimación de la postura humana puede ser una tarea difícil, especialmente en situaciones en las que la persona está ocluida o las condiciones de iluminación son deficientes. Existen varios métodos para realizarla, entre ellos el uso de algoritmos de aprendizaje automático para aprender sobre la anatomía humana. Esto consiste en entrenar a un modelo con un conjunto de datos de poses y referencias previamente etiquetados a mano. Cuando se finaliza el entrenamiento el sistema será capaz de predecir posiciones de articulaciones en datos que no estaban en el conjunto de entrenamiento.

Otro enfoque para la estimación de la pose humana implica el uso de modelos geométricos para representar el cuerpo y sus partes y el ajuste de estos modelos a las imágenes o videos. Esto puede implicar estimar las posiciones de los puntos clave, como las articulaciones y usarlos para inferir las posiciones de otras partes del cuerpo (Encord, 2025).

2.2.5 Seguimiento de Movimiento, Motion Tracking

Se refiere al proceso de identificar y seguir objetos a lo largo del tiempo en secuencias de vídeo. Desempeña un papel fundamental en numerosas aplicaciones, que van desde la vigilancia y el control del tráfico hasta la realidad aumentada y el análisis deportivo.

Seguimiento de objetos de imagen

El seguimiento de objetos de imagen, a menudo denominado seguimiento de un sólo fotograma implica la identificación y el seguimiento de objetos dentro de una sola imagen fija.

Seguimiento de objetos de vídeo

El seguimiento de objetos de vídeo, por otro lado, amplía el concepto de seguimiento a través de varios fotogramas en una secuencia de vídeo. Esta forma dinámica de seguimiento se

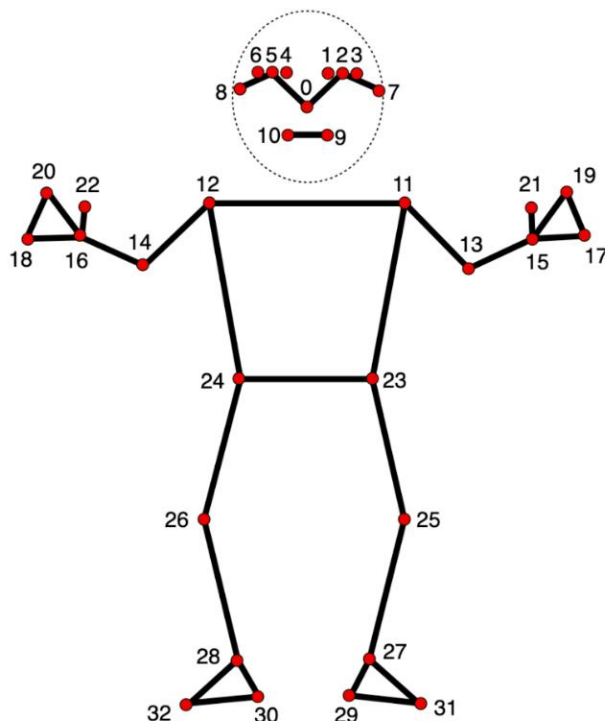
ocupa de detectar y seguir objetos a medida que se mueven y cambian con el tiempo (GeeksforGeeks, 2024).

2.2.6. MediaPipe

MediaPipe te permite detectar puntos de referencia de cuerpos humanos en una imagen o un video. Puedes usar esta tarea para identificar ubicaciones clave del cuerpo, analizar la postura y categorizar los movimientos. Esta tarea usa modelos de aprendizaje automático que trabajan con imágenes o videos individuales. La tarea genera puntos de referencia de la postura del cuerpo en coordenadas de imagen y en coordenadas mundiales tridimensionales (Google, 2025). Esto constituye el motor de inteligencia artificial del proyecto, encargado de detectar los 33 puntos de referencia del cuerpo necesarios para el análisis de la pose.

Ilustración 1

Modelo de Marcador de Posición



Nota: El modelo de marcador de pose realiza un seguimiento de 33 ubicaciones de puntos de referencia del cuerpo, que representan la ubicación aproximada de partes externas e internas de extremidades como rodillas, piernas, pies, hombros, codos, ojos, boca, orejas, manos y pulgares.

2.2.7 Aplicaciones de Fitness

Se refiere a una aplicación móvil diseñada para promover la actividad física y la salud en general al proporcionar herramientas para planes de entrenamiento, seguimiento de la actividad y análisis del progreso del fitness.

Las aplicaciones de fitness ayudan a los usuarios a lograr estilos de vida más saludables al abordar los comportamientos sedentarios y fomentar rutinas de ejercicio constantes (Sarvaiya, 2024).

2.2.8 Kotlin

Kotlin es un lenguaje de programación de código abierto creado por JetBrains que se ha popularizado gracias a que se puede utilizar para programar aplicaciones Android.

Este lenguaje es de tipado estático, ya que se puede desarrollar sobre JVM o JavaScript; o desde hace unos meses, incluso sin necesidad de ninguna de ellas, ya que paralelamente se está desarrollando en nativo con LLVM.

Kotlin destaca por las ventajas que tiene respecto a Java, las cuales se documentan en las siguientes secciones, referidas al desarrollo de la aplicación móvil, además de por presentar características como simplificar la lectura del código y el propio desarrollo de este.

Estas son algunas de las ventajas y características de Kotlin.

Interoperabilidad con código Java

Una de las características principales de Kotlin es que está diseñado para interoperar completamente con la sintaxis del lenguaje de Java. Es decir, con una base de código existente escrita en Java, puede interactuar correctamente con Kotlin y viceversa.

Curva de aprendizaje sencilla

La sencillez de la sintaxis permite una curva de aprendizaje fluida, intuitiva y fácil de usar, perfecta para los que quieran aprender su primer lenguaje de programación. Además, como es de código abierto, hay un gran apoyo de la comunidad de Kotlin, lo que supone una gran ventaja.

Menor tiempo de programación

Uno de los puntos fuertes de Kotlin es que elimina el código redundante, además de ser compacto y conciso, lo que optimiza mucho el proceso de escritura de código y evita la repetición.

Orientado a objetos y programación funcional

Aunque lo habitual en el desarrollo de aplicaciones móviles es un paradigma orientado a objetivos, Kotlin demuestra que también se puede trabajar de la mano de la programación funcional. La posibilidad de trabajar con lambdas en este entorno simplifica las tareas más comunes y tediosas en el desarrollo.

Corrutinas

Otra de las grandes ventajas de Kotlin es que las Corrutinas optimizan la programación asíncrona. Simplifican así el aburrido trabajo de las llamadas de red y acceso a las bases de datos y dejan atrás los callbacks.

Desarrollo multiplataforma

Kotlin se puede utilizar para cualquier tipo de desarrollo, desde la web del lado del servidor y del lado del cliente, hasta Android y iOS. Como el lenguaje se ejecuta en JVM, permite compartir código entre diferentes plataformas.

Flexibilidad

Kotlin da a los desarrolladores libertad de trabajar con el estilo que elijan. Por tanto, es un lenguaje altamente flexible que tiene construcciones funcionales y orientadas a objetos. Todo ello se traduce en una mejor experiencia a la hora de programar (Conorea, 2022).

Se seleccionó este lenguaje para el desarrollo nativo en Android debido a su eficiencia en la gestión de procesos asíncronos requeridos por el tracking.

2.2.9 Android Studio

Android Studio es el entorno de desarrollo integrado (IDE) oficial que se usa en el desarrollo de aplicaciones para Android. Basado en el potente editor de código y las

herramientas para desarrolladores de IntelliJ IDEA, Android Studio ofrece aún más funciones que mejoran tu productividad cuando compilas aplicaciones para Android, como las siguientes:

- Un sistema de compilación flexible basado en Gradle
- Un emulador rápido y cargado de funciones
- Un entorno unificado donde puedes desarrollar para todos los dispositivos Android
- Ediciones en vivo para actualizar elementos componibles en emuladores y dispositivos físicos, en tiempo real
- Integración con GitHub y plantillas de código para ayudarte a compilar funciones de aplicaciones comunes y también importar código de muestra
- Variedad de marcos de trabajo y herramientas de prueba
- Herramientas de Lint para identificar problemas de rendimiento, usabilidad y compatibilidad de versiones, entre otros
- Compatibilidad con C++ y NDK
- Compatibilidad integrada con Google Cloud Platform, que facilita la integración con Google Cloud Messaging y App Engine (Android Developer, 2025).

2.2.10 Jetpack Navigation

El componente Navigation controla diversos casos de uso de navegación, desde clics simples en botones hasta patrones más complejos, como las barras de la aplicación y el panel lateral de navegación (Google Developers, 2025).

Facilita la implementación de un flujo de usuario coherente y fluido, permitiendo una transición lógica entre el catálogo de ejercicios, la cámara de tracking y el historial de progreso.

2.2.11 Text to Speech (TTS)

Es una tecnología de síntesis de voz que permite al sistema convertir texto escrito en audio hablado de manera natural mediante el motor de voz del dispositivo (Android Developers, 2025).

Constituye el motor del "Entrenador de Voz", permitiendo que el usuario reciba correcciones de postura sin necesidad de mirar la aplicación, lo que mejora la seguridad y la experiencia de usuario.

2.2.12 Firebase

Firebase es una plataforma en la nube destinada al desarrollo de aplicaciones web y móviles que ayuda a simplificar el desarrollo de estas. Para ello proporciona un conjunto de herramientas destinadas a 4 objetivos principales: desarrollo, crecimiento, monetización y análisis de datos.

Gracias a esta plataforma, los desarrolladores pueden dedicarle menos tiempo al backend de las aplicaciones, centrándose así en la experiencia del usuario y en la innovación de características, lo que permite acelerar el ciclo de desarrollo y ofrecer productos de alta calidad de manera más rápida y eficiente. Todo esto difiere de la programación de software tradicional, donde normalmente implicaba desarrollar tanto la parte frontend como el backend (Merino, 2024).

Firebase actúa como la plataforma backend que centraliza la lógica de datos, permitiendo un desarrollo ágil sin necesidad de gestionar servidores físicos.

2.2.13 Firebase Authentication

La autenticación es un proceso fundamental en prácticamente toda aplicación, sin embargo, su implementación requiere de un almacenamiento seguro de los datos haciendo uso de técnicas de cifrado para las contraseñas, así como otras medidas de protección contra ataques por inyección SQL o de fuerza bruta. Además de todo esto, es necesario llevar a cabo una gestión de la sesión del usuario mediante el uso de tokens, lo que complica aún más la lógica.

Es por ello que Firebase presenta Authentication, un producto destinado a facilitar la creación de sistemas de autenticación seguros que admite cuentas de correo electrónico y contraseña, autenticación telefónica, autenticación anónima e integración con proveedores de identidad federada, tales como Google, Facebook, Apple, X o GitHub (Merino, 2024).

2.2.14 Cloud Firestore

Cloud Firestore es una base de datos flexible y escalable para el desarrollo en servidores, dispositivos móviles y la Web desde Firebase y Google Cloud. Al igual que Firebase Realtime Database, mantiene tus datos sincronizados entre aplicaciones cliente a través de objetos de escucha en tiempo real y ofrece soporte sin conexión para dispositivos móviles y la Web, por lo que puedes compilar aplicaciones con capacidad de respuesta que funcionan sin importar la latencia de la red ni la conectividad a Internet.

Cloud Firestore, almacenas los datos en documentos que contienen campos que se asignan a valores. Estos documentos se almacenan en colecciones, que son contenedores para tus documentos y que puedes usar para organizar tus datos y compilar consultas. Los documentos admiten varios tipos de datos diferentes, desde strings y números simples, hasta objetos anidados complejos. También puedes crear subcolecciones dentro de documentos y crear estructuras de datos jerárquicas que se ajustan a escala a medida que tu base de datos crece (Firebase, 2025).

Base de datos NoSQL empleada para almacenar de forma jerárquica y en tiempo real el historial de desempeño y rutinas de los usuarios.

2.2.15 Cloud Storage for Firebase

Cloud Storage for Firebase es un servicio de almacenamiento de objetos potente, simple y rentable construido para el escalamiento de Google. Los SDK de Firebase para Cloud Storage agregan la seguridad de Google a las operaciones de carga y descarga de archivos de tus aplicaciones de Firebase, sin importar la calidad de la red (Firebase, 2026).

Actúa como repositorio central para todos los archivos multimedia pesados de la aplicación, separando la carga de archivos grandes de la aplicación, permitiendo una gestión de contenido escalable y de alta disponibilidad

2.2.16 Glide

Es una biblioteca de código abierto para Android especializada en la gestión, carga y almacenamiento en caché de imágenes y contenido multimedia de forma eficiente (BumpTech, 2024).

Permite la carga dinámica de guías visuales y GIFs demostrativos desde la nube sin comprometer el rendimiento del procesamiento de video en tiempo real de MediaPipe.

2.2.17 AndroidChart

Es una biblioteca de visualización de datos diseñada específicamente para Android. Permite la generación de gráficos interactivos y dinámicos soportando una amplia variedad de representaciones como diagramas de barras, líneas, sectores y radares (AppDevNext, 2025). Usado para los gráficos de representación de progreso semanal.

2.2.18 Material Calendar Design

Es un widget simple y personalizable para Android basado en Material Design. Tiene 2 funcionalidades, un selector de fechas y un calendario clásico, el selector de fechas funciona como selector de fecha única o selector de rango de fechas (Applandeo, 2023). Permite a los usuarios visualizar su historial de entrenamientos en una interfaz de calendario, facilitando el seguimiento de la consistencia en sus rutinas.

2.2.19 Ucrop

Es una librería para recorte de imágenes hecha para Android, permite controlar parámetros como la relación de aspecto, rotación y calidad de archivo resultante (Yalantis, 2025). Usada en módulo de perfil al usuario ajustar y recortar su foto de perfil.

2.3. Conclusiones relacionadas al marco teórico

En el presente capítulo se concluyó que el desarrollo de aplicaciones móviles orientadas al ejercicio físico ha evolucionado considerablemente gracias a la integración de tecnologías de visión por computadora, inteligencia artificial y bibliotecas como MediaPipe. Estas herramientas

permiten identificar y analizar la postura humana en tiempo real utilizando únicamente la cámara del dispositivo móvil, lo que reduce la necesidad de sensores externos o equipos especializados.

La actividad y el ejercicio físicos son conceptos claves que sustentan el propósito de la presente investigación, ya que su práctica adecuada contribuye directamente al bienestar de los usuarios, especialmente en jóvenes principiantes que buscan mejorar su condición física. Las aplicaciones de fitness se presentan como una alternativa efectiva para motivar y guiar estos procesos lo que permite el seguimiento personalizado del rendimiento del usuario.

El uso de herramientas como Firebase y Android Studio facilita el desarrollo de este tipo de soluciones al ofrecer servicios integrados como autenticación, almacenamiento y análisis de datos en tiempo real.

Firebase y Android estudio son una combinación ideal para el proyecto, Android Studio por ser el entorno de desarrollo de Android por defecto y Firebase por sus servicios integrados de autenticación y almacenamiento. Lo que permite crear una aplicación robusta para que asista al usuario durante sus sesiones de ejercicios y también para el almacenamiento y gestión de los datos de manera segura.

Los antecedentes mencionados confirman que es posible usar soluciones de tracking en tiempo real para la corrección de ejercicios lo que respalda la viabilidad del proyecto integrador para fomentar hábitos de vida más saludables y activos.

C APITULO 3:

MARCO INVESTIGATIVO

CAPÍTULO 3: MARCO INVESTIGATIVO

3.1 Tipos de Investigación

3.1.1 Investigación Aplicada

De acuerdo con (Stewart, 2025) la investigación aplicada se centra en el análisis destinado a resolver problemas prácticos. Su principal objetivo no es simplemente ampliar la base de conocimientos existentes, sino aprovechar esos conocimientos para desarrollar soluciones, innovaciones o intervenciones que puedan aplicarse directamente en el mundo real.

El enfoque de investigación aplicada se usa porque el objetivo principal es el desarrollar una aplicación móvil capaz de realizar seguimiento y corrección de ejercicios en tiempo real mediante el uso de la cámara y el conjunto de bibliotecas de software MediaPipe, este conjunto de bibliotecas proporciona herramientas técnicas de inteligencia artificial y aprendizaje automático rápidamente implementables en diversas aplicaciones, entre ellas las del sector del bienestar y la salud como la presentada en este proyecto integrador. También para la ejecución de la propuesta se asume un alcance descriptivo debido a la necesidad de analizar y registrar las posturas que definen la correcta ejecución de los ejercicios.

3.2 Métodos de investigación

3.2.1 Método deductivo

De acuerdo con (Univesidad UNADE, 2025) el método deductivo es una estrategia de razonamiento ampliamente utilizada en diversos campos del conocimiento, especialmente en la educación, la filosofía y las ciencias formales. Este método parte de principios generales para llegar a conclusiones específicas, siguiendo una lógica estructurada y coherente.

El método deductivo se utiliza en la etapa de diseño e implementación del sistema, a partir de los criterios de postura definidos en la fase inductiva, se deducen condiciones específicas que permiten al algoritmo determinar cuándo un ejercicio es ejecutado de forma correcta o incorrecta.

3.2.2 Método Inductivo

El investigador (Ameneyro-Ameneyro, 2024) explica que el método inductivo parte de la observación de casos particulares que se recopilan de manera sistemática y, a través del reconocimiento de patrones de regularidad, se explican las interdependencias entre clases de eventos fácticos, con las que se formula la regla general que cumplen y se construyen marcos teóricos.

El método inductivo se aplica en esta investigación durante la fase de análisis inicial de los ejercicios. A través de la observación de múltiples ejecuciones, se identifican patrones de movimiento, poses características y errores recurrentes cometidos por los usuarios.

3.3 Herramientas de recolección de datos

La entrevista es un método de recolección de datos que consiste en preguntar a una o varias personas su opinión sobre una empresa, producto o tema, tienen un carácter cualitativo por lo que se centran en la experiencia personal, cuyo objetivo principal es conocer comportamientos, actitudes y opiniones de las personas (Muguira, 2023).

La entrevista se planteó a una persona que realiza ejercicios y deporte de manera regular, a la cual se le realizaron una serie de preguntas para identificar puntos críticos en la ejecución técnica y facilitar el levantamiento de requisitos funcionales del sistema.

3.4 Fuentes de Información

3.4.1 Fuentes Primarias

Las fuentes primarias fueron conformadas por los usuarios que participaron en la ejecución de los ejercicios evaluados mediante la aplicación móvil, quienes proporcionaron información útil a través de los movimientos captados por la cámara. También se consideran las mediciones generadas por la aplicación, como los ángulos detectados, la secuencia de poses y los resultados de las pruebas del prototipo, ya que representan información original producida durante el desarrollo y validación de la aplicación.

3.4.2 Fuentes secundarias

Se consideran fuentes secundarias los libros, artículos científicos, páginas web, documentos académicos, tesis, publicaciones especializadas y material digital consultados para fundamentar teóricamente la investigación.

3.5 Instrumental Operacional

3.5.1 Estructura y características de los instrumentos de recolección de datos

Para entrevistar a la persona que hace ejercicio regular, se elaboró una guía de entrevista semiestructurada, elaborada para alinearse con los objetivos del desarrollo del proyecto y los requerimientos planteado en capítulos posteriores.

Las preguntas fueron formuladas con un lenguaje sencillo de comprender, para favorecer la rápida identificación de puntos críticos en la ejecución de ejercicios mediante el uso de MediaPipe, tecnología de gran repercusión en el campo de la visión por computadora. Además, del empleo conveniente de Kotlin como lenguaje de programación orientado a objetos multiplataforma, estáticamente tipado, de alto nivel y propósito general con inferencia de tipos.

3.6 Caracterización del Usuario Objetivo

Para el levantamiento de requerimientos funcionales se contó con la participación de una persona clave. Esto responde a la naturaleza de la investigación aplicada, donde se busca profundizar las necesidades de un perfil en específico antes de la escalabilidad del sistema.

- **Experiencia:** Nivel medio de entrenamiento de fuerza con conocimiento de técnicas de posturas básica.
- **Rango de Edad:** Entre 18 y 30 años
- **Perfil tecnológico:** Usuario habitual de dispositivos móviles como apoyo para registro de rutinas
- **Interés:** Optimización de la técnica mediante herramientas digitales que garanticen privacidad y fluidez

3.7 Estrategia Operacional para la recolección y tabulación de datos

Como se indicó, esta investigación es de tipo mixto, aplicada con detalles descriptivos. Como fuente de información, además de la bibliográfica se realizó una entrevista a una persona que practica ejercicio con regularidad, complementada con la técnica de observación, con el fin de conocer, a partir de la experiencia de la persona entrevistada, acerca de los errores y problemas que se suele tener al momento de realizar los ejercicios.

Se define como perfil de usuario objetivo a los deportistas y entusiastas de la actividad física a aquellos que realizan entrenamientos y ejercicio de forma autónoma. Este perfil se caracteriza por el uso de dispositivos móviles como herramientas de apoyo en su rutina diaria y la búsqueda de precisión en la ejecución de ejercicios de fuerza. La información recolectada del usuario fuente permitió validar que este perfil requiere soluciones tecnológicas que no interrumpan el flujo de su entrenamiento.

Con esto se procuró asegurar que la lógica de seguimiento de poses sea validada en un entorno de uso real y dado que los datos de la entrevista son cualitativos, se organizaron los resultados en una matriz de hallazgos y soluciones (ver **Tabla 1**), lo que permitió vincular las declaraciones del entrevistado con un requerimiento técnico específico del sistema.

Tabla 1

Matriz de Hallazgos y Soluciones de la entrevista realizada

Nº	Dimensión / Pregunta	Hallazgo (Respuesta del Entrevistado)	Solución Técnica Implementada (App)
1	Frecuencia y rutinas	Entrena 4 veces por semana con rutinas de fuerza y calistenia.	Módulo de gestión de ejercicios y rutinas personalizadas en Kotlin.
2	Desafío al entrenar sólo	Identifica la falta de supervisión técnica como su mayor obstáculo.	Integración de MediaPipe Pose para supervisión postural constante.
3	Conteo de repeticiones	Reporta pérdida de la cuenta mental debido a la fatiga física.	Lógica de Contador Automático mediante estados (Up/Down) detectados por la cámara.
4	Verificación de postura	El uso de espejos resulta ineficiente y poco práctico durante el ejercicio.	Visualización de <i>landmarks</i> en tiempo real mediante PoseOverlayView.
5	Dudas articulares	Incertidumbre sobre la profundidad de la sentadilla y la alineación lumbar.	Configuración de umbrales angulares específicos en archivos JSON.

6	Ángulos	Valida la importancia de tener una medición para evitar lesiones.	Algoritmos trigonométricos para el cálculo de ángulos y poses.
7	Tipo de alerta	Prefiere alertas auditivas para no distraerse durante el ejercicio.	Implementación Text to Speech para correcciones de voz.
8	Elementos visuales	Requiere colores fuertes y números grandes para visibilidad a distancia.	Interfaz diseñada con Material Design y feedback con colores. (Verde/Rojo).
9	Aceptación de la IA	Total disposición a utilizar el monitoreo por cámara para su entrenamiento.	Procesamiento local para garantizar baja latencia en el tracking.

Nota. Matriz que resalta lo obtenido de la entrevista con la persona clave, las preguntas completas se encuentran en el **Anexo A**. Elaboración Propia.

La interpretación presentada en la tabla anterior se llevó a cabo a través de un análisis explicativo-descriptivo, comenzando con los resultados de la matriz y finalizando con la validación de las funcionalidades del software. Este procedimiento posibilitó establecer la adecuación de las alertas visuales y auditivas, garantizando que la propuesta tecnológica funcionara y que sea bien acogida por el usuario final.

A continuación, se presenta el análisis narrativo por dimensiones:

- **Dimensión 1: Hábitos y Estructura (Ítems 1-3):** El usuario rutina de ejercicio constante, lo que justifica la estructura de la base de datos en Firebase Firestore para organizar su progreso. Automatizar el conteo de repeticiones resuelve el problema de perder el conteo por la fatiga física permitiendo al usuario concentrarse en el ejercicio.
- **Dimensión 2: Postura y Precisión (Ítems 4-6):** El usuario no puede identificar los errores técnicos por sí mismo, lo que hace necesario el uso de visión por computadora. La aplicación utiliza los 33 puntos de referencia de MediaPipe mostrados en la . No se encuentra el origen de la referencia. para remplazar la función de un espejo, lo que proporciona una validación, que el usuario por sí mismo no podría detectar
- **Dimensión 3: Interfaz y Retroalimentación (Ítems 7-9):** La preferencia por lo auditivo se traduce en un requerimiento funcional crítico. El sistema utiliza el motor de voz de Android para emitir correcciones inmediatas, asegurando que la interacción sea fluida y

no intrusiva. La usabilidad se optimizó mediante una interfaz de alta visibilidad, facilitando el seguimiento del ejercicio a más de 2 metros de distancia.

C APITULO 4:

MARCO PROPOSITIVO

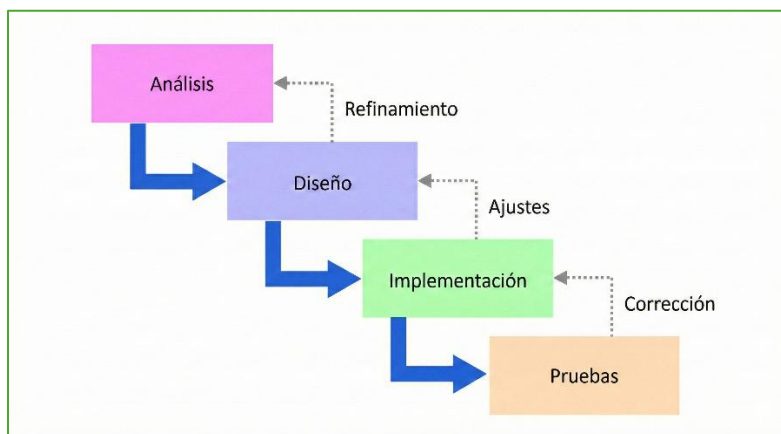
CAPÍTULO 4: MARCO PROPOSITIVO

4.1 Introducción

La metodología en cascada es un enfoque lineal y secuencial, para la gestión de proyectos. Sus fases se ejecutan en orden y cada etapa se completa antes de pasar a la siguiente. Para fines de este proyecto integrador se adaptó y adoptó en enfoque de la metodología de Cascada, empleada comúnmente en el desarrollo de software (Singh, 2022). A este enfoque se le refiere como Cascada Modificada, este mantiene la característica estructura secuencial, con el detalle de que se permite el retroceso a fases previas para la corrección de errores, pero no es una metodología ágil tradicional.

Ilustración 2

Estructura de Cascada Modificada



Nota. Debido a la naturaleza de la propuesta se excluye la fase de mantenimiento tradicional ya que es un prototipo y las correcciones se hacen en implementación

Se eligió Cascada modificada para esta aplicación porque se adapta a los requisitos que se han definido. También permite la incorporación de iteraciones en fases importantes como el diseño y evaluación de la aplicación. Por otro lado, la metodología se adapta adecuada para la elaboración del prototipo de la aplicación, ya que se requiere una documentación detallada en cada fase del proceso.

4.2 Descripción de la propuesta

La propuesta consiste en el desarrollo de una aplicación móvil de trackeo de ejercicios que utiliza tecnologías de visión por computadora para el seguimiento y validación de ejercicios físicos en tiempo real.

El objetivo es proporcionar a los usuarios una herramienta que les permita realizar rutinas de ejercicios de manera segura y correcta, sin necesidad de máquinas o implementos adicionales. Para esto, la aplicación integra modelos de detección de poses, configuraciones específicas de cada ejercicio y un sistema de retroalimentación inmediata.

4.3 Determinación de Recursos

3.3.1 Recursos Humanos

Tabla 2

Recursos Humanos necesarios

Recursos Humanos	Función
Ing. Pincay Ponce Jorge Iván	Director del Proyecto Integrador
Ortega Giler Miguel Ángel	Autor y Desarrollador del Proyecto Integrador
Baque Barcia Josthin Ariel	Autor y Desarrollador del Proyecto Integrador

Nota. Elaboración Propia.

3.3.2 Recursos Tecnológicos

Tabla 3

Recursos Tecnológicos necesarios

Recursos tecnológicos	Función
Computadora portátil	Utilizada para el desarrollo de la lógica de programación, diseño de interfaces y documentación del proyecto.
Celular (Android)	Usado para realizar las pruebas con la cámara y validar el tracking del ejercicio en tiempo real.

Recursos tecnológicos	Función
Android Studio	Entorno de desarrollo utilizado para el desarrollo de aplicaciones nativas de Android.
Lenguaje Kotlin	Lenguaje de programación oficial de Android.
Google MediaPipe	Framework de visión artificial utilizado para la detección de los 33 landmarks corporales y el análisis postural.
Firebase Firestore	Base de datos NoSQL en la nube utilizada para el almacenamiento y sincronización del historial de ejercicios y perfiles de usuario.
Firebase Storage	Servicio en la nube utilizado para el alojamiento y servicio de recursos multimedia pesados como GIFs y videos demostrativos.
Text to Speech (TTS)	Motor de síntesis de voz utilizado para proporcionar retroalimentación auditiva y correcciones técnicas al usuario en tiempo real.
Material Design	Framework de diseño utilizado para garantizar una interfaz gráfica amigable, accesible y moderna para usuarios principiantes.
Jetpack Navigation	Biblioteca utilizada para gestionar de forma eficiente el flujo y la navegación entre los diferentes módulos de la aplicación.
Kotlin Coroutines	Herramienta utilizada para la gestión de tareas asíncronas, evitando el bloqueo de la interfaz mientras se procesa el video de la cámara.
Librería Glide	Utilizada para la carga eficiente y el manejo de caché de las imágenes y guías visuales almacenadas en la nube.
AndroidChart	Librería de gráficos para aplicaciones de Android usada para el grafico de progreso de ejercicios
Material Calendar Design	Librería de calendarios para aplicaciones de Android usada para que el usuario haga seguimiento de los días que hizo ejercicio con la aplicación.
Ucrop	Librería para recorte de imágenes utilizada para el redimensionamiento de imágenes para la foto de perfil del usuario

Nota. Describes todos los recursos tecnológicos usados en el desarrollo de la aplicación.

Elaboración Propia.

3.3.3 Recursos Económicos

Tabla

4

Recursos Económicos necesarios

Categoría	Recurso	Descripción	Total Estimado
Humanos	Desarrolladores (2)	Diseño, lógica de IA y arquitectura (240h c/u a \$5/h)	\$ 2,400.00
Tecnológicos	Laptops (2)	Equipos de gama media para desarrollo en Android Studio	\$ 800.00
Tecnológicos	Smartphone Android	Dispositivo para pruebas de tracking (MediaPipe/CameraX)	\$ 200.00
Servicios	Firebase Blaze Plan	Siguiente nivel (Pay-as-you-go) para escalabilidad	\$ 25.00*
Servicios	Internet	Conectividad mensual para sincronización de datos	\$ 180.00
TOTAL			\$ 3,605.00

Nota. Los costos son aproximados y no representan un precio preciso. Elaboración Propia.

4.4 Etapas de acción para el desarrollo de la propuesta

4.4.1 Análisis

En esta fase, los ejercicios se seleccionaron bajo 1 criterio específico, ejercicios con peso corporal, para evitar posibles problemas de la tecnología del tracking con máquinas robustas o con muchas partes en movimiento. Así se definió un conjunto de 12 ejercicios. Para los ejercicios se definirán parámetros biométricos necesarios, tales como ángulos ideales de las articulaciones, las fases del movimiento del ejercicio y puntos críticos, es decir, poses iniciales, intermedias y finales, a fin de monitorearlos durante su ejecución.

También se tendrá que identificar errores frecuentes que ocurren en cada ejercicio, con el objetivo de que el programa pueda sugerir correcciones y proporcionar retroalimentación correctiva en tiempo real. Finalmente se revisó las tecnologías disponibles para el seguimiento de poses en tiempo real, se determinó que MediaPipe era la mejor, ya que fue desarrollada con

aplicaciones nativas en mente, en el caso del proyecto, para Android lo cual le da una compatibilidad superior a la de otras tecnologías ya que el desarrollo de MediaPipe está orientado a Android.

Los requisitos del sistema describen las funciones y condiciones que la aplicación móvil debe cumplir para garantizar su correcto funcionamiento. Se dividen en tres categorías: requisitos de información, requisitos funcionales y requisitos no funcionales.

4.4.1.1 Requisitos de información

- **IRQ_01:** Registro y autenticación de usuarios
- **IRQ_02:** Visualización de ejercicios y rutinas
- **IRQ_03:** Detección y corrección de posturas corporales

4.4.1.2 Requisitos Funcionales

- **RF1 Registro e Inicio de sesión:** La aplicación deberá permitir el registro e inicio de sesión tras la validación de las credenciales con Firebase Auth.
- **RF2 Consulta de Ejercicios:** La aplicación deberá permitir consultar la lista de ejercicios disponibles según nivel verificando que los ejercicios que se muestran coincidan con los almacenados en Cloud Firestore.
- **RF3 Gestión de Rutinas:** La aplicación deberá permitir al usuario la creación, edición y eliminación de sus propias rutinas personalizadas para organizar su plan de entrenamiento.
- **RF4 Activación de cámara:** La aplicación deberá activar la cámara del dispositivo para el seguimiento de ejercicios en tiempos menores a 3 segundos para garantizar la fluidez de uso.
- **RF5 Detección de pose:** El sistema deberá detectar la pose corporal en tiempo real utilizando MediaPipe de forma constante y que el esqueleto virtual se muestre de manera constante.
- **RF6 Retroalimentación:** La aplicación deberá mostrar retroalimentación visual cuando el usuario adopte una postura incorrecta, estas se mostrarán de manera visual o auditiva, dependiendo de los parámetros definidos en los archivos JSON.

4.4.1.3 Requisitos No Funcionales

- **RNF1 Latencia:** El sistema debe procesar los fotogramas de video de forma local con latencia inferior a 200 ms para asegurar el tracking en tiempo real, se validará durante el monitoreo del ejercicio.
- **RNF2 Visibilidad:** La interfaz debe garantizar que los indicadores de proceso se puedan visualizar a una distancia mínima de dos metros, validando en pruebas donde el usuario interactúa con la aplicación alejado del dispositivo para simular el entorno real.
- **RNF3 Sincronización:** La persistencia de los datos históricos a Cloud Firestore en máximo 3 segundos después de finalizar la serie, validando que los datos se suban a la base de datos y comprobando en la interfaz de Historial del usuario
- **RNF4 Estabilidad:** La aplicación debe operar de forma continua sin presentar bloqueos de interfaz o caídas de fluidez durante rutinas de larga duración.
- **RNF5 Uso de Recursos:** Se establece como criterio de éxito que la aplicación no exceda el 70% de la capacidad de procesamiento (CPU) y memoria RAM del dispositivo durante la ejecución del tracking en tiempo real.

Tablas de requerimiento de información

Tabla 5

Requerimiento de Información IQR_1

Variable	Descripción
IQR_01	Registro y autenticación de usuarios mediante Firebase Authentication.
Versión	1.0 — 23/10/2025
Autores	Ortega Giler Miguel Ángel y Baque Barcia Josthin Ariel
Fuente	Autores del presente documento
Requisito Asociado	RF1, RF2
Descripción	La aplicación móvil debe permitir el registro e inicio de sesión de nuevos usuarios, validando la información ingresada (nombre, correo, edad y nivel) y almacenando los datos en Firebase Authentication y Firestore.
Datos específicos	Nombre, edad, correo electrónico, nivel de experiencia, credenciales de acceso y token de sesión.

Variable	Descripción
Importancia	Vital
Estabilidad	Alta

Nota. Elaboración Propia.

Tabla 6

Requerimiento de Información IQR_2

Variable	Descripción
IQR_02	Visualización de ejercicios y rutinas según el nivel del usuario.
Versión	1.0 — 23/10/2025
Autores	Ortega Giler Miguel Ángel y Baque Barcia Josthin Ariel
Fuente	Autores del presente documento
Requisito	RF2
Asociado	
Descripción	El sistema debe mostrar una lista de ejercicios disponibles y recomendados de acuerdo con el nivel del usuario registrado. Cada ejercicio incluirá nombre, descripción, video demostrativo y errores comunes durante su ejecución.
Datos específicos	Nombre del ejercicio, nivel, descripción, URL de video, lista de errores frecuentes.
Importancia	Alta
Estabilidad	Media

Nota. Elaboración Propia.

Tabla 7

Requerimiento de Información IQR_3

Variable	Descripción
IQR_03	Detección y corrección de posturas corporales en tiempo real.
Versión	1.0 — 23/10/2025
Autores	Ortega Giler Miguel Ángel y Baque Barcia Josthin Ariel
Fuente	Autores del presente documento
Requisito	RF3. RF4, RF5
Asociado	

Variable	Descripción
Descripción	El sistema debe activar la cámara del dispositivo para capturar la postura del usuario durante el ejercicio y analizar los puntos articulares mediante la librería MediaPipe Pose Landmarker. En caso de detectar desviaciones, debe mostrar mensajes de corrección visual en la pantalla.
Datos específicos	Coordenadas de puntos corporales, nombre del ejercicio, puntos de referencia (iniciales, intermedios finales), alertas visuales y registro temporal de la sesión.
Importancia	Vital
Estabilidad	Media

Nota. Elaboración Propia.

4.4.2 Diseño

4.4.2.1 Arquitectura del sistema

El sistema adopta una arquitectura Cliente–Servidor en la nube, donde la aplicación Android, desarrollada en Kotlin, actúa como cliente enriquecido y Firebase como backend como servicio. La estructura interna del cliente sigue el patrón MVVM o también conocido como Model–View–ViewModel, lo que favorece la separación de responsabilidades, el mantenimiento y la evolución de funcionalidades.

Debido a que Firebase Firestore es una base de datos NoSQL orientada a documentos, el sistema se beneficia de un esquema flexible que facilita la iteración sobre entidades (usuarios, ejercicios, rutinas y sesiones) sin migraciones complejas.

La aplicación requiere conexión a Internet para autenticación, consulta del catálogo (ejercicios/rutinas) y registro de sesiones. El análisis de pose se ejecuta localmente en el dispositivo, pero su uso en el flujo de la aplicación está condicionado a la sesión activa en línea.

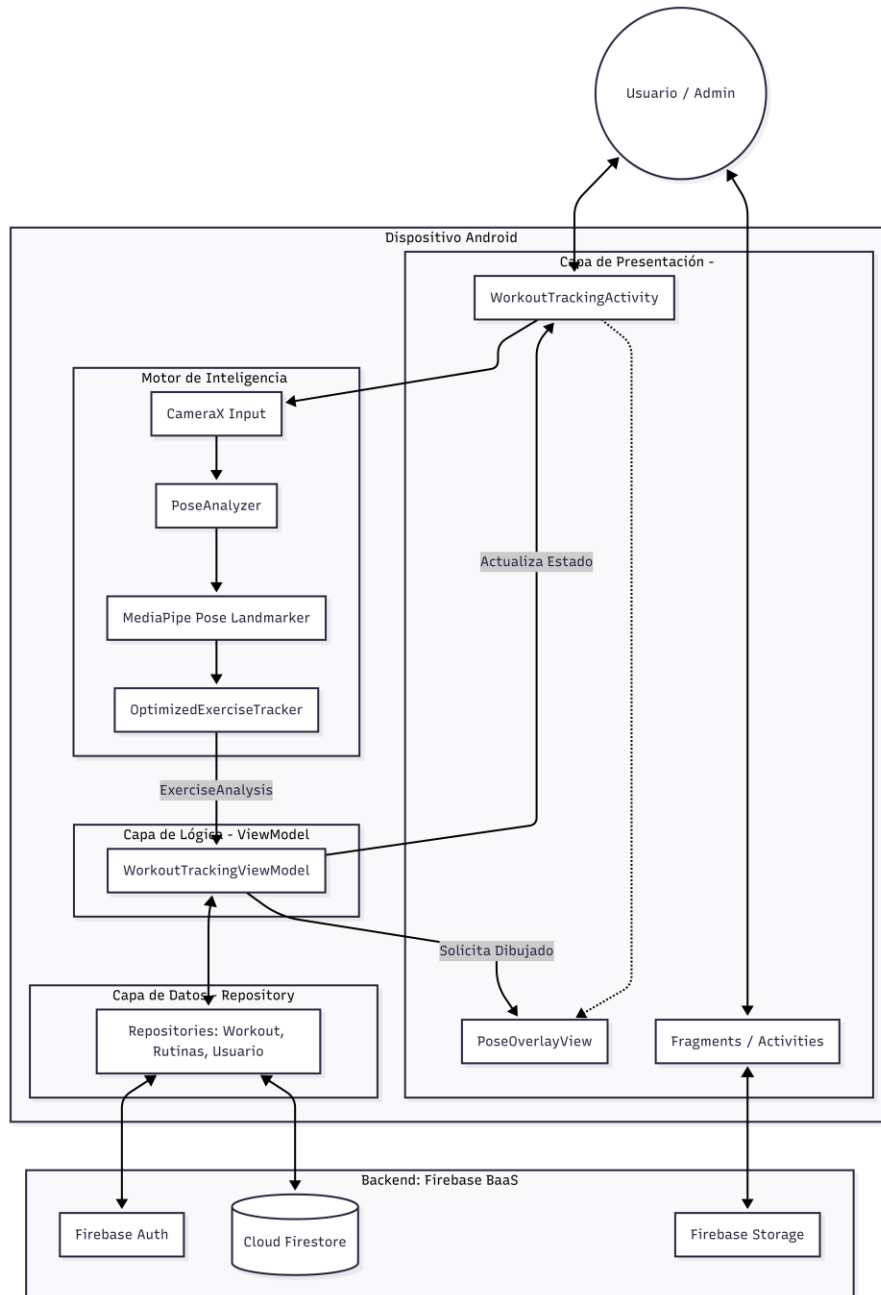
Descripción general de la arquitectura:

- **Capa de Vista (View):** La interfaz está construida con layouts XML y componentes Material Design. Incluye las pantallas de registro, inicio de sesión, selección de ejercicios, seguimiento con cámara, estadísticas y administración, mostrando superposiciones (overlay) con feedback visual durante el tracking. La más importante es PoseOverlayView la cual se encarga de renderizar los puntos de referencia postural y dar el feedback durante el ejercicio.

- **Capa de Lógica de Negocio (ViewModel):** Los ViewModels orquestan el estado de las interfaces y la sesión de entrenamiento, coordinan el pipeline de tracking (inicio, pausa y finalización) y agregan las métricas del ejercicio. El acceso a datos se delega a repositorio. El WorkoutTrackingViewModel actúa como el orquestador central, manteniendo el estado de la sesión (_workoutSession) y procesando los eventos enviados por el motor de tracking.
- **Capa de Datos (Model + Repositorios):** El modelo define clases de datos para los usuarios, ejercicios, rutinas, sesiones y análisis de ejercicio. Los repositorios encapsulan las tecnologías de Firebase como lo son: Authentication, Firestore y Storage, y exponen las Apis reactivas a los ViewModels. Las configuraciones de ejercicios se cargan desde un archivo JSON en los que están fases, rangos angulares y feedback mediante ExerciseConfigManager. El tracking es local con CameraX + MediaPipe Pose Landmarker y dos enfoques de análisis: OptimizedExerciseTracker máquina de estados/umbrales y ConfigurableExerciseAnalyzer que son las reglas parametrizadas desde JSON.
- **Capa de IA:** Se encarga del procesamiento de visión por computadora mediante el PoseAnalyzer, que transforma frames de CameraX en 33 puntos clave mostrados en la **¡Error! No se encuentra el origen de la referencia.** usando MediaPipe. Luego el OptimizedExerciseTracker aplica lógica biométrica para calcular ángulos y validar la técnica del ejercicio.

Ilustración 3

Diagrama de Arquitectura del Sistema



Nota. El diagrama demuestra mas a detalle la interacción de las partes importantes de la arquitectura. Elaboración propia

Tabla 8*Componentes principales del sistema*

Componente	Descripción	Tecnología Asociada
Autenticación	Gestiona el registro e inicio de sesión de usuarios.	Firebase Authentication
Base de datos en la nube	Almacenamiento de perfiles, rutinas y el historial de sesiones (workout_sessions)	Firebase Firestore
Almacenamiento de archivos	Guarda imágenes y videos de demostración de ejercicios.	Firebase Storage
Motor de Visión Artificial	Análisis de frames en tiempo real para detección de 33 puntos corporales	MediaPipe Pose Landmarker + CameraX
Capa de Abstracción de Datos	Gestión centralizada de la información local y remota (Repositorios).	WorkoutRepository / RutinasRepository
Interfaz y Feedback	Presentación de datos y renderizado del esqueleto gráfico sobre el usuario	XML / PoseOverlayView

Nota. Describe los principales componentes para el funcionamiento de la aplicación así como la tecnología encargada. Elaboración Propia.

4.4.2.2 Modelo Entidad Relación de la Base de Datos del sistema

El diseño lógico de datos define cómo se organiza la información dentro del sistema, dado que Firebase utiliza una base de datos NoSQL, los datos se almacenan en colecciones y documentos, en lugar de tablas relacionales.

Ilustración 4

Diseño lógico de base de datos

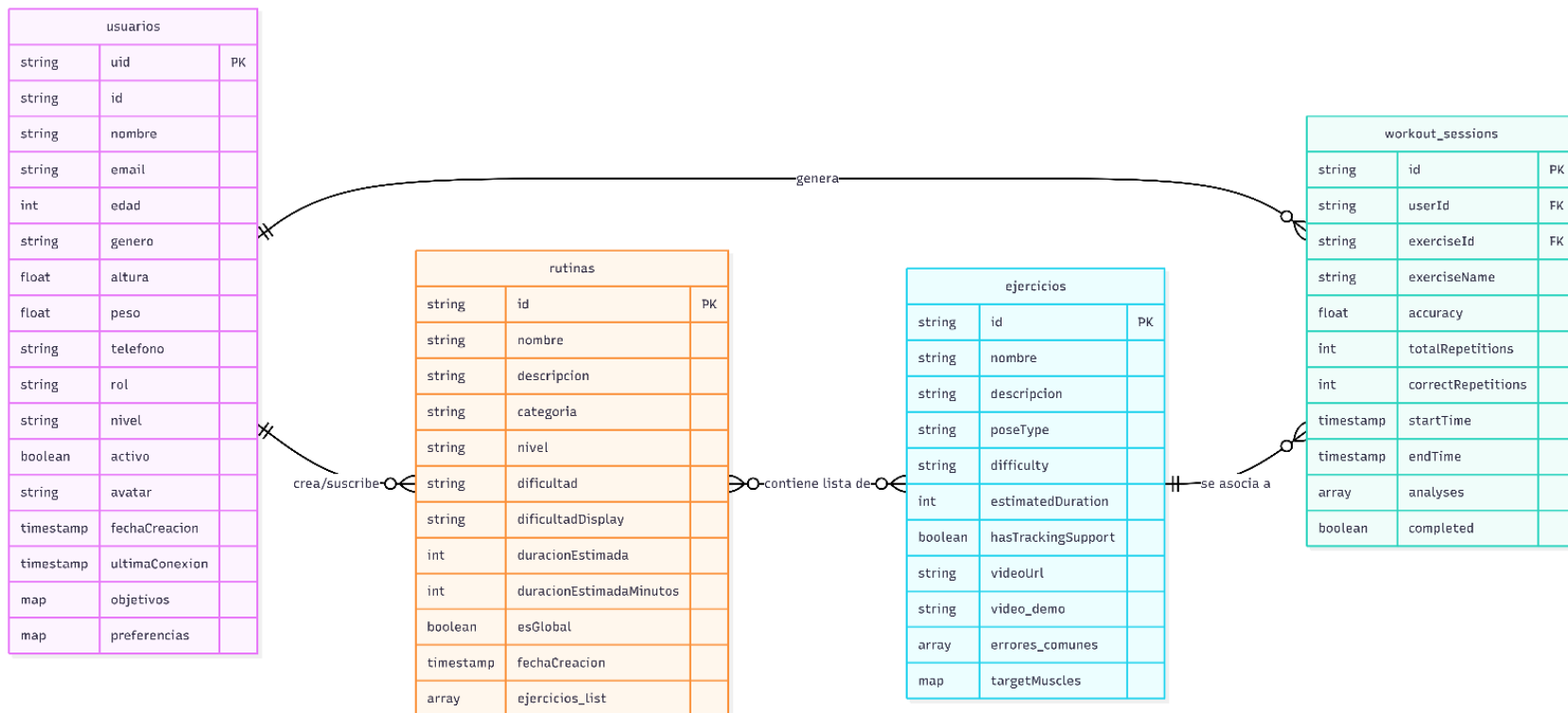


Tabla 9

Descripción de Colecciones

Colección	Descripción
Usuarios	Centraliza la información del perfil, datos biométricos y las preferencias de personalización de la interfaz.
Ejercicios	Actúa como repositorio de metadatos técnicos (landmarks) y recursos multimedia necesarios para el motor de tracking.
Rutinas	Define la organización lógica de los entrenamientos, permitiendo agrupar ejercicios por niveles de dificultad.
workout_sessions	Almacena la telemetría de desempeño, resultados de precisión y registros temporales de cada sesión.

Nota. Describe las particularidades de las colecciones de la base de datos. Elaboración Propia.

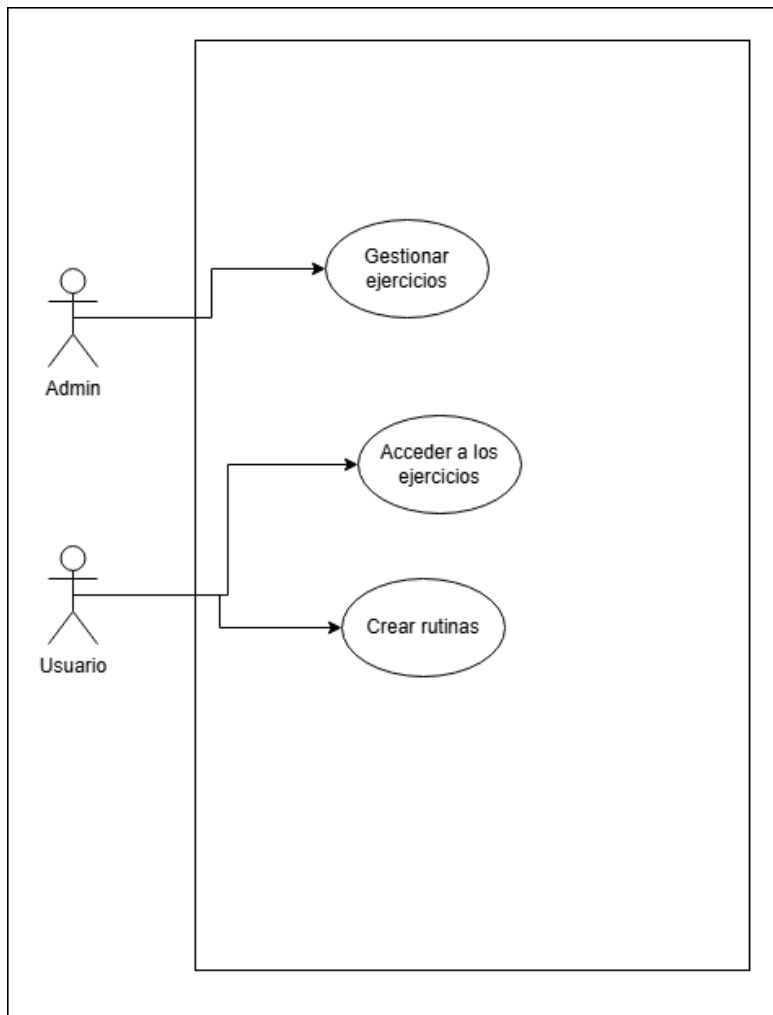
Relaciones conceptuales:

- **Vínculo Usuario-Rutina:** Las rutinas están ligadas al usuario según su experiencia. De este modo los usuarios pueden buscar rutinas que se ajusten a su experiencia
- **Composición de Rutinas:** Para mantener las colecciones ligeras, las rutinas no almacenan los datos de los ejercicios de forma anidada, en su lugar utilizan una lista de referencias o ID's que apuntan a la colección principal de ejercicios.
- **Generación de Desempeño:** El progreso se gestione en la colección workout_sessions. Cuando se termina un ejercicio se crea un registro con el historial de su entrenamiento.

4.4.2.3 Casos de Uso

Ilustración 5

Caso de uso de usuarios del sistema



Nota. El diagrama muestra a los 2 tipos de usuarios, el administrador encargado de la gestión de ejercicios e información y el usuario final que usa la aplicación. Elaboración propia

Tabla 10

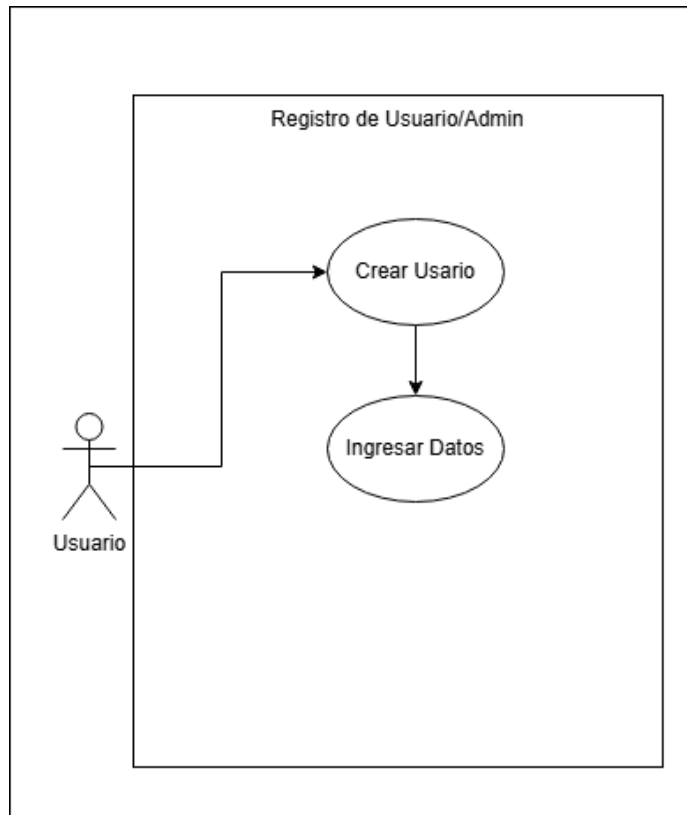
Caso de uso de usuarios del sistema

Autor	Descripción
Admin	Encargado de gestionar los ejercicios de la aplicación, incluyendo agregar, actualizar y eliminar los ejercicios.
Usuario	El que usa la aplicación, puede consultar todo tipo de información de los ejercicios, los detalles específicos, crear rutinas de ejercicios y realizar los ejercicios con asistencia del tracking, el usuario es el responsable de crear su propia cuenta.

Nota. Elaboración Propia.

Ilustración 6

Caso de uso de Registro de Usuarios



Nota. El diagrama muestra el proceso de registro de los usuarios en la aplicación.

Tabla 11

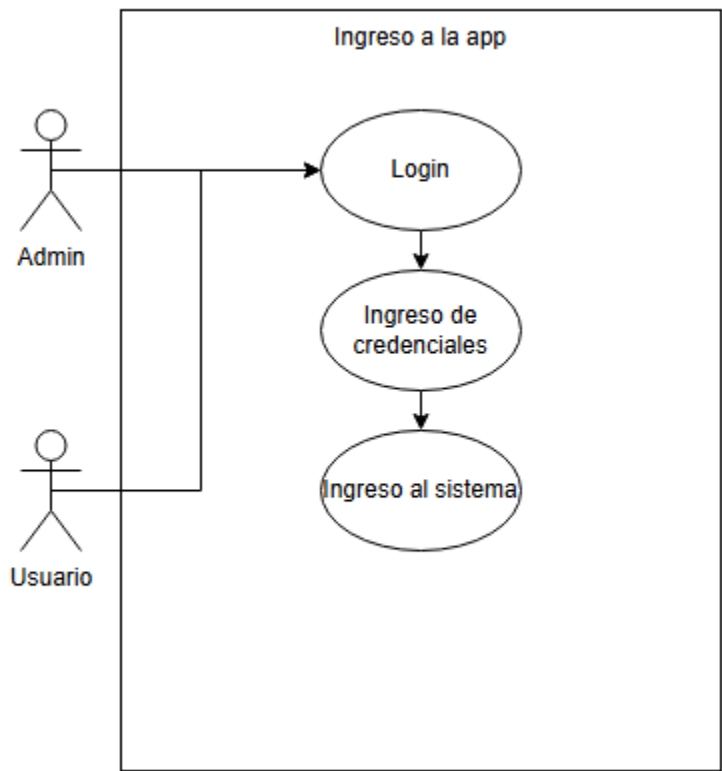
Caso de uso de Registro de Usuarios

Referencia	CU-01
Nombre	Crear usuario
Descripción	Proceso de creación de usuario
Actores	Usuario
Precondición	El usuario no está creado
Postcondición	Se habrá creado el usuario
Flujo Básico	El usuario ingresa sus datos para crear una cuenta
Flujo Alternante	Si hay algún problema durante el proceso se notificará con un mensaje de error

Nota. Elaboración Propia.

Ilustración 7

Caso de uso de Ingreso del Usuario



Nota. El diagrama describe el proceso de inicio de sesión de los usuarios. Elaboración propia

Tabla 12

Caso de uso de Ingreso del Usuario

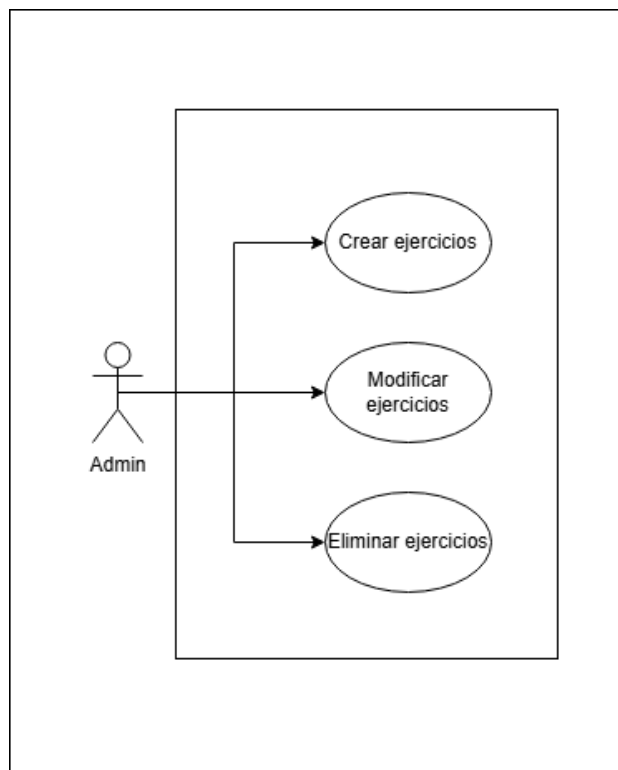
Referencia	CU-02
Nombre	Inicio de sesión
Descripción	Proceso de inicio de sesión
Actores	Usuario / Admin
Precondición	Tener una cuenta creada
Postcondición	Ingreso a la aplicación

Flujo Básico	El usuario / Admin ingresa sus credenciales Intenta el ingreso El usuario ingresa a la aplicación
Flujo Alternante	Si las credenciales son incorrectas o no existen se mostrará un mensaje de error

Nota. Elaboración Propia.

Ilustración 8

Caso de Uso de operaciones del Admin



Nota. El diagrama muestra los tipos de operaciones que puede realizar el administrador en la aplicación. Elaboración propia.

Tabla 13

Caso de Uso de creación de ejercicio

Referencia	CU-03
Nombre	Crear Ejercicios
Descripción	Creación de ejercicios para la aplicación

Actores	Admin
Precondición	Que el ejercicio no exista
Postcondición	Se crea un nuevo ejercicio
Flujo Básico	El Admin crea / añade un nuevo ejercicio a la aplicación, el ejercicio puede o no puede contener el apartado para tracking, en cualquier caso, el ejercicio mostrara una ilustración de cómo se realiza el ejercicio y otros detalles de este
Flujo Alternante	Si existe algún problema con el ejercicio se mostrará un mensaje de error y el ejercicio no se creará.

Nota. Elaboración Propia.

Tabla 14

Caso de Uso de edición de ejercicio

Referencia	CU-04
Nombre	Editar ejercicios
Descripción	Edición de ejercicios existentes en la aplicación
Actores	Admin
Precondición	Que exista el ejercicio
Postcondición	El ejercicio existente se modifica
Flujo Básico	El Admin modifica algún parámetro específico del ejercicio, entre ellos incluye el apartado para tracking, este se crea aparte e involucra modificar el código directamente y añadir JSON de configuración
Flujo Alternante	Si existe algún problema con el ejercicio se mostrará un mensaje de error y el ejercicio no se actualizará

Nota. Elaboración Propia.

Tabla 15

Caso de Uso de eliminación de ejercicio

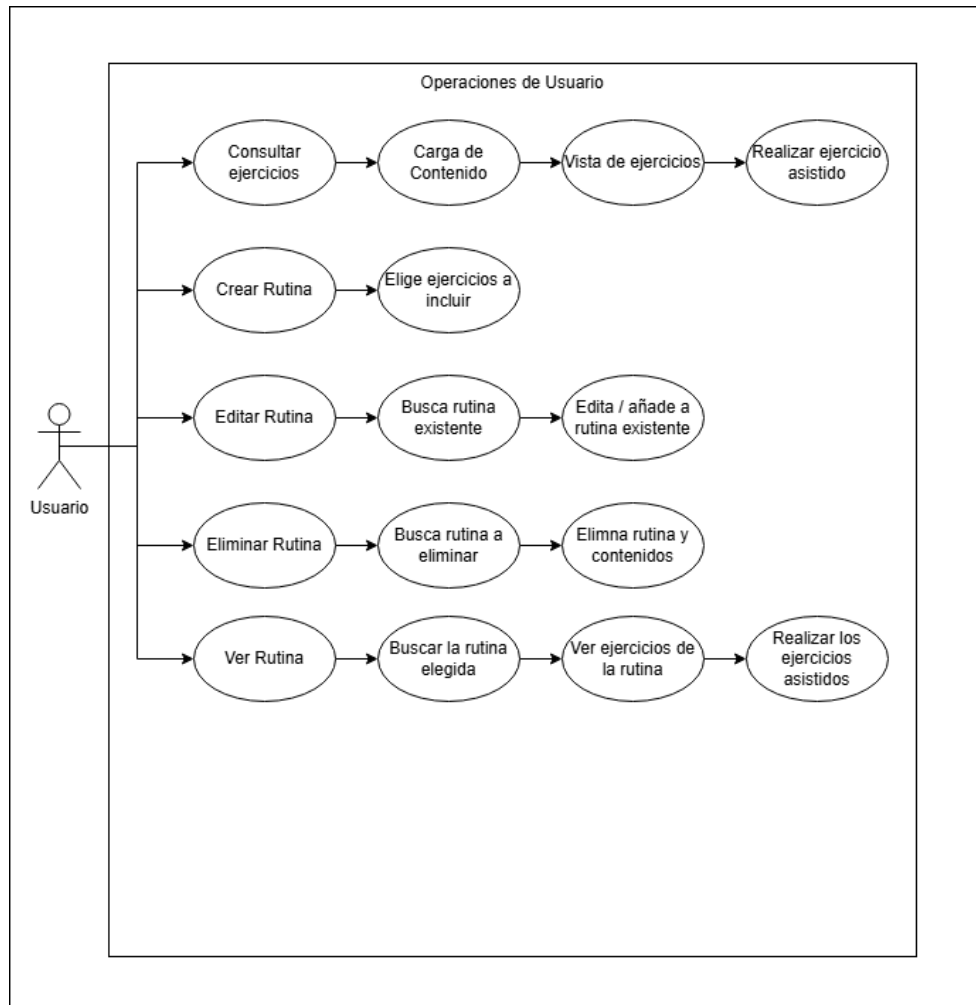
Referencia	CU-05
Nombre	Eliminar ejercicio
Descripción	Eliminación de un ejercicio existente en la aplicación
Actores	Admin
Precondición	Que el ejercicio exista
Postcondición	El ejercicio es eliminado

Flujo Básico	El Admin elije un ejercicio a eliminar, luego se pedirá una confirmación de si quiere eliminarlo, al aceptar se eliminará el ejercicio
Flujo Alternante	Si el Admin cancela la eliminación no ocurrirá nada y el ejercicio seguirá en la aplicación

Nota. Elaboración Propia.

Ilustración 9

Caso de Uso de Operaciones del Usuario



Nota. El diagrama muestra el flujo funcional del usuario, desde la consulta de ejercicios hasta la ejecución de ejercicios con tracking. Elaboración propia,

Tabla 16*Caso de Uso de consulta de ejercicio*

Referencia	CU-06
Nombre	Consultar ejercicios
Descripción	Consulta de los ejercicios que hay en la aplicación, vista general de todos los que existen
Actores	Usuario
Precondición	Que existan ejercicios
Postcondición	Vista general de los ejercicios y si el ejercicio lo permite hacer el ejercicio asistido sin necesidad de crear una rutina
Flujo Básico	El usuario consulta todos los ejercicios existentes en la aplicación Luego de que se cargue el contenido el usuario puede elegir un ejercicio para verlo a detalle si el usuario desea también puede hacer el ejercicio asistido
Flujo Alternante	Si ocurre un error al cargar los ejercicios o en algún punto del proceso se notificará con un mensaje de error

Nota. Elaboración Propia.

Tabla 17*Caso de Uso de creación de rutina*

Referencia	CU-07
Nombre	Crear Rutina
Descripción	Creación de una rutina personalizada por el usuario
Actores	Usuario
Precondición	No tener una rutina personalizada (o querer crear una nueva)
Postcondición	Creación de rutina personalizada
Flujo Básico	La persona crea una rutina personalizada Luego de cargar los ejercicios de la aplicación elige cuales incluir y crea la rutina
Flujo Alternante	Si ocurre un error al cargar los ejercicios o en algún punto del proceso se notificará con un mensaje de error

Nota. Elaboración Propia.

Tabla 18*Caso de Uso de edición de rutina*

Referencia	CU-08
Nombre	Editar Rutina
Descripción	Editar una rutina personalizada existente
Actores	Usuario
Precondición	Una rutina personalizada existente
Postcondición	La rutina tendrá menos o más ejercicios, o datos como el nombre de la rutina cambiaran
Flujo Básico	El usuario elige editar una rutina existente Luego el usuario elige si añadir o quitar ejercicios de la rutina, cambiar datos como el nombre de la rutina Al guardar los cambios la rutina habrá cambiado
Flujo Alternante	Si ocurre un error al cargar la rutina o en algún punto del proceso se notificará con un mensaje de error

Nota. Elaboración Propia.

Tabla 19*Caso de Uso de eliminación de rutina*

Referencia	CU-09
Nombre	Eliminar Rutina
Descripción	Eliminación de Rutina personalizada existente
Actores	Usuario
Precondición	Que exista la rutina a eliminar
Postcondición	La rutina será eliminada
Flujo Básico	EL usuario puede decidir eliminar una rutina si ya no la necesita o no le gusta Al querer eliminarla saltará un aviso de eliminación al aceptar la rutina se eliminará
Flujo Alternante	Si ocurre un error al cargar la rutina o en algún punto del proceso se notificará con un mensaje de error

Nota. Elaboración Propia.

Tabla 20*Caso de Uso de consulta de rutina*

Referencia	CU-10
Nombre	Ver Rutina
Descripción	Consulta de una Rutina personalizada existente
Actores	Usuario
Precondición	Que exista la rutina a consultar / usar
Postcondición	La rutina será eliminada
Flujo Básico	El usuario al entrar a su rutina aparecerán los ejercicios que tiene dentro El usuario puede elegir por cual ejercicio comenzar o ver detalles del mismo Si el usuario desea puede comenzar con los ejercicios asistidos por tracking dentro de su Rutina Cuando el usuario se equivoque o no complete la repetición aparecerá un mensaje en pantalla con un consejo para corregir a manera de feedback Una vez terminadas las repeticiones deseadas podrá salir del ejercicio y continuar con el siguiente En cualquier punto de la rutina el usuario puede salir
Flujo Alternante	Si ocurre un error al cargar la rutina o en algún punto del proceso se notificará con un mensaje de error, también se notificará al usuario durante el proceso del ejercicio asistido los errores que cometa y consejos (feedback) para corregir dichos errores

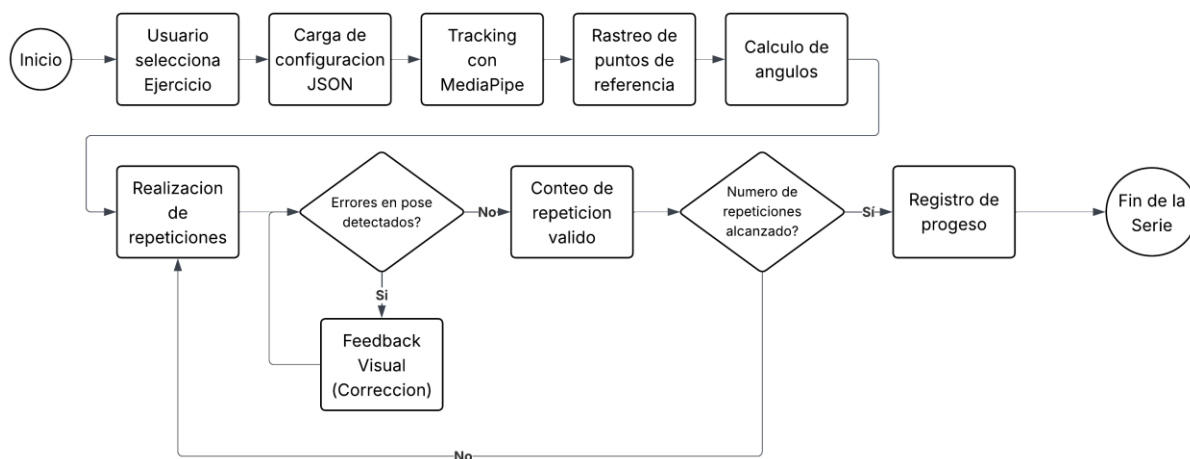
Nota. Elaboración Propia.

4.4.3.4 Flujo Lógico del proceso de Tracking

En este apartado se muestra la secuencia que permite la detección y validación de los ejercicios en tiempo real

Ilustración 10

Secuencia lógica del proceso de Tracking



Nota. El proceso inicia con la carga de parámetros desde un archivo JSON y utiliza la biblioteca MediaPipe para el rastreo de landmarks en tiempo real. Elaboración propia con LucidChart.

El Flujo del proceso se divide en 4 etapas críticas:

1. **Inicialización y Configuración:** Una vez que el usuario selecciona un ejercicio el sistema invoca al ExerciseConfigManager para cargar un archivo JSON específico. Este archivo contiene los umbrales de los ángulos articulares, las fases del movimiento y los mensajes de error asociados a cada postura incorrecta.
2. **Procesamiento de Visión Artificial:** Al activar la cámara mediante CameraX, los fotogramas son procesados localmente por el motor de MediaPipe Pose Landmarker. Este componente extrae las coordenadas de los 33 puntos de referencia corporales en tiempo real, permitiendo reconstruir la postura del usuario de forma digital.
3. **Análisis Postural:** El componente OptimizedExerciseTracker recibe las coordenadas y utiliza algoritmos trigonométricos para calcular los ángulos entre las articulaciones.

Si los ángulos detectados se encuentran fuera del rango definido en el JSON, el sistema activa la retroalimentación visual y auditiva para corregir al usuario. Si la postura es correcta, el sistema avanza en una máquina de estados para validar la repetición.

4. Final y Persistencia: Al completarse la serie o ser finalizada por el usuario, el sistema recopila la telemetría del desempeño. Estos datos se envían a la colección `workout_sessions` en Firebase Firestore para su almacenamiento histórico y posterior consulta en el módulo de progreso.

4.4.3.5 Matriz de Trazabilidad de Requisitos

A continuación, se muestra la matriz que vincula los requisitos definidos con los casos de uso

Tabla 21

Matriz de Trazabilidad de Requisitos

ID Requisito	Descripción del Requisito	Caso de Uso Asociado	Métrica de Validación
RF1	Gestión de registro y acceso para usuarios principiantes.	CU-01, CU-02	Firebase Auth / Éxito en el 100% de los accesos válidos.
RF1	Acceso administrativo para la gestión de contenidos.	CU-02	Firebase Auth / Control de roles y permisos.
RF2	Consulta de la lista de ejercicios según el nivel de experiencia del usuario.	CU-06	Cloud Firestore / Verificación de integridad de datos.
RF2	Gestión del repositorio: creación, edición y eliminación de ejercicios.	CU-03, CU-04, CU-05	Cloud Firestore / Persistencia atómica de documentos.
RF3	Gestión de rutinas: creación, edición y eliminación de planes personalizados.	CU-07, CU-08, CU-09	Cloud Firestore / Vínculo Usuario-Rutina.
RF4, RF5, RF6	Ejecución asistida: activación de cámara, rastreo de pose constante y emisión de feedback visual/auditivo.	CU-10	MediaPipe + TTS / Latencia < 200 ms y respuesta inmediata.

ID Requisito	Descripción del Requisito	Caso de Uso Asociado	Métrica de Validación
RNF1, RNF4	Calidad técnica: eficiencia en procesamiento y estabilidad de la interfaz durante el uso.	Todos los CUs	Monitoreo de frames (> 24 FPS) y gestión asíncrona.
RNF2, RNF3	Calidad de uso: visibilidad de indicadores a distancia y sincronización de datos históricos.	CU-06, CU- 10	Distancia > 2 m / Sincronización en la nube < 3 s.

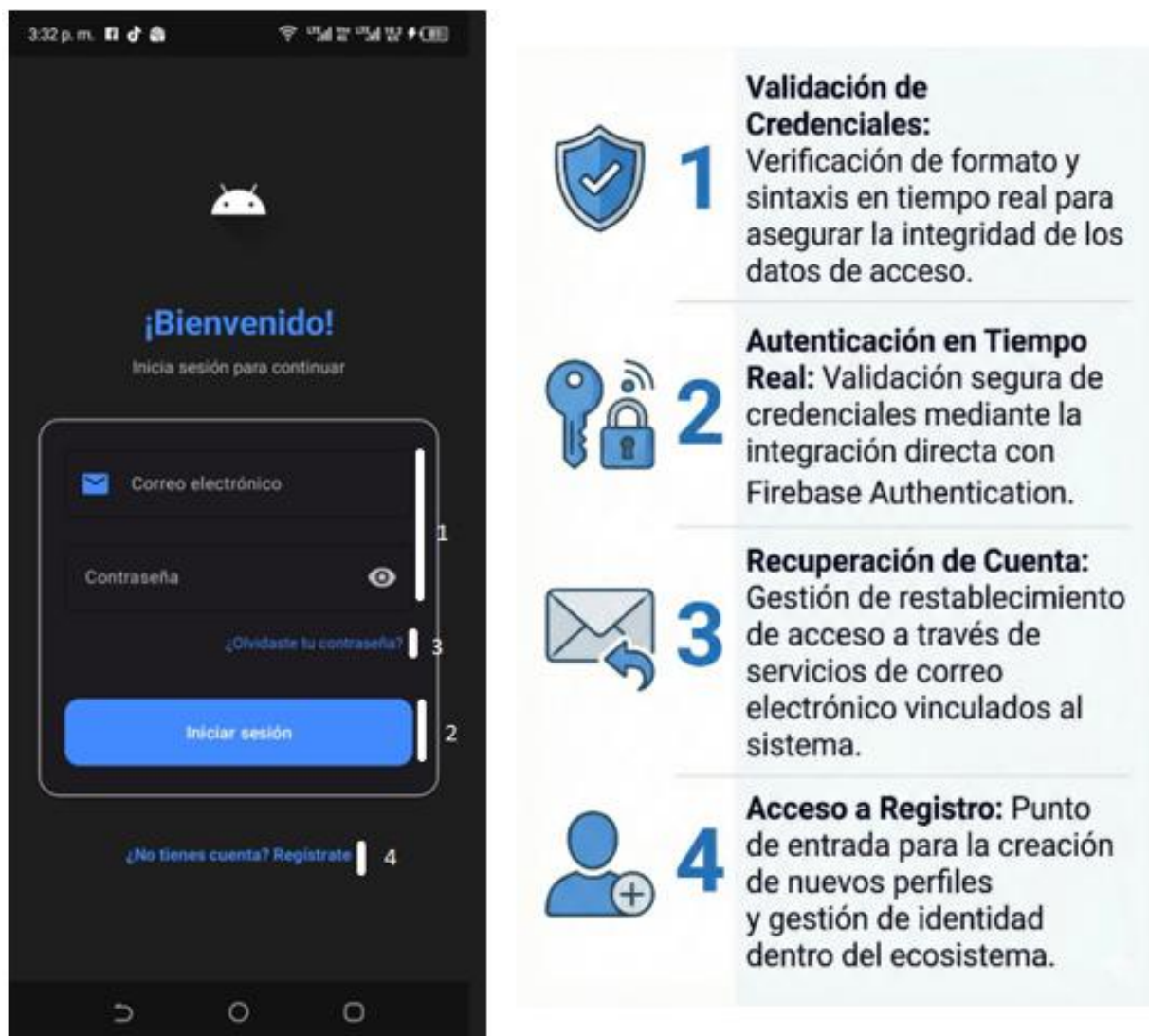
Nota. Matriz que traza la relación entre los requisitos y los casos de uso. Elaboración Propia.

4.4.3 Implementación

4.4.3.1 Prototipos de Interfaces de la Propuesta de Aplicación móvil

Ilustración 11

Interfaz de Inicio de Sesión



Nota. Interfaz de inicio de sesión, usa validaciones con Firebase Auth para garantizar la integridad de las credenciales. Elaboración Propia asistida por Gemini para ilustración del proceso lógico.

Ilustración 12

Interfaz de Registro de cuenta

8:19 p. m.

Crear Cuenta

Nombre completo

Edad

Correo electrónico

Nivel

Contraseña

Confirmar contraseña

Crear Cuenta

1

2

3

4



1

Validación de Entradas:
Sanitización y verificación de datos (Regex) mediante vinculación reactiva para asegurar la integridad previa a la persistencia.



2

Categorización de Usuario:
Selector de nivel (Material Design) para la segmentación de perfiles y escalabilidad del motor de IA.



3

Seguridad de Credenciales:
Validación de concordancia y robustez de claves con componentes de visibilidad dinámica.



4

Persistencia Atómica:
Registro de credenciales en Firebase Auth y creación simultánea de perfil en Firestore (Integridad Referencial).

Nota. Formulario de datos para el registro validado con expresiones regulares. Elaboración Propia asistida por Gemini para ilustración del proceso lógico.

Ilustración 13

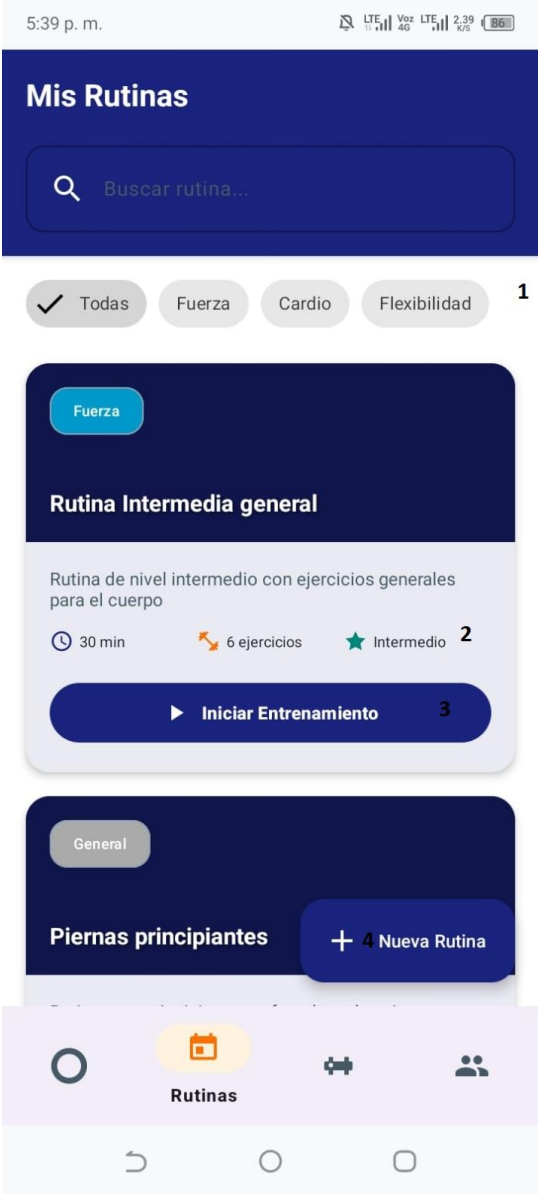
Interfaz de perfil del usuario



Nota. Panel central para la gestión del perfil del usuario, permite visualización básica de meticas acumuladas y la configuración de parámetros de accesibilidad del usuario. Elaboración Propia asistida por Gemini para ilustración del proceso lógico.

Ilustración 14

Interfaz de las rutinas del usuario



The screenshot shows the 'Mis Rutinas' app interface. At the top, there's a status bar with the time 5:39 p. m. and various icons. Below it, the app title 'Mis Rutinas' is displayed. A search bar with the placeholder 'Buscar rutina...' is present. Below the search bar, there are four category filters: 'Todas' (selected), 'Fuerza', 'Cardio', and 'Flexibilidad'. The main content area shows a list of routines. The first routine is 'Rutina Intermedia general', which is a 30-minute intermediate routine with 6 exercises. It has a '30 min' duration, '6 ejercicios', and 'Intermedio' difficulty. A 'Iniciar Entrenamiento' button is visible. The second routine is 'Piernas principiantes', which is a 'General' routine. A '+ Nueva Rutina' button is visible next to it. At the bottom, there's a navigation bar with four icons: a home icon, a calendar icon (labeled 'Rutinas'), a settings icon, and a profile icon.

-  **Filtrado Dinámico:** Clasificación de entrenamientos por categorías (Fuerza, Cardio) mediante consultas optimizadas a la base de datos.
-  **Gestión de Metadatos:** Navegación hacia el módulo de visión artificial, transfiriendo los parámetros de configuración del ejercicio a la IA.
-  **Disparador de Tracking:** Navegación de parámetros visión artificial, duración, volumen de ejercicios y nivel de dificultad técnica.
-  **Inyección de Datos NoSQL:** Interfaz para la creación de nuevos documentos de rutinas personalizadas con persistencia directa en la nube.

Nota. Catalogo de rutinas personalizadas con filtrado dinámico por categorías. Elaboración propia. Elaboración Propia asistida por Gemini para ilustración del proceso lógico.

Ilustración 15

Interfaz de creación de rutinas del usuario

The screenshot shows a mobile application interface for creating a new routine. At the top, the status bar displays the time 7:37 p.m. and various connectivity icons. The app title 'Nueva Rutina' is centered at the top of the screen.

The interface is divided into three main sections:

- Detalles Generales 1:** This section contains three input fields: 'Nombre de la rutina' (Routine Name), 'Categoría' (Category) with a dropdown menu currently set to 'General', and 'Descripción (Opcional)' (Optional Description).
- Dificultad 2:** This section displays three difficulty level buttons: 'Fácil' (Easy) with a checkmark, 'Intermedio' (Intermediate), and 'Difícil' (Difficult).
- Plan de Entrenamiento 3:** This section features a 'Guardar Rutina' (Save Routine) button with a checkmark icon.

At the bottom of the screen, there is a navigation bar with four icons: a home icon, a calendar icon, a dumbbell icon, and a profile icon labeled 'Perfil'. The Android system navigation bar is visible at the very bottom.

To the right of the screenshot, a diagram illustrates the three steps of the routine creation process:

- Definición de Metadatos:** Captura de nombre y descripción para la identificación única y organización lógica del plan de entrenamiento. (Icon: Document with Title and Description fields)
- Nivel de Intensidad:** Parámetro de dificultad técnica para ajustar el seguimiento del progreso y la carga de trabajo sugerida. (Icon: Dumbbell with an upward arrow)
- Persistencia del Plan:** Almacenamiento del objeto de rutina en Firebase Firestore vinculado permanentemente al perfil del usuario. (Icon: Cloud with an upward arrow and a user profile icon)

Nota. Interfaz donde el usuario crea sus rutinas escogiendo ejercicios previamente cargados a la aplicación por el Administrador. Elaboración Propia asistida por Gemini para ilustración del proceso lógico.

Ilustración 16

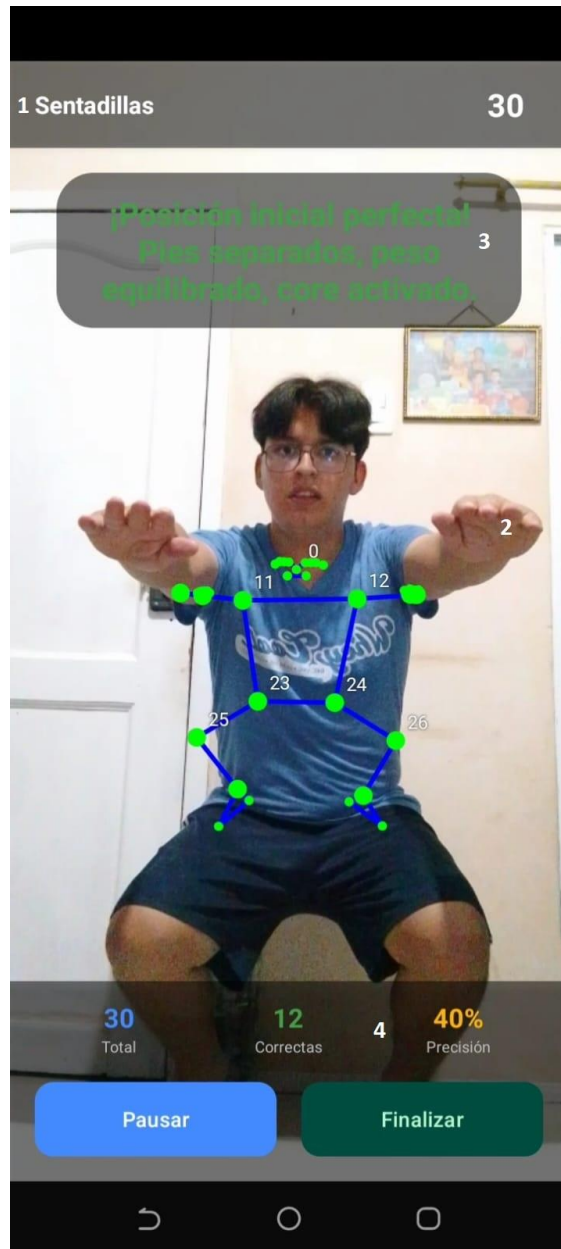
Interfaz de ejercicios



Nota. Repositorio de ejercicios donde se puede iniciar el tracking del ejercicio sin necesidad de crear una rutina previa, muestra información específica sobre cada ejercicio. Elaboración Propia asistida por Gemini para ilustración del proceso lógico.

Ilustración 17

Interfaz del ejercicio asistido

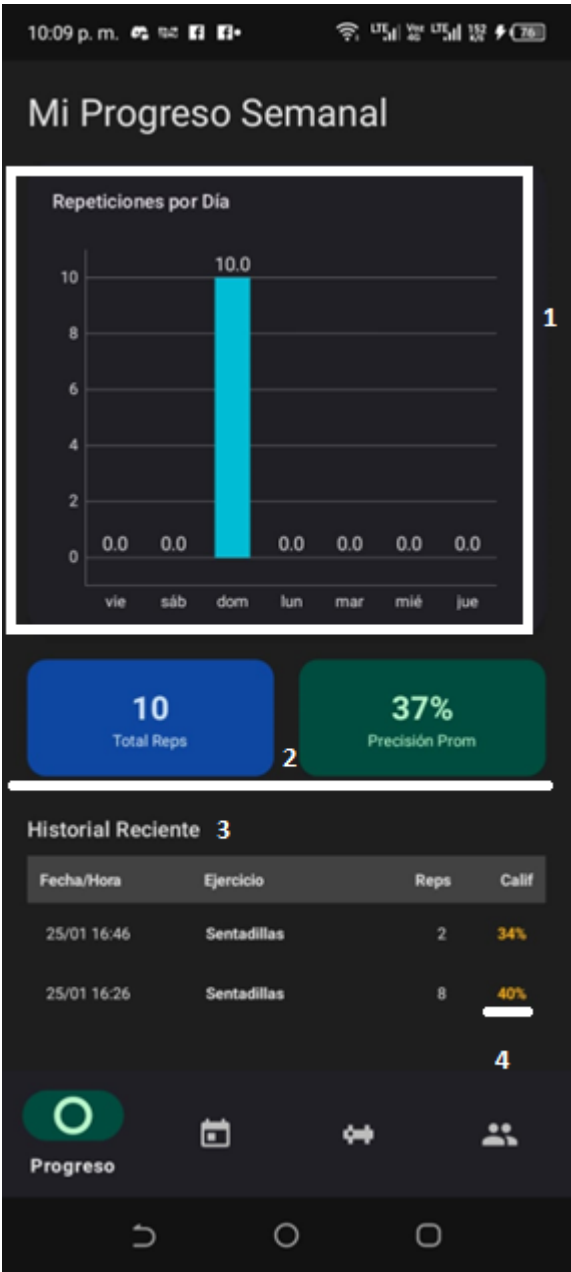


Nota. Interfaz de ejecución en tiempo real que integra el procesamiento local de MediaPipe para la detección de los 33 landmarks corporales y la emisión de feedback visual/auditivo.

Elaboración Propia asistida por Gemini para ilustración del proceso lógico.

Ilustración 18

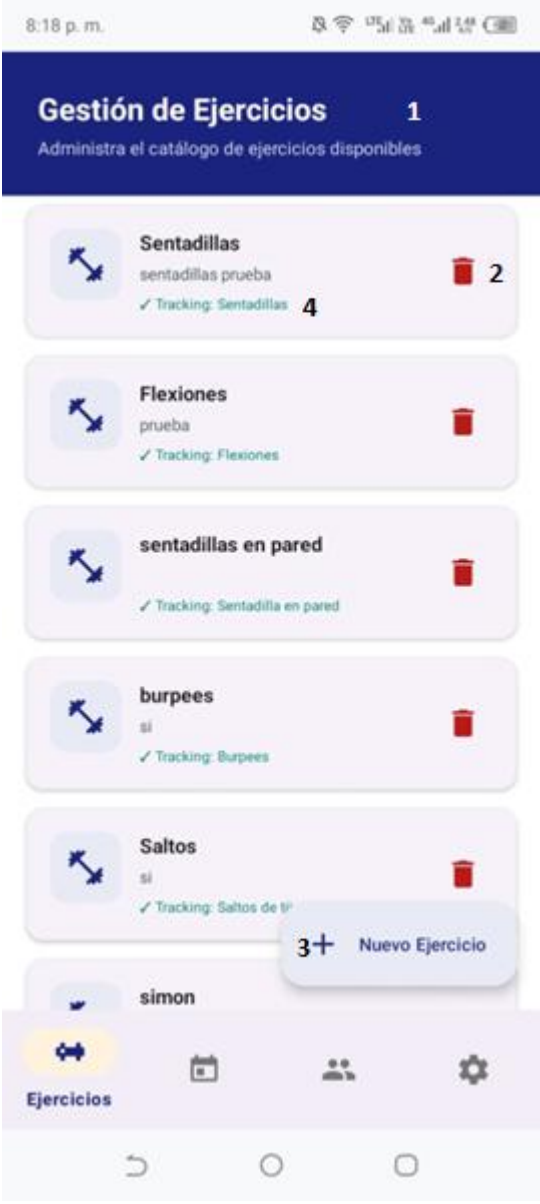
Interfaz de progreso de ejercicios



Nota. Visualización del rendimiento semanal mediante gráficos que muestran la precisión de las posturas y la cantidad de repeticiones hechas durante la sesión. Elaboración Propia asistida por Gemini para ilustración del proceso lógico.

Ilustración 19

Interfaz de Gestión de ejercicios del Administrador



- **1**
Control de Repositorio:
Visualización y mantenimiento del catálogo global de ejercicios almacenado en Firebase Firestore.
- **2**
Depuración de Datos:
Funcionalidad para eliminar registros obsoletos, garantizando la optimización y relevancia del contenido del sistema.
- **3**
Escalabilidad del Catálogo:
Punto de entrada para la inyección de nuevos tipos de entrenamiento con persistencia directa en la base de datos NoSQL.
- **4**
Parámetros de Detección:
Configuración técnica que define el tipo de pose y el modelo de visión artificial (MediaPipe) asociado a cada ejercicio.

Nota. Panel de control de ejercicios y ajustes de parámetros de detección de cada ejercicio.

Elaboración Propia asistida por Gemini para ilustración del proceso lógico.

Ilustración 20

Interfaz de gestión de usuarios del Administrador

8:17 p. m.

Gestión de Usuarios

2 Total Usuarios

2 Activos Hoy

Buscar usuarios...

Todos Administradores Usuarios

Josthin Ba... Usuario
josthinjan@gmail.com
Fecha no disponible Habilitado

Josthin Baq... Admin
josthinar96@gmail.com
Fecha no disponible

Nuevo Usuario

Usuarios

- Métricas de Control:** Visualización en tiempo real del total de cuentas registradas y el volumen de usuarios activos por jornada.
- Filtrado Multicapa:** Localización de perfiles mediante consultas dinámicas por nombre, correo electrónico o privilegios de acceso (Admin/User).
- Control de Estado:** Indicador visual de la disponibilidad de cuenta en el ecosistema, permitiendo la activación o suspensión rápida del acceso.
- Jerarquía de Permisos:** Diferenciación técnica de usuarios finales y personal administrativo para la gestión de seguridad del sistema.
- Provisionamiento de Cuentas:** Acceso al módulo de creación manual de perfiles para el registro controlado de nuevos usuarios administrativos o deportistas.

Nota. Módulo de monitoreo de las cuentas registradas en la aplicación, creación manual de usuarios, control de estados y asignación de rol de Administrador. Elaboración Propia asistida por Gemini para ilustración del proceso lógico.

4.4.4 Pruebas

4.4.4.1 Entorno de Pruebas

Para garantizar la validez de los resultados y la compatibilidad del sistema las pruebas de aceptación y rendimiento se ejecutaron principalmente en 2 dispositivos móviles con diferentes especificaciones las cuales son las siguientes:

- **Dispositivo A:** Infinix Hot 20s
- **Dispositivo B:** Tecno Spark 10C
- **Condiciones ambientales:** Las pruebas se realizaron en entornos de iluminación variable para verificar la robustez del algoritmo de tracking ante cambios de luz.

4.4.4.2 Definición de Métricas y Criterios de Aceptación

- Se establecieron indicadores técnicos para los requisitos funcionales y no funcionales, cada uno cuenta con una métrica de validación para asegurar la integración del sistema.
- **Rendimiento:** Para los RNF1 y RNF4, latencia de procesamiento inferior a 200 ms y estabilidad de cuadros por segundo en ambos dispositivos de prueba.
- **Precisión Postural:** Para RF5, el mantenimiento del tracking de los 33 landmarks corporales durante la ejecución de los ejercicios, sin importar las variaciones de hardware entre el Tecno e Infinix.
- **Interacción y Feedback:** Para RF4 y RF6, la activación de la cámara en menos de 3 segundos y emisión de alertas auditivas sincronizadas con los errores detectados.
- **Persistencia y Seguridad:** Para RF1, RF3 y RNF3, Sincronización exitosa de perfiles y rutinas con Cloud Firestore en un tiempo menor a 3 segundos tras la acción del usuario.

C

APITULO 5:

EVALUACIÓN DE RESULTADOS

CAPÍTULO 5: EVALUACIÓN DE RESULTADOS

5.1. Introducción

El presente capítulo expone los resultados obtenidos tras el desarrollo y verificación de la aplicación móvil de asistencia de ejercicios. El análisis se fundamenta en los objetivos planteados al inicio de la investigación, describiendo el cumplimiento de los requerimientos funcionales a través de cuatro dimensiones críticas: funcionalidad, usabilidad, eficiencia y portabilidad.

5.2. Evaluación y monitoreo de resultados

Las pruebas de aceptación documentadas a continuación fueron ejecutadas con un grupo pequeño de usuarios pertenecientes al rango de edad de 18 – 30 años, lo que permitió asegurar que el diseño de interfaces, capacidades de tracking y retroalimentación visual y auditiva son adecuadas para las capacidades del público objetivo de la investigación.

5.2.1. Resultados con respecto a la funcionalidad

Esta sección valida que la aplicación ejecute las tareas para la que fue diseñada vinculando las pruebas con sus debidos requisitos.

- **RF1 y RF2 – Gestión de Acceso y Perfiles:** Se verificó la capacidad del sistema para registrar nuevos usuarios y autenticar credenciales mediante Firebase, asegurando el control de acceso a la plataforma.

Tabla 22

Prueba de aceptación crear estudiante

Título	Descripción
Nº De prueba	1
Módulo de prueba	Crear usuario
Desarrollo	Front-End / Back-End (Firebase)
Evento	El sistema debe permitir el ingreso de datos para el registro de un nuevo usuario.
Resultado	Se creó el usuario exitosamente en la base de datos.

Estado	Aprobado
---------------	----------

Nota. Elaboración Propia

Tabla 23

Prueba de aceptación de ingreso de usuario

Título	Descripción
N° De prueba	2
Módulo de prueba	Inicio de sesión
Desarrollo	Front-End / Back-End (Auth)
Evento	El sistema debe permitir el ingreso de credenciales para acceder a la aplicación.
Resultado	El usuario o administrador ingresa correctamente a la aplicación.
Estado	Aprobado

Nota. Elaboración Propia

- **RF4 – Gestión administrativa de los ejercicios:** Se logro implementar la gestión de los ejercicios por parte del administrador.

Tabla 24

Prueba de aceptación de creación de ejercicios

Título	Descripción
N° De prueba	3
Módulo de prueba	Crear Ejercicios
Desarrollo	Front-End / Back-End
Evento	El Administrador debe poder añadir un nuevo ejercicio con su respectiva ilustración y detalles.
Resultado	Se crea un nuevo ejercicio disponible en el catálogo de la aplicación.
Estado	Aprobado

Nota. Elaboración Propia

Tabla 25*Prueba de aceptación edición de ejercicios*

Título	Descripción
N° De prueba	4
Módulo de prueba	Editar ejercicios
Desarrollo	Front-End / Lógica (JSON)
Evento	El Administrador debe poder modificar parámetros específicos y el JSON de configuración para tracking.
Resultado	El ejercicio existente se actualiza con los nuevos parámetros definidos.
Estado	Aprobado

Nota. Elaboración Propia**Tabla 26***Prueba de aceptación eliminación de ejercicio*

Título	Descripción
N° De prueba	5
Módulo de prueba	Eliminar ejercicio
Desarrollo	Front-End / Back-End
Evento	El Administrador debe poder elegir un ejercicio y confirmar su eliminación permanente.
Resultado	El ejercicio es removido de la base de datos y de la aplicación.
Estado	Aprobado

Nota. Elaboración Propia

- **RF3 – Gestión de rutinas Personalizadas:** Se validó que el usuario pudiera crear sus propias rutinas y su persistencia en Cloud Firestore

Tabla 27*Prueba de aceptación consultar ejercicios*

Título	Descripción
--------	-------------

N° De prueba	6
Módulo de prueba	Consultar ejercicios
Desarrollo	Front-End
Evento	El sistema debe mostrar una lista general de todos los ejercicios existentes para el usuario.
Resultado	El usuario visualiza correctamente el catálogo y puede elegir ejercicios para ver sus detalles.
Estado	Aprobado

Nota. Elaboración Propia

Tabla 28

Prueba de aceptación crear rutinas

Título	Descripción
N° De prueba	7
Módulo de prueba	Crear Rutina
Desarrollo	Front-End
Evento	El usuario debe poder seleccionar múltiples ejercicios para conformar una rutina personalizada.
Resultado	Se genera y guarda la nueva rutina personalizada del usuario.
Estado	Aprobado

Nota. Elaboración Propia

Tabla 29

Prueba de aceptación edición de rutina

Título	Descripción
N° De prueba	8
Módulo de prueba	Editar Rutina
Desarrollo	Front-End
Evento	El usuario debe poder añadir, quitar ejercicios o cambiar el nombre de una rutina existente.
Resultado	Los cambios en la rutina se guardan reflejando la nueva configuración de ejercicios.
Estado	Aprobado

Nota. Elaboración propia.

Tabla 30

Prueba de aceptación de eliminación de rutina

Título	Descripción
Nº De prueba	9
Módulo de prueba	Eliminar Rutina
Desarrollo	Front-End
Evento	El sistema debe permitir al usuario eliminar una rutina que ya no necesite tras una confirmación.
Resultado	La rutina es eliminada definitivamente del perfil del usuario.
Estado	Aprobado

Nota. Elaboración Propia

- **RF5 y RF6 Tracking y Retroalimentación en tiempo real:** Mediante la integración de MediaPipe se logró el seguimiento de los 33 landmarks corporales y la emisión de alertas visuales y auditivas.

Tabla 31

Prueba de aceptación visualización de rutina

Título	Descripción
Nº De prueba	10
Módulo de prueba	Ver Rutina (Ejecución y Tracking)
Desarrollo	Front-End / MediaPipe
Evento	El sistema debe activar el tracking asistido y proporcionar feedback visual o auditivo ante errores de postura.
Resultado	El usuario realiza el ejercicio con asistencia en tiempo real y recibe consejos de corrección.
Estado	Aprobado

Nota. Elaboración Propia

5.2.2. Resultados con respecto a la usabilidad

Dada la naturaleza del prototipo, la usabilidad se evaluó con base en el cumplimiento de estándares de diseño y navegación.

- **RNF2 – Visibilidad a Distancia:** Como se menciona en la **Tabla 31**, se verificó que los indicadores y los esqueletos virtuales son visibles a una distancia de dos metros, lo que le permite al deportista realizar el ejercicio con libertad de movimiento.
- **Diseño de Interfaz:** La adopción de Material Design permitió que el usuario navegara por la aplicación de forma intuitiva, completando flujos de trabajo en menos de 3 clics en promedio.

5.2.3. Resultados con respecto a la eficiencia

El rendimiento técnico se midió directamente en los dispositivos **Infinix Hot 20s** y **Tecno Spark 10C** para asegurar la viabilidad del procesamiento local:

- **RNF1 - Latencia de Procesamiento:** El sistema mantuvo una latencia por debajo de los 200 ms planteados tanto en el Tecno como en el Infinix cumpliendo el requisito para la percepción de tiempo real.
- **RNF4 - Estabilidad de la Interfaz:** El uso de las Corrutinas de Kotlin para la gestión de tareas asíncronas permitió que el flujo de video se mantenga estable, evitando el bloqueo de la aplicación mientras los cálculos del MediaPipe se ejecutan en segundo plano.
- **RNF5 Consumo de Hardware y Fluidez:** Durante las pruebas de estrés en los dispositivos Infinix y Tecno, se monitoreó el consumo de recursos mediante las herramientas de perfilado de Android. Se determinó que la aplicación utiliza, en promedio, entre un 50% y 60% de la capacidad del procesador y memoria RAM.

5.2.4. Resultados con respecto a la portabilidad

Se evaluó la capacidad del software para ser desplegado en el entorno objetivo:

- **Sincronización en la nube:** La persistencia de los resultados después de los entrenamientos de completo en tiempos menores a 3 segundos lo que garantiza la disponibilidad inmediata del historial en cualquier dispositivo.
- **RNF3 Sincronización en varios dispositivos:** Todas las pruebas de aceptación Tablas 22 a la 31, fueron ejecutadas con éxito en ambos terminales Android.

5.3 Análisis de Impacto y Validación frente a la Entrevista inicial

Tras la evaluación técnica del prototipo, se realizó un análisis entre los hallazgos de la entrevista y las capacidades finales del sistema. Esto permite validar como la aplicación resuelve de forma directa las necesidades del usuario :

- **Mitigación de riesgos posturales:** Se confirmo que la incertidumbre postural constituía al mayor riesgo de lesión percibido. El sistema mitiga esto mediante el uso de los umbrales parametrizados definidos en la lógica de negocio, lo que transforma la duda del usuario en una validación objetiva procesada por MediaPipe.
- **Reducción de Distracciones:** La implementación del motor de Texto a Voz responde directamente a la preferencia del usuario de no interactuar con la pantalla durante el ejercicio. Los resultados de las pruebas demuestran que las correcciones auditivas permiten concentrarse en la técnica durante el ejercicio.
- **Eficiencia y Privacidad:** El interés del usuario por una asistencia fluida y privada se satisface mediante el procesamiento local de los 33 puntos de referencia. Al ejecutar el tracking en el dispositivo, se garantiza respuestas inmediatas sin depender de retrasos de red.

Conclusiones

Respecto a la selección de tecnologías, se concluye que la librería de Google MediaPipe es la solución más eficiente para el rastreo de movimiento específicamente en dispositivos Android. Aparte de tener una compatibilidad nativa con Android, su capacidad para procesar 33 landmarks corporales, tales como: Partes internas de los brazos, codos, hombros, manos, rodillas y pies, con baja latencia permite un análisis postural preciso sin depender de hardware especializado.

En relación con el diseño de la estructura del sistema, se determinó que la arquitectura basada en el patrón Model-View-ViewModel junto con los servicios de Firebase resultaron fundamentales. Esto permitió separar la lógica de procesamiento del tracking de la gestión de datos en la nube, lo que hace que el progreso del usuario se sincronice desde la nube sin comprometer el rendimiento de la aplicación.

Sobre la implementación de algoritmos y retroalimentación, se concluye que la combinación de algoritmos de detección postural y el texto a voz son los componentes de mayor valor de la aplicación. La detección de errores posturales en tiempo real las cuales son comunicadas de forma auditiva o por pantalla, permite que el usuario corrija su postura sin interrumpir mucho el ejercicio.

En cuanto a la verificación del prototipo, la ejecución de pruebas en escenarios controlados evidenció que el prototipo responde a los requerimientos funcionales definidos en la fase de análisis. El sistema mostró un comportamiento estable en el procesamiento del conteo de

repeticiones y en la emisión de alertas visuales y auditivas durante las pruebas lo que valida la factibilidad técnica de la arquitectura propuesta para el entrenamiento asistido.

Recomendaciones

A estudiantes o futuros investigadores, se les sugiere potenciar el uso de frameworks de visión artificial como MediaPipe en proyectos integradores, debido a su baja latencia y facilidad de implementación para resolver problemáticas sociales de salud y bienestar.

Siguiendo con los estudiantes o futuros investigadores, se les recomienda el uso de trípodes o soportes estables para el dispositivo móvil durante las sesiones, debido a que la precisión de los algoritmos de validación postural depende de un ángulo de visión constante y sin vibraciones de la cámara.

Trabajos Futuros

Ampliación del catálogo de ejercicios: Aunque el sistema actual cubre ejercicios de peso corporal básicos, se identificó la frontera de no incluir disciplinas como yoga o pilates; futuros proyectos podrían integrar configuraciones JSON específicas para estas áreas aprovechando la arquitectura modular actual, también hacer una vista específica del ejercicio más elaborada ya que no hay una bien definida. También, se sugiere implementar un modulo de perfil de salud donde se pueda registrar si el usuario posee alguna lesión temporal o impedimento que le impida hacer un tipo de ejercicio añadiendo así exclusión de ejercicios o avisos antes de realizar ejercicios que coincidan con el perfil de salud del usuario.

Fortalecimiento del módulo administrativo: Se reconoce que las funciones del administrador son actualmente limitadas; se propone desarrollar en futuras versiones una interfaz de gestión más robusta que permita la creación de ejercicios directamente desde la Aplicación sin modificar código fuente.

Modo de funcionamiento Offline: Debido a que la aplicación depende de una conexión constante para consultar rutinas, se plantea como trabajo futuro el desarrollo de una base de datos local (Room) que sincronice con la nube sólo cuando haya red disponible.

Multiplataforma: El enfoque de la aplicación desde un inicio se orientó para Android nativo con el uso de Kotlin como lenguaje de programación, se propone que para futuras versiones la transición hacia frameworks de desarrollo hibrido como Flutter o React Native.

Referencias

- Alex Moran, B. G. (23 de Marzo de 2022). *Muscle Vision: Real Time Keypoint Based Pose Classification of Physical Exercises*. <https://arxiv.org/pdf/2203.12111>
- Ameneyro-Ameneyro, H. M. (2024). Razonamiento inductivo desde diversos paradigmas de investigación. *Revista Ciencia & Sociedad*, 4(3), 267–281.
<https://doi.org/https://cienciaysociedaduatf.com/index.php/ciesocieuatf/article/view/159>
- Android Developer. (23 de Mayo de 2025). *Introducción a Android Studio* .
<https://developer.android.com/studio/intro?hl=es-419>
- Android Developers. (10 de Febrero de 2025). *TextToSpeech*.
<https://developer.android.com/reference/android/speech/tts/TextToSpeech>
- AppDevNext. (2025). *AndroidChart*. GitHub: <https://github.com/AppDevNext/AndroidChart>
- Applandeo. (2023). *Material-Calendar-View*. GitHub: <https://github.com/Aplandeo/Material-Calendar-View>
- BumpTech. (2024). *glide*. Github: <https://gemini.google.com/app/2257a1e5c46984b3>
- Conorea, E. (30 de Marzo de 2022). *¿Qué es Kotlin y para qué sirve?* plainconcepts:
<https://www.plainconcepts.com/es/kotlin-android/>
- Dawn, K. (20 de Diciembre de 2022). *Build an AI Fitness Trainer Using MediaPipe For Squat Analysis*. LearnOpenCV: <https://learnopencv.com/ai-fitness-trainer-using-mediapipe/>
- Encord. (13 de Junio de 2025). *Human Pose Estimation*. Encord:
<http://encord.com/glossary/human-pose-estimation-definition/>
- Escalante, Y. (Agosto de 2011). *Actividad física, ejercicio físico y condición física en el ámbito de la salud pública*. https://scielo.isciii.es/scielo.php?script=sci_arttext&pid=S1135-57272011000400001&lng=es&nrm=iso&tlng=es
- Firebase. (5 de Junio de 2025). *Cloud Firestore* .
<https://firebase.google.com/docs/firestore?hl=es-419>

- Firebase. (9 de Enero de 2026). *Cloud Storage for Firebase*. Firebase:
<https://firebase.google.com/docs/storage?hl=es-419>
- GeeksforGeeks. (20 de Julio de 2024). *Object Tracking in Computer Vision*.
<https://www.geeksforgeeks.org/object-tracking-in-computer-vision/>
- Google. (13 de Enero de 2025). *Guía de detección de puntos de referencia de posiciones*. Google AI for Developers:
https://ai.google.dev/edge/mediapipe/solutions/vision/pose_landmarker?hl=es-419
- Google Developers. (11 de Diciembre de 2025). *Navegación*.
<https://developer.android.com/guide/navigation?hl=es-419>
- Hanaoka, Y. (21 de Junio de 2023). *Fitness App Development with Real-Time Posture Detection using MediaPipe and React*. dev.to: <https://dev.to/yoshan0921/fitness-app-development-with-real-time-posture-detection-using-mediapipe-38do>
- IBM. (27 de Julio de 2021). *What is computer vision?* IBM:
<https://www.ibm.com/think/topics/computer-vision>
- Merino, Á. M. (18 de Septiembre de 2024). *Firestore: Una plataforma integral para el desarrollo de aplicaciones Web y Móviles*. sngular:
<https://www.sngular.com/es/insights/313/firebase>
- Muguira, A. (24 de Mayo de 2023). *¿Qué es una entrevista? Tipos, ejemplos y cómo responder*. QuestionPro: https://www.questionpro.com/blog/es/tecnicas-de-recoleccion-de-datos-entrevista/#Que_es_una_entrevista
- Organizacion Mundial de la Salud. (6 de Junio de 2024). *Actividad física*.
<https://www.who.int/es/news-room/fact-sheets/detail/physical-activity>
- Sarvaiya, D. (3 de Diciembre de 2024). <https://www.intelivita.co.uk/blog/fitness-app>. intelivita:
<https://www.intelivita.co.uk/blog/fitness-app>
- Stewart, L. (18 de Octubre de 2025). *Investigación básica vs. aplicada*. ATLAS.ti:
<https://atlasti.com/es/research-hub/investigacion-basica-vs-aplicada>

Univesidad UNADE. (30 de Enero de 2025). *Método deductivo: qué es y cómo aplicarlo en el aprendizaje*. UNADE: <https://unade.edu.mx/metodo-deductivo-que-es-y-como-aplicarlo-en-el-aprendizaje/>

Yalantis. (2025). *Ucrop*. Github: <https://github.com/Yalantis/uCrop>

Anexos

Anexo A

Preguntas de la encuesta realizada a la persona clave

Preguntas de Entrevista

Dimensión 1: Hábitos y Experiencia

¿Con qué frecuencia realiza ejercicio físico y qué rutinas sigue?

¿Cuál es su mayor desafío al entrenar solo?

¿Cómo contabiliza sus repeticiones?

Dimensión 2: Técnica y Postura

¿Cómo verifica su postura por su cuenta?

¿Ha sentido molestias? ¿En qué articulaciones duda más?

¿Qué importancia le da a un sistema que identifique ángulos articulares?

Dimensión 3: Tecnología

¿Prefiere alertas visuales o auditivas en tiempo real?

¿Qué elementos visuales facilitarían el uso a distancia?

¿Estaría dispuesto a usar una aplicación que automatice todo con la cámara?

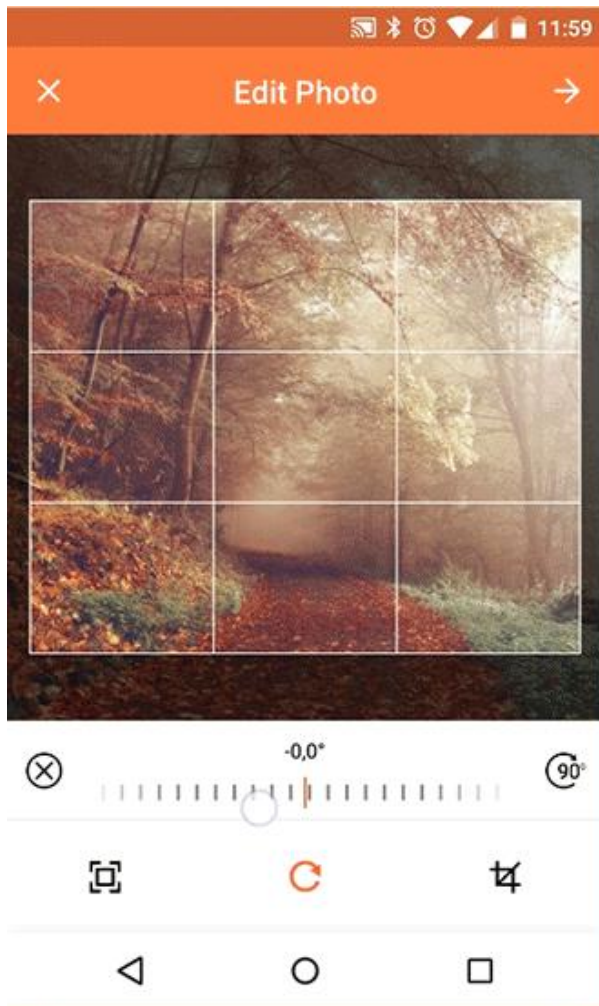
Anexo B

Imagen de la entrevista con la persona clave



Anexo C

Interfaz de edición de imagen de perfil con la librería Ucrop

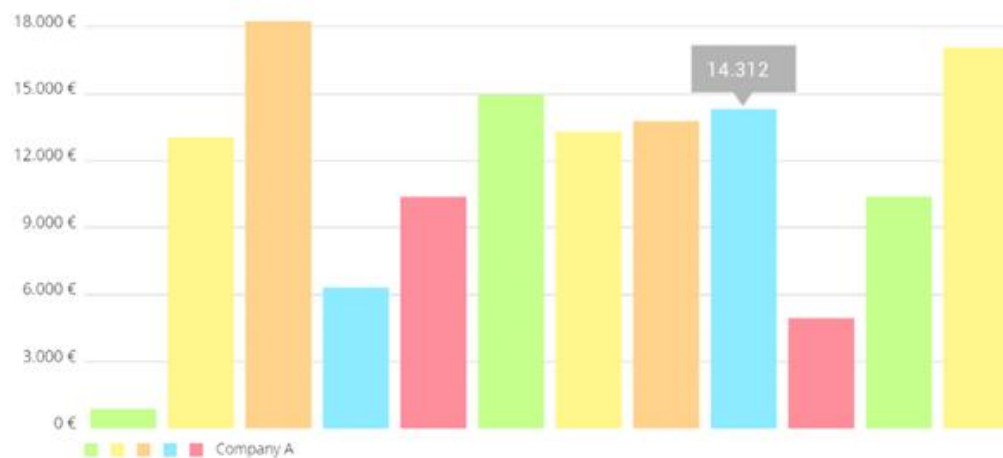


Fuente: Repositorio oficial de Ucrop en GitHub (<https://github.com/Yalantis/uCrop>)

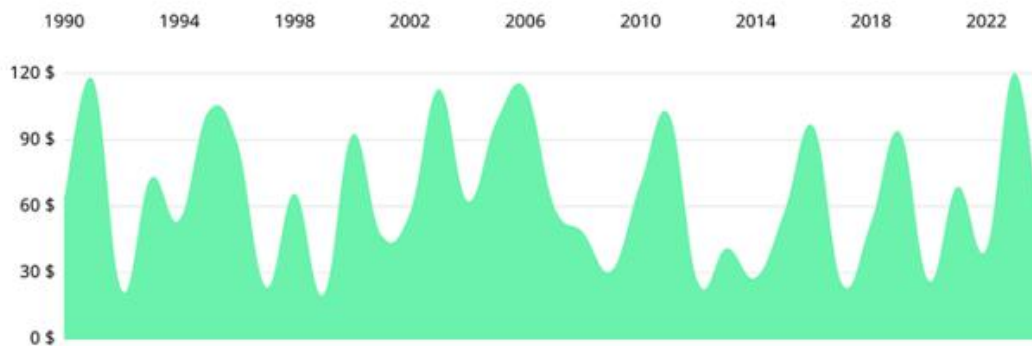
Anexo D

Algunas de las opciones de Gráficos de AndroidChart

BarChart (with legend, simple design)



LineChart (cubic lines)



Fuente: Repositorio Oficial de AndroidChart en GitHub

(<https://github.com/AppDevNext/AndroidChart>)