

UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

Colegio de Ciencias e Ingenierías

**Diseño y desarrollo de prototipo de aplicación móvil en Android Studio para la gestión
Optimizada de un centro vacacional.**

Eduardo Augusto Ríos Salgado

Ingeniería en Ciencias de la Computación

Trabajo de fin de carrera presentado como requisito

para la obtención del título de

Ingeniero en Ciencias de la Computación

Quito, 15 de mayo de 2024

UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

Colegio de Ciencias e Ingenierías

HOJA DE CALIFICACIÓN DE TRABAJO DE FIN DE CARRERA

**Diseño y desarrollo de prototipo de aplicación móvil en Android Studio para la gestión
Optimizada de un centro vacacional.**

Eduardo Augusto Ríos Salgado

Nombre del profesor, Título académico Fausto Pasmay, M.Sc.

Quito, 15 de mayo de 2024

© DERECHOS DE AUTOR

Por medio del presente documento certifico que he leído todas las Políticas y Manuales de la Universidad San Francisco de Quito USFQ, incluyendo la Política de Propiedad Intelectual USFQ, y estoy de acuerdo con su contenido, por lo que los derechos de propiedad intelectual del presente trabajo quedan sujetos a lo dispuesto en esas Políticas.

Asimismo, autorizo a la USFQ para que realice la digitalización y publicación de este trabajo en el repositorio virtual, de conformidad a lo dispuesto en la Ley Orgánica de Educación Superior del Ecuador.

Nombres y apellidos: Eduardo Augusto Ríos Salgado

Código: 212656

Cédula de identidad: 1724427958

Lugar y fecha: Quito, 15 de mayo de 2024

ACLARACIÓN PARA PUBLICACIÓN

Nota: El presente trabajo, en su totalidad o cualquiera de sus partes, no debe ser considerado como una publicación, incluso a pesar de estar disponible sin restricciones a través de un repositorio institucional. Esta declaración se alinea con las prácticas y recomendaciones presentadas por el Committee on Publication Ethics COPE descritas por Barbour et al. (2017)

Discussion document on best practice for issues around theses publishing, disponible en <http://bit.ly/COPETheses>.

UNPUBLISHED DOCUMENT

Note: The following capstone project is available through Universidad San Francisco de Quito USFQ institutional repository. Nonetheless, this project – in whole or in part – should not be considered a publication. This statement follows the recommendations presented by the Committee on Publication Ethics COPE described by Barbour et al. (2017) Discussion document on best practice for issues around theses publishing available on <http://bit.ly/COPETheses>.

RESUMEN

En la era digital actual, la rápida evolución de la tecnología ha transformado significativamente la manera en que interactuamos con los servicios, ofreciendo oportunidades sin precedentes para mejorar la eficiencia y la satisfacción del cliente en diversos. En este contexto, las aplicaciones móviles emergen como herramientas cruciales para facilitar la gestión y mejora de la calidad en el servicio de centros vacacionales, beneficiando tanto a clientes como a administradores. Este proyecto de tesis se centra en el desarrollo de un prototipo de aplicación móvil diseñado específicamente para un centro vacacional, con el objetivo de optimizar la gestión de servicios y mejorar la comunicación entre los clientes y el personal administrativo.

La aplicación móvil desarrollada en este proyecto utiliza Android Studio y Kotlin como entorno de desarrollo y lenguaje de programación, respectivamente, aprovechando las capacidades de Firebase para la gestión de bases de datos en tiempo real y autenticación de usuarios. Esta solución tecnológica busca abordar desafíos comunes en la gestión de centros vacacionales, tales como la reserva de actividades, la consulta de horarios de eventos y servicios, y la facilitación de un canal de comunicación directo y eficiente entre los clientes y la administración.

A través de una interfaz amigable e intuitiva, la aplicación permite a los usuarios registrarse, iniciar sesión, y acceder a una variedad de funciones adaptadas a sus necesidades, ya sean clientes buscando disfrutar de su estancia sin contratiempos o administradores esforzándose por mantener una operación fluida y satisfactoria. Para los clientes, esto se traduce en la capacidad de inscribirse fácilmente en actividades, visualizar horarios de manera clara, y

comunicarse con la administración para resolver dudas o solicitar servicios especiales. Por otro lado, el personal administrativo se beneficia de herramientas que facilitan la organización de actividades, la gestión de reservas, y el mantenimiento de una comunicación constante con los clientes, todo ello contribuyendo a una gestión más eficiente y una mejora en la calidad del servicio ofrecido.

Este proyecto no solo demuestra la viabilidad de integrar tecnologías móviles en la gestión de centros vacacionales, sino que también destaca el potencial de estas herramientas para enriquecer la experiencia del cliente y facilitar las operaciones administrativas. Al finalizar, se ofrece una reflexión sobre los logros obtenidos, los desafíos enfrentados, y se esbozan posibles direcciones para futuras investigaciones y desarrollos en este ámbito.

ABSTRACT

The rapid evolution of technology in the digital era has significantly altered the way we interact with services, providing unprecedented opportunities to enhance efficiency and customer satisfaction across various sectors, including tourism and recreation. Mobile applications stand out as key tools for facilitating service management and improving quality in holiday resorts, benefiting both clients and administrators. This thesis project focuses on developing a mobile application prototype specifically designed for a holiday resort, aiming to optimize service management and enhance communication between guests and administrative staff.

Developed using Android Studio and Kotlin, with Firebase for real-time database management and user authentication, this technological solution addresses common challenges in holiday resort management, such as activity booking, event and service scheduling, and establishing a direct, efficient communication channel between guests and the administration. Through a user-friendly interface, the application offers functionalities tailored to the needs of both guests, enhancing their stay, and administrators, streamlining operations. The project illustrates the feasibility of integrating mobile technologies in holiday resort management and highlights their potential to improve customer experience and administrative processes. It concludes with a reflection on achievements, challenges, and outlines future research and development directions in this field.

TABLA DE CONTENIDOS

1. INTRODUCCIÓN.....	9
2. ESTADO DEL ARTE	11
2.1. Tecnologías de desarrollo de aplicaciones móviles	11
2.2. Revisión de tecnologías aplicadas	12
2.3. Necesidades específicas de centros vacacionales y cómo abordarlas.....	16
2.4. Importancia de la UX/UI en aplicaciones recreacionales	17
2.5. Impacto de la tecnología móvil.....	19
3. DESARROLLO DEL TEMA.....	20
3.1. Identificación del problema	20
3.2. Objetivos del Proyecto.....	22
4. MARCO TEÓRICO	24
4.1. Tecnologías de desarrollo	24
4.2. Diseño de la aplicación	27
4.3. Funcionamiento del sistema.....	29
4.4. Diagrama de arquitectura de la aplicación.....	32
5. IMPLEMENTACIÓN.....	34
5.1. Desarrollo de funcionalidades clave	34
5.2. Integración con Firebase:.....	38
5.3. Creación y manejo de clases necesarias.....	41
6. PRUEBAS Y VALIDACIÓN	42
6.1. Metodología de Pruebas: Cómo se planificaron y ejecutaron las pruebas para asegurar la funcionalidad y usabilidad de la aplicación.	42
6.2. Resultados de las Pruebas: Análisis de los resultados obtenidos y cómo estos impactan en la calidad final del proyecto.	44
7. CONCLUSIONES Y TRABAJOS FUTUROS.....	45
7.1. Conclusiones	45
7.2. Evaluación del Proyecto: Reflexión sobre el cumplimiento de los objetivos, problemas enfrentados y soluciones implementadas.	46
7.3. Recomendaciones para futuras mejoras: Identificación de áreas de oportunidad y sugerencias para el desarrollo futuro de la aplicación.....	47
8. REFERENCIAS	48

1. INTRODUCCIÓN

En la era digital actual, las aplicaciones móviles se han convertido en herramientas esenciales para facilitar y mejorar diversos aspectos de nuestra vida cotidiana. Desde aplicaciones de comunicación, redes sociales, aplicaciones que aportan soluciones para la gestión de tareas, productividad, entretenimiento, etc. La tecnología móvil ofrece un sinfín de posibilidades para satisfacer las necesidades de los usuarios. En este contexto, el desarrollo de aplicaciones móviles que sean eficientes, intuitivas y seguras representa un desafío constante para los desarrolladores. Este proyecto se centra en el desarrollo de un prototipo de aplicación móvil enfocado en permitir tanto a los clientes (padres de familia) y administración gestionar eficientemente las inscripciones, consultar horarios, y comunicarse de manera efectiva, todo esto a través de una interfaz intuitiva y accesible, abordando tanto los retos técnicos como las expectativas de los usuarios finales.

La presente tesis se centra en el diseño y desarrollo de un prototipo de aplicación móvil, concebido como una herramienta que busca ofrecer una solución práctica y funcional a un problema de gestión que es un desafío tanto para padres de familia, clientes del centro vacacional y el personal perteneciente del mismo centro. Este proyecto se desarrolla utilizando Android Studio, seleccionado por su amplia aceptación como entorno de desarrollo integrado (IDE) para la creación de aplicaciones en el ecosistema Android y Kotlin, elegido por sus ventajas en términos de seguridad, claridad en el código y compatibilidad con Android. Además, se integra Firebase como sistema de gestión de base de datos, aprovechando su escalabilidad, flexibilidad y

eficiencia en el manejo de datos en tiempo real, así como su robusta infraestructura de autenticación de usuarios.

La gestión eficiente de las actividades del centro vacacional representa un desafío significativo, tanto para los padres de familia como para el personal administrativo. La comunicación ineficaz, la gestión inadecuada de inscripciones, y la dificultad para acceder a información actualizada sobre horarios y disponibilidad de actividades son problemas comunes que afectan la experiencia educativa de los estudiantes. Este proyecto de tesis aborda estos desafíos mediante el desarrollo de un prototipo de aplicación móvil, diseñado específicamente para facilitar la gestión y comunicación de actividades extracurriculares en un entorno educativo.

La aplicación propuesta se caracteriza por su capacidad de identificar el tipo de usuario (cliente o administración) al momento del inicio de sesión, direccionándolos hacia interfaces diseñadas a medida para satisfacer sus necesidades específicas. Para los padres, esto se traduce en un acceso rápido a inscripciones, información sobre clases disponibles según el club al que pertenezcan sus hijos, consulta de horarios, y un canal directo de comunicación con la administración. Para el personal administrativo, la aplicación facilita la visualización y gestión de horarios, la inscripción de estudiantes en actividades, y mejora la comunicación con los padres.

Además, se contempla la incorporación de una funcionalidad de "super administrador", que permitirá una gestión más flexible de los permisos de usuario y proporcionará herramientas avanzadas para el manejo de la información dentro de la aplicación. Este enfoque no solo busca

mejorar la eficiencia operativa y la satisfacción de los usuarios, sino también promover una mayor integración de las tecnologías de la información en los procesos educativos.

A través de este proyecto, se espera demostrar cómo una aplicación móvil bien diseñada y correctamente implementada puede ser una solución efectiva a los problemas de gestión en el contexto de centros vacacionales. El desarrollo de este prototipo pretende no solo solucionar problemas específicos identificados, sino también explorar las posibilidades que las tecnologías móviles modernas ofrecen para la mejora continua de los procesos educativos y administrativos.

2. ESTADO DEL ARTE

2.1. Tecnologías de desarrollo de aplicaciones móviles

2.1.1. Historia y evolución del desarrollo de aplicaciones móviles enfocadas en educación y recreación

La historia y el desarrollo de aplicaciones móviles, particularmente aquellas enfocadas en la administración, la educación y el manejo de datos, reflejan una evolución significativa en cómo las tecnologías digitales se han integrado en nuestros sistemas educativos y administrativos para ofrecer soluciones más eficientes, accesibles y personalizadas. Desde sus inicios, las aplicaciones móviles han transitado de ser simples herramientas de comunicación y entretenimiento a convertirse en plataformas complejas capaces de gestionar datos importantes, facilitar la administración de instituciones y mejorar la experiencia educativa tanto para educadores como para estudiantes (Phongtraychack y Dolgaya, 2018).

En los primeros días de la tecnología móvil, las aplicaciones se centraban en funciones básicas como el correo electrónico, calendarios y juegos. Sin embargo, con el avance tecnológico

de los smartphones y las tablets, se abrió un nuevo mundo de posibilidades. Los desarrolladores comenzaron a explorar cómo estas tecnologías podrían aplicarse en el ámbito educativo y administrativo, dando lugar a las primeras aplicaciones diseñadas para mejorar la gestión de aulas, la comunicación entre profesores y padres, y la administración eficiente de recursos y datos institucionales.

A medida que la tecnología avanzaba, también lo hacía la complejidad y la funcionalidad de estas aplicaciones. Se introdujeron sistemas de gestión de aprendizaje (LMS, por sus siglas en inglés) que permitían a los educadores crear contenido didáctico digital, asignar tareas, calificar y seguir el progreso de sus estudiantes todo desde sus dispositivos móviles. Paralelamente, en el ámbito administrativo, surgieron aplicaciones capaces de manejar extensos conjuntos de datos, facilitar la gestión de recursos humanos, la planificación financiera y la toma de decisiones estratégicas, todo ello accesible desde cualquier lugar y en cualquier momento, lo que representó una revolución en la gestión institucional (Phongtraychack y Dolgaya, 2018).

En la actualidad, el mundo del desarrollo de aplicaciones móviles para la educación y la administración está experimentando un crecimiento sin precedentes. La constante evolución de las tecnologías móviles ha abierto nuevas posibilidades para la innovación en estos campos, promoviendo soluciones cada vez más sofisticadas y efectivas. La demanda de aplicaciones que ofrezcan soluciones integrales para la gestión educativa y administrativa sigue creciendo, impulsada por la necesidad de sistemas que no solo sean eficientes y accesibles, sino también capaces de adaptarse a las cambiantes dinámicas del aprendizaje y la gestión institucional.

2.2. Revisión de tecnologías aplicadas

2.2.1. Características y ventajas de Android Studio como entorno de desarrollo integrado (IDE)

Android Studio es el entorno de desarrollo integrado (IDE) oficial para el desarrollo de aplicaciones Android (Hagos, 2018). Este se apoya de la plataforma IntelliJ IDEA, brindando una experiencia de desarrollo excepcional con capacidades superiores en edición de código, optimización de rendimiento y funcionalidades de depuración.

Para apoyar el desarrollo de aplicaciones dentro del sistema operativo Android, Android Studio utiliza un sistema de construcción basado en Gradle, Emulador Android, plantillas de código e integración con GitHub (Hagos, 2018). Android Studio ofrece un entorno unificado que agrupa todas las herramientas esenciales para el desarrollo, prueba y depuración de aplicaciones Android, centralizando el proceso en una única plataforma. Se mantiene constantemente actualizado con las más recientes tecnologías y las mejores prácticas recomendadas, asegurando que los desarrolladores tengan acceso a las últimas innovaciones en el campo.

2.2.2. Características y ventajas de Kotlin como lenguaje de programación

Kotlin es un lenguaje de programación contemporáneo y adaptable que destaca por su interoperabilidad fluida con Java, permitiendo a los desarrolladores integrar ambos lenguajes fácilmente, lo que favorece la transición de Java a Kotlin (Moskala y Wojda, 2017).

Se distingue por su sistema de tipos que minimiza los errores comunes como las null pointer exceptions al diferenciar claramente entre tipos que pueden ser nulos y los que no (Ardito et al., 2020). Su capacidad para inferir tipos reduce la necesidad de declaraciones explícitas, haciendo el código menos verboso (Moskala y Wojda, 2017). También mejora la extensibilidad

mediante funciones de extensión sin necesidad de herencia, ofrecen una solución robusta para la programación asíncrona, permitiendo operaciones no bloqueantes (Moskala y Wojda, 2017). Además, Kotlin apoya el desarrollo multiplataforma, posibilitando la reutilización de código entre diferentes plataformas mientras se personaliza la interfaz de usuario según sea necesario.

2.2.3. Uso de Android Studio y Kotlin

La combinación de Android Studio con Kotlin eleva la eficiencia de los desarrolladores, al minimizar el código innecesario y reforzar la prevención de errores comunes, como los null pointer exceptions, mediante sus sólidas funcionalidades de seguridad (Hagos, 2018). Esta alianza facilita la adopción de las metodologías de programación más recientes, tanto funcionales como orientadas a objetos, promoviendo así un enfoque de desarrollo de aplicaciones Android más contemporáneo y efectivo (Moskala y Wojda, 2017).

En el 2017 Google anunció a Kotlin como un lenguaje oficialmente soportado para el desarrollo de aplicaciones Android (Panchal y Patel, 2017). Gracias al respaldo constante de Google, Android Studio y Kotlin se actualizan de manera regular, incorporando nuevas funciones, optimizando el rendimiento y corrigiendo fallos para garantizar que los desarrolladores cuenten con herramientas de punta y actualizadas para la realización de sus proyectos (Hagos, 2018). Este soporte ininterrumpido asegura una base tecnológica avanzada y confiable para el desarrollo de aplicaciones.

2.2.4. Características y ventajas de Firebase como back-end

El desarrollo de aplicaciones móviles atrae tanto a empresas como a desarrolladores autónomos, utilizando diversas herramientas y marcos para crear aplicaciones para distintas

plataformas. Para operar adecuadamente con la interfaz de usuario frontal, las aplicaciones móviles requieren de servicios y código de back-end, así como de servidores específicos (Cheng, 2018).

Firestore es una plataforma integral de Google que se destaca por simplificar este proceso, al ofrecer una base de datos en formato JSON que facilita las operaciones de datos sin requerir consultas complejas (Khawas y Shah, 2018). Además, Firestore complementa, en lugar de reemplazar, las API existentes, brindando servicios claves como bases de datos, autenticación segura y sistemas de mensajería. Al integrar las aplicaciones móviles con Firestore, ya no es necesario escribir código de back-end ni gestionar la infraestructura, simplificando así el proceso de desarrollo (Cheng, 2018). Liberando a los desarrolladores de tener que preocuparse por la gestión del back-end, pueden concentrarse en características distintivas de sus aplicaciones (Moroney, 2017).

Firestore ofrece ocho tecnologías diseñadas para optimizar el proceso de desarrollo de aplicaciones. En este proyecto se hace uso de las siguientes:

2.2.4.1. Firestore Authentication.

Para ofrecer datos únicos a cada usuario, es esencial un método de inicio de sesión seguro. Esta plataforma cuenta con un sistema para gestionar usuarios, brindando la opción de autenticación mediante correo electrónico y contraseña, los cuales se almacenan en Firestore (Khawas y Shah, 2018).

2.2.4.2. Firestore Cloud Storage.

Además de almacenar datos, es común que las aplicaciones necesiten guardar archivos como fotos o videos. El almacenamiento en la Nube para Firebase simplifica esto al ofrecer una API sencilla respaldada por Google Cloud Storage (Khawas y Shah, 2018). Los desarrolladores pueden utilizarlo para almacenar imágenes, audio, video o cualquier otro contenido generado por los usuarios.

2.2.4.3. Real-time Database.

Firestore proporciona una API que permite a los desarrolladores sincronizar y almacenar datos de aplicaciones en la nube, de acuerdo con Khawas y Shah (2018). Esta base de datos NoSQL, alojada en la nube, facilita la sincronización entre dispositivos sin necesidad de conexión a Internet mediante un caché local. Difiere de las bases de datos SQL tradicionales por su arquitectura basada en eventos y porque la programación se gestiona íntegramente en el cliente, sin requerir código en el servidor (Moroney, 2017).

2.3. Necesidades específicas de centros vacacionales y cómo abordarlas

Las necesidades específicas de los centros vacacionales, como la organización de actividades, gestión de reservas y comunicación efectiva entre el personal y los visitantes, pueden abordarse mediante una aplicación móvil diseñada para optimizar estos procesos.

La aplicación puede facilitar la personalización de la experiencia del visitante, permitiendo reservas y personalización de actividades a través del dispositivo móvil, así como proporcionar un canal directo de comunicación para resolver dudas o recibir asistencia, mejorando así la satisfacción general del cliente.

2.3.1. Casos de estudio de aplicaciones exitosas en contextos de centros vacacionales

Si bien la información e investigación en torno a aplicaciones móviles en contextos de centros vacacionales es limitada, se ha encontrado aplicaciones como myCAMPapp que permite conectar fácilmente con padres, compartir actualizaciones importantes y mantener a todos al tanto sobre las actividades del campamento, horarios y documentos (myCAMPapp, s.f.).

Por otro lado, Bunk1 es una empresa que colabora con campamentos de verano en Estados Unidos y Canadá, ofreciendo a las familias el poder estar conectados con sus hijos. Cuenta con una página web y aplicaciones individuales para usuarios y personal administrativo. Con Bunk1, las familias pueden acceder a galerías de fotos seguras en línea con etiquetas de fotos de sus hijos, enviar notas impresas, leer boletines diarios y ver videos del campamento (Bunk1, s.f.).

Se encuentra también la aplicación ClassDojo (ClassDojo, s.f.) que consiste en una comunidad escolar virtual, ofreciendo así un espacio para profesores y familias en el que mantengan contacto, comunicación y se compartan experiencias de aprendizaje. Esta plataforma cuenta con contacto al instante entre profesores y familiar, el poder compartir fotos, videos y actualizaciones, así como crear y visualizar el calendario y los eventos.

2.4. Importancia de la UX/UI en aplicaciones recreacionales

En el desarrollo de software, la interfaz de usuario (UI) comprende los elementos visuales e interactivos, como botones, menús y pantallas (Miya y Govender, 2022). La experiencia de usuario (UX), en cambio, engloba la forma en que las personas interactúan con el software,

incluyendo aspectos como la usabilidad, la accesibilidad y el grado de implicación del usuario (Miya y Govender, 2022).

La UX/UI en aplicaciones educativas es crucial porque determina la eficacia con la que los usuarios y administradores interactúan con la tecnología. Una buena experiencia de usuario (UX) asegura que la aplicación sea intuitiva y fácil de usar, lo cual es esencial para mantener la motivación y el compromiso por parte del usuario. Por otro lado, una interfaz de usuario (UI) bien diseñada mejora la accesibilidad y la claridad, lo que facilita el proceso de organización y comunicación. En conjunto, una UX/UI cuidadosamente planificada puede significativamente enriquecer la experiencia del usuario (Miya y Govender, 2022).

2.4.1. *Diseño XML*

El XML, que es un lenguaje de marcado extensible, utiliza etiquetas, elementos y atributos para estructurar texto. En el desarrollo de aplicaciones para Android, este lenguaje se emplea específicamente para configurar y definir la interfaz de usuario (UI) de las apps (Android Developers, s.f.). La interfaz de usuario de una aplicación para Android se organiza en una estructura jerárquica que contiene varios componentes, conocidos como widgets, y los diseños visuales de esos componentes en la pantalla (Android Developers, s.f.).

En Android, la personalización de la apariencia de una aplicación se realiza a través del diseño XML, donde se especifican estilos, temas y otros recursos como colores, dimensiones y textos. Esto permite ajustar el diseño tanto en elementos específicos como en toda la aplicación, ayudando a manejar la diversidad y adaptación a diferentes idiomas y formatos (Android Developers, s.f.).

2.5. Impacto de la tecnología móvil

2.5.1. Cómo las aplicaciones móviles transforman la gestión y comunicación

Las aplicaciones móviles han cambiado fundamentalmente cómo las personas y organizaciones gestionan sus actividades y se comunican. Ofreciendo acceso a información y herramientas de forma instantánea, estas aplicaciones facilitan la realización de tareas y el mantenimiento del contacto sin limitaciones geográficas (Islam, Islam, y Mazumber, 2010). Favorecen una colaboración eficaz mediante herramientas como mensajería instantánea y videoconferencias. Además, permiten una gestión de tareas y proyectos más organizada y eficiente, optimizando así la productividad y el éxito en las labores cotidianas (Islam et al., 2010).

2.5.2. Beneficios de integrar tecnologías móviles en procesos recreacionales y administrativos

Actualmente, cerca del 80% de las personas acceden a internet a través de dispositivos móviles, y la mayoría prefiere utilizar aplicaciones móviles por su facilidad de uso y capacidad para ejecutar tareas de manera instantánea (Phongtraychack y Dolgaya, 2018).

La integración de tecnologías móviles en centros vacacionales enriquece la experiencia de administradores y clientes. Las aplicaciones ofrecen información valiosa sobre servicios y facilitan reservas en línea, mejorando la planificación de visitas. Actúan como canales de comunicación directa para notificaciones instantáneas y personalización de servicios basada en preferencias y demografía, incrementando la satisfacción y fidelidad. Incluir funciones de

seguridad y recopilar datos para análisis optimiza la operatividad y la oferta de actividades, elevando la experiencia del visitante.

3. DESARROLLO DEL TEMA

3.1. Identificación del problema

3.1.1. *Necesidad de Mejora. Descripción detallada de los problemas actuales.*

3.1.1.1. Gestión ineficiente de las inscripciones: uno de los principales problemas radica en la pobre gestión de las personas inscritas en actividades extracurriculares. La falta de un sistema eficiente que permita una inscripción fluida y el seguimiento en tiempo real de las plazas disponibles conduce a confusiones, sobre inscripciones o incluso la pérdida de oportunidades para los interesados. Esta ineficiencia no solo frustra a los padres que buscan inscribir a sus hijos en actividades de su interés, sino que también sobrecarga al personal administrativo con tareas manuales y repetitivas que podrían automatizarse.

3.1.1.2. Necesidad de mejorar la atención: la calidad de la atención ofrecida a padres e hijos es fundamental para la reputación y el éxito de cualquier centro vacacional. La eficiencia operativa, la capacidad de responder rápidamente a las consultas y la atención personalizada son aspectos que actualmente necesitan ser mejorados. La implementación de soluciones tecnológicas podría desempeñar un papel crucial en la mejora de estos aspectos, permitiendo al personal dedicar más tiempo a proporcionar una atención de calidad en lugar de gestionar papeleo o resolver problemas administrativos.

3.1.1.3. Eficiencia operativa: la eficiencia operativa en la gestión de actividades extracurriculares es un área que requiere innovación. Los procesos manuales y la dependencia de sistemas obsoletos para la planificación y gestión de actividades limitan la capacidad del centro para adaptarse a las demandas cambiantes y aumentar la satisfacción del cliente. La adopción de herramientas tecnológicas modernas ofrece una oportunidad para optimizar estos procesos, reduciendo errores, mejorando la planificación y ejecución de actividades, y permitiendo una gestión más dinámica y adaptable a las necesidades de los usuarios.

3.1.2. *Impacto en usuarios. Análisis del impacto de estos problemas en la experiencia de padres y personal administrativo.*

3.1.2.1. Padres: para los padres, la gestión ineficiente de inscripciones y la falta de comunicación efectiva pueden generar frustración y descontento. La dificultad para acceder a información actualizada sobre las actividades disponibles, horarios y cupos conlleva a una experiencia de usuario negativa, afectando su percepción de la calidad del servicio ofrecido por el centro vacacional. Además, la falta de un sistema de comunicación directo y eficiente impide que los padres se sientan parte de la comunidad educativa, lo que puede resultar en una menor participación en las actividades propuestas y, eventualmente, en la búsqueda de alternativas en otros centros.

3.1.2.2. Personal administrativo: el personal administrativo enfrenta un aumento significativo en la carga de trabajo debido a la gestión manual de inscripciones

y comunicaciones. Esta situación no solo reduce la eficiencia operativa, sino que también aumenta el riesgo de errores, lo que puede llevar a situaciones conflictivas con padres e hijos. La falta de herramientas adecuadas para la gestión y comunicación eficiente limita la capacidad del personal para ofrecer un servicio de calidad, afectando su satisfacción laboral y potencialmente incrementando la tasa de rotación del personal.

3.2. Objetivos del Proyecto

3.2.1. *Objetivo general:* Desarrollar una aplicación móvil que mejore la gestión de actividades extracurriculares, facilitando la comunicación y el proceso de inscripción.

3.2.2. *Objetivos específicos.*

3.2.2.1. Diseñar una interfaz de usuario intuitiva y accesible. Crear interfaces claras y amigables para todos los roles de usuario, garantizando una navegación fluida y una experiencia de usuario positiva. Esto incluye pantallas específicas para el registro de usuarios, la gestión de hijos, la visualización e inscripción en clases, y las funcionalidades avanzadas para administradores y super administradores.

3.2.2.2. Implementar un sistema de gestión de usuarios con roles diferenciados. Desarrollar un esquema de autenticación y autorización que permita diferenciar entre clientes (padres), administradores y super

administradores, cada uno con accesos y permisos específicos dentro de la aplicación.

3.2.2.3. Desarrollar un sistema de inscripciones y consulta de horarios. Crear

un mecanismo eficiente para que los usuarios puedan inscribir a sus hijos en clases adecuadas según su edad, visualizar horarios de clases y gestionar inscripciones existentes, facilitando la organización de su participación en actividades.

3.2.2.4. Asegurar la integración efectiva con Firebase para la gestión de datos

y autenticación. Utilizar Firebase Firestore para almacenar y gestionar dinámicamente los datos de usuarios, clases e inscripciones, Firebase Authentication para un sistema seguro de registro e inicio de sesión, y Firebase Storage para el manejo de archivos multimedia.

3.2.2.5. Implementar un rol de super administrador para el manejo avanzado

de la información y configuración de la base de datos: Permitir el manejo avanzado de la información y la configuración de la base de datos, otorgando a este rol funcionalidades exclusivas para gestionar integralmente la aplicación. El super administrador podrá asignar roles de administrador a otros usuarios, facilitando la organización del sistema y permitiendo la delegación de responsabilidades. Además, tendrá la capacidad de crear, visualizar y eliminar clases, brindando una gestión completa de las actividades ofrecidas. Gestión de imágenes, tanto generales como personales de los administradores. Este rol

permitirá al superadministrador actualizar cualquier contenido relevante de la aplicación, asegurando que la información sea precisa y esté actualizada.

3.2.2.6. Facilitar la gestión y actualización de contenido en la aplicación sin necesidad de actualizaciones de software. Realizar cambios en el contenido de la aplicación, como la oferta e información de las clases, información relevante para cada usuario, directamente a través de la interfaz de la aplicación, sin requerir actualizaciones de software por parte de los clientes.

4. MARCO TEÓRICO

4.1. Tecnologías de desarrollo

4.1.1. *Android Studio: Ventajas y por qué es adecuado para el desarrollo de la aplicación.*

4.1.1.1. Integración completa: Android Studio proporciona un entorno de desarrollo unificado que incluye todas las herramientas necesarias para desarrollar, probar y depurar aplicaciones Android, lo que mejora significativamente la eficiencia del proceso de desarrollo.

4.1.1.2. Compatibilidad con Kotlin: dado que Android Studio soporta plenamente Kotlin, permite a los desarrolladores aprovechar todas las características y ventajas de este lenguaje de programación moderno.

4.1.1.3. Herramientas de prueba integradas: ofrece un conjunto de herramientas para realizar pruebas unitarias y de interfaz de usuario dentro del mismo

entorno, facilitando la identificación y corrección de errores en etapas tempranas del desarrollo.

4.1.1.4. Emulador rápido y versátil: el emulador incluido en Android Studio es rápido y ofrece configuraciones personalizables, permitiendo probar aplicaciones en diferentes versiones de Android y tamaños de pantalla sin necesidad de dispositivos físicos.

4.1.2. *Kotlin: Beneficios de usar Kotlin en el desarrollo de aplicaciones Android.*

4.1.2.1. Sintaxis concisa: Kotlin ofrece una sintaxis más concisa y expresiva en comparación con Java, lo que reduce la cantidad de código y facilita la lectura y mantenimiento del código.

4.1.2.2. Compatibilidad: es totalmente compatible con Java, permitiendo utilizar todas las bibliotecas existentes de Java en proyectos Kotlin, además de integrarse sin problemas en el ecosistema Android.

4.1.3. *Firestore como solución integral.*

4.1.3.1. Firestore se presenta como una plataforma integral: para el desarrollo de aplicaciones móviles, ofreciendo un conjunto de herramientas y servicios que simplifican tareas cruciales como la gestión de datos, autenticación de usuarios, y análisis de aplicaciones, entre otros. Su enfoque en la facilidad de uso y la escalabilidad lo hace particularmente atractivo para desarrolladores que buscan maximizar la eficiencia y minimizar los costos operativos.

4.1.3.2. Gestión de bases de datos en tiempo real: Firebase Firestore proporciona una solución poderosa para la gestión de bases de datos en tiempo real, permitiendo a las aplicaciones móviles almacenar, sincronizar y consultar datos de manera eficiente y sin esfuerzo. Los desarrolladores pueden estructurar y acceder a los datos a través de una API intuitiva, lo que facilita la creación de aplicaciones ricas y reactivas que responden en tiempo real a los cambios en los datos. La capacidad de Firestore para actualizarse automáticamente con los últimos datos sin requerir recargas manuales mejora significativamente la experiencia del usuario final.

4.1.3.3. Autenticación de usuarios: Firebase Authentication se destaca por ofrecer un servicio de autenticación completo y fácil de implementar, soportando métodos con correo electrónico y contraseña, así como autenticación de terceros a través de proveedores como Google, Facebook, y Twitter. Esta diversidad en opciones de permite a los desarrolladores ofrecer a los usuarios múltiples maneras de registrarse e iniciar sesión en la aplicación, mejorando la accesibilidad y la flexibilidad del sistema de autenticación. Además, Firebase Authentication integra características de seguridad avanzadas, como la verificación de correo electrónico y el restablecimiento de contraseñas, asegurando que el proceso sea tanto seguro como amigable para el usuario.

4.1.3.4. Facilidad de uso y escalabilidad: una de las ventajas más significativas de Firebase es su facilidad de integración y configuración en proyectos de aplicaciones móviles. Los desarrolladores pueden incorporar rápidamente los

servicios de Firebase en sus aplicaciones con unas pocas líneas de código, lo que permite centrarse más en el desarrollo de características específicas de la aplicación en lugar de en la infraestructura subyacente. Además, Firebase ofrece una escalabilidad automática, adaptándose a las demandas crecientes de los usuarios y datos sin necesidad de intervención manual. Esto asegura que la aplicación pueda crecer fluidamente, manejando picos de tráfico y aumentos en la base de usuarios sin comprometer el rendimiento o la disponibilidad.

4.2. Diseño de la aplicación

4.2.1. *Arquitectura de la aplicación:* la aplicación se ha estructurado utilizando una arquitectura modular y basada en componentes que facilita el mantenimiento y la escalabilidad. Cada módulo se enfoca en una funcionalidad clave, como la autenticación de usuarios, la gestión de hijos, y la administración de clases, permitiendo una separación clara de responsabilidades. Para el manejo de datos y la lógica de negocio, se ha optado por patrones como Modelo-Vista-Controlador (MVC) para mejorar la separación entre la interfaz de usuario y la lógica de negocio, lo que resulta en un código más limpio y fácil de testear. La integración con Firebase se maneja de manera centralizada, facilitando el acceso y la manipulación de datos en tiempo real y la autenticación de usuarios de manera segura y eficiente.

4.2.2. *Interfaz de usuario:* diseño de Pantallas para Diferentes Usuarios: La aplicación presenta diseños de interfaz adaptados a los roles específicos de los usuarios:

- 4.2.2.1. Padres:** la interfaz para padres está diseñada para ser directa y fácil de usar, con flujos de trabajo simplificados para añadir y gestionar la información de los hijos, inscribirlos en clases y comunicarse con el personal administrativo. Se ha puesto especial atención en hacer que el proceso de inscripción sea lo más intuitivo posible, incluyendo filtros automáticos que presentan solo las clases relevantes basadas en la edad de cada hijo.
- 4.2.2.2. Administración y super administrador:** interfaces avanzadas que facilitan la gestión integral de la aplicación, desde la asignación de roles hasta la creación y eliminación de clases, con herramientas diseñadas para simplificar estas tareas administrativas.
- 4.2.3. Experiencia de usuario y usabilidad:** la aplicación ha sido diseñada con un fuerte enfoque en la usabilidad, asegurando que todas las interacciones sean intuitivas y que la información importante sea fácilmente accesible. El diseño responsivo garantiza una experiencia coherente en diferentes dispositivos y tamaños de pantalla.
- 4.2.4. Gestión de usuarios y roles:** la aplicación implementa un sistema de gestión de roles detallado, utilizando Firebase para asignar roles específicos (padre, administrador, super administrador) y controlar el acceso a las funcionalidades correspondientes. Esto permite una experiencia personalizada para cada usuario, asegurando que tengan acceso solo a las herramientas y la información que necesitan.

4.2.5. *Filtrado automático por edad:* implementación de un sistema que

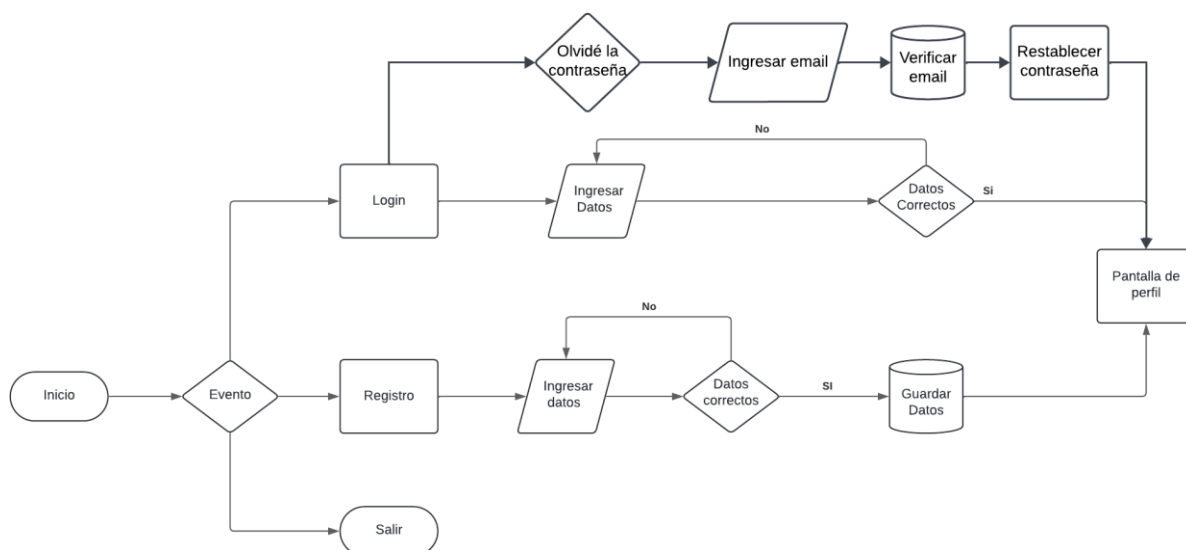
automáticamente filtra y muestra clases que son adecuadas para la edad de cada hijo registrado por los padres, simplificando el proceso de inscripción y asegurando que las actividades sean apropiadas y seguras para los niños.

4.2.6. *Gestión de cupos y horarios:* Desarrollo de una funcionalidad que permite a los usuarios ver la disponibilidad de cupos en tiempo real para cada clase, así como los horarios de inicio y fin, facilitando a los padres la planificación de actividades extracurriculares de sus hijos sin conflictos de horario.

4.3. Funcionamiento del sistema

Figura 1

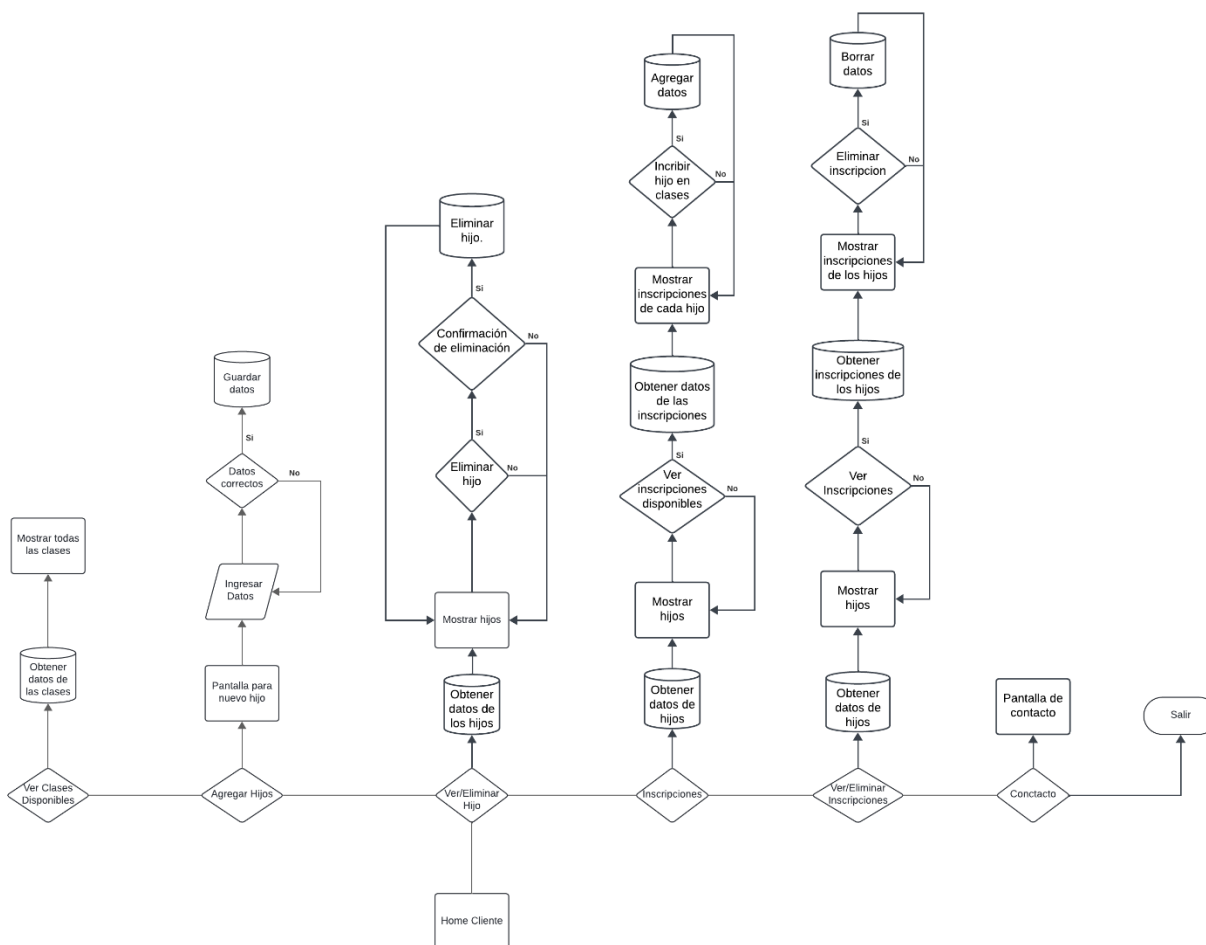
Diagrama de flujo inicio de la aplicación.



La Figura 1 ilustra el proceso de inicio en la aplicación, detallando las opciones de interacción del usuario desde el momento en que ingresa a la aplicación hasta que accede a su perfil. El flujo comienza con el "Inicio", donde el usuario tiene dos eventos principales a elegir: "Login" o "Registro" (ver Imagen 1 del documento Anexos), dependiendo de si ya tiene una cuenta o necesita crear una nueva. Existe la opción de restablecer la contraseña por si es necesario para el usuario.

Figura 2

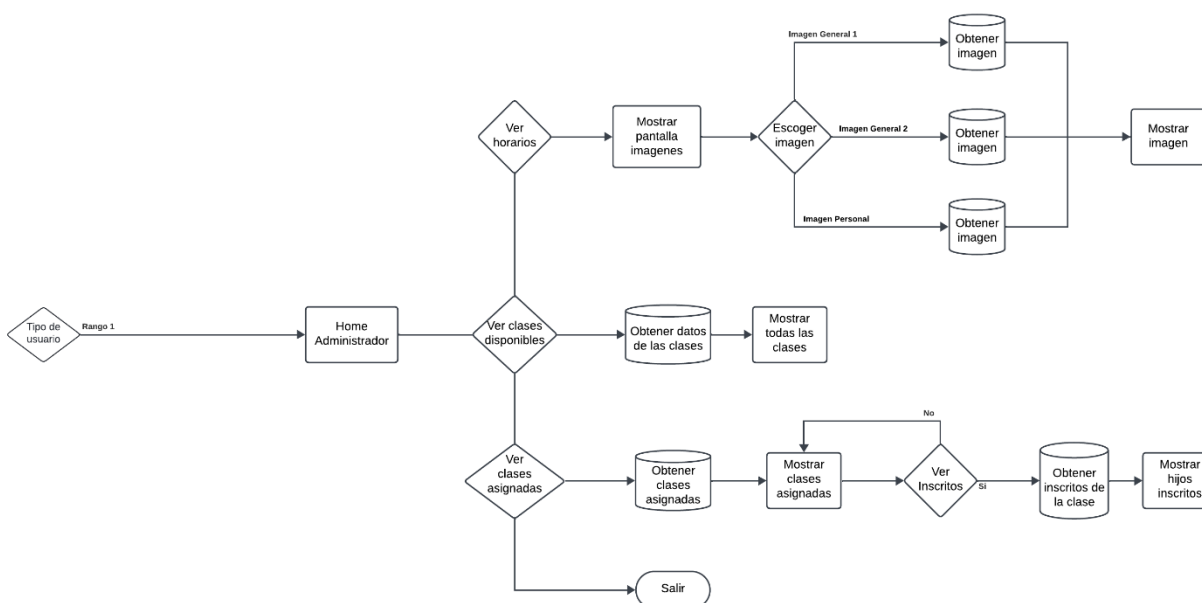
Diagrama de flujo usuario rango 0.



La Figura 2 explica el diagrama de flujo para usuarios con rango 0, padres de familia, ilustra cómo estos usuarios pueden interactuar con la aplicación para gestionar eficazmente la información de su familia y las inscripciones de sus hijos en actividades (ver Imagen 12 del documento Anexos). Este diagrama proporciona una visión clara de las operaciones disponibles y cómo se ejecutan dentro de la plataforma.

Figura 3

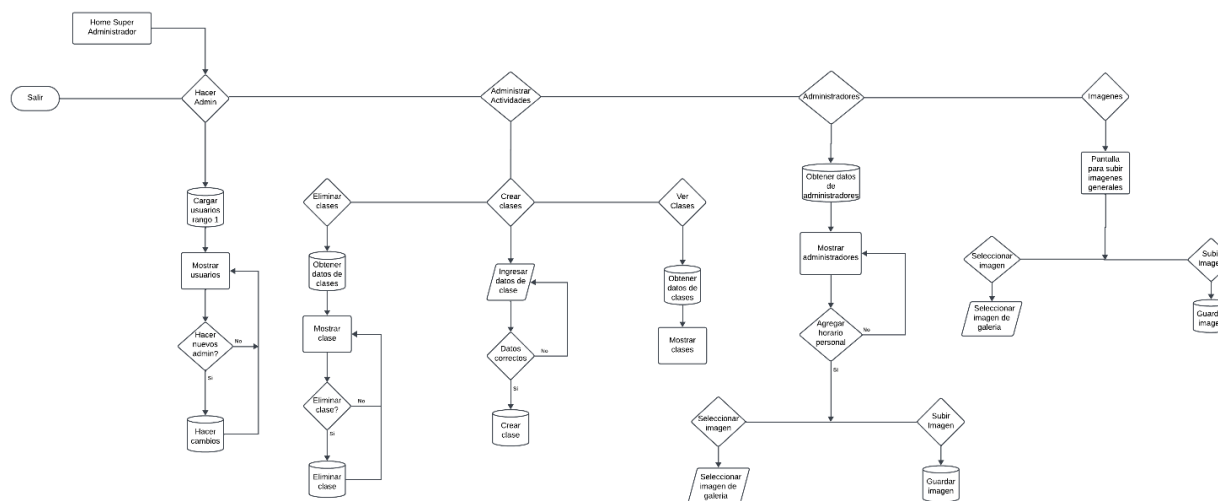
Diagrama de flujo usuario rango 1.



La Figura 3 ilustra el funcionamiento para usuarios de rango 1, son los administradores del centro vacacional, muestra las diversas funcionalidades que están a su disposición para la gestión eficiente del centro. Estos usuarios tienen acceso a un conjunto ampliado de operaciones que les permiten manejar información detallada sobre clases, inscripciones, y las imágenes utilizadas en la plataforma (ver Imagen 41 del documento Anexos).

Figura 4

Diagrama de flujo usuario rango 2, super administradores.



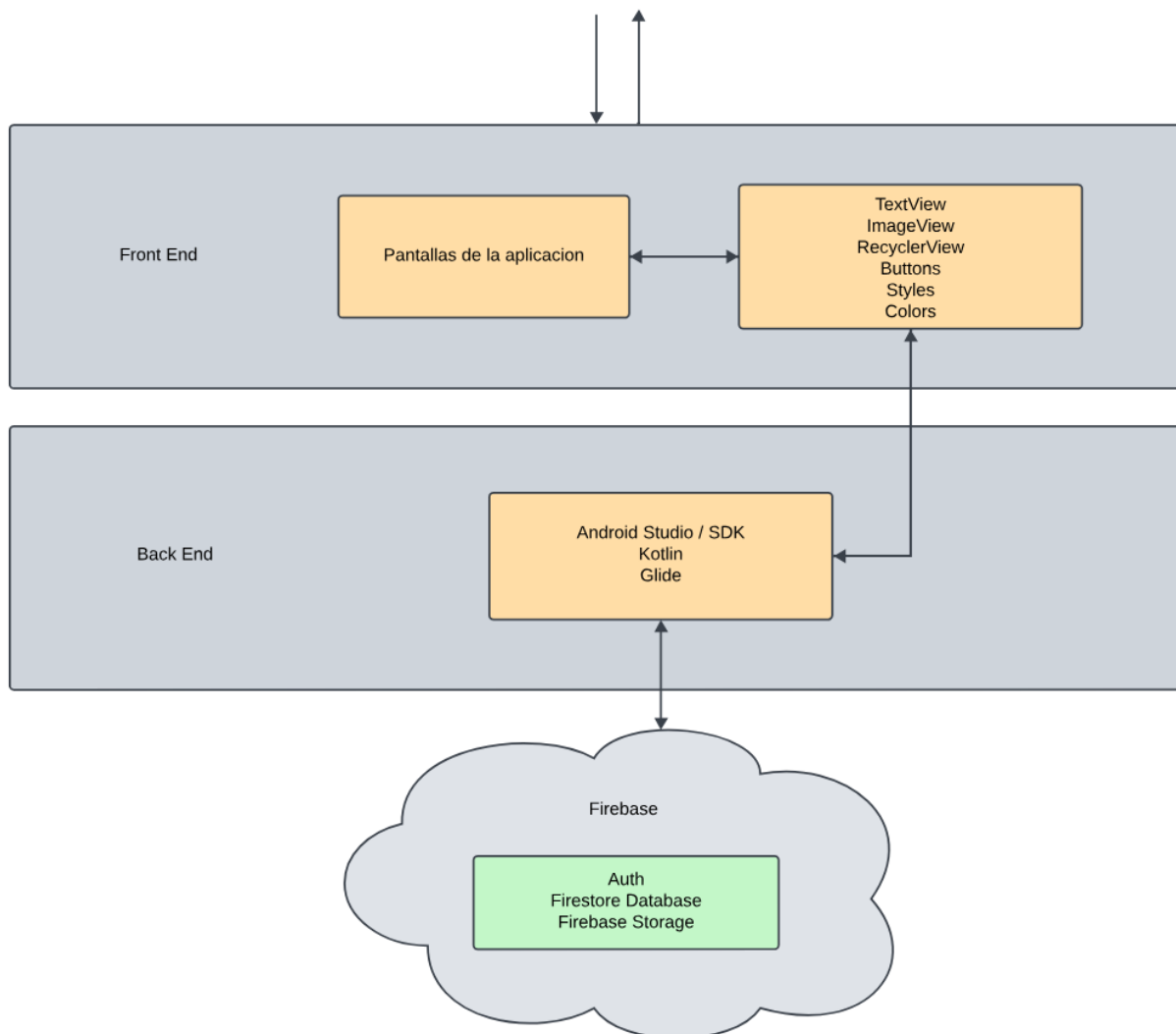
La Figura 4 detalla las funciones avanzadas disponibles para los usuarios de rango 2, conocidos como super administradores, dentro de la aplicación. Este nivel de usuario tiene la capacidad de realizar tareas de gestión de alto nivel, que incluyen administrar otros administradores, crear y eliminar clases, y manejar imágenes importantes para la aplicación (ver Imagen 27 del documento Anexos).

4.4. Diagrama de arquitectura de la aplicación

La arquitectura de la aplicación Android presentada en el diagrama ilustra cómo los diferentes componentes interactúan para proporcionar una experiencia de usuario fluida mientras se gestiona la persistencia de datos y la autenticación a través de Firebase. Aquí está el detalle de cada capa y su función en el sistema:

Figura 5

Arquitectura de la aplicación.



4.4.1. Front End

Incluye las interfaces de usuario (UI) de la aplicación que interactúan directamente con los usuarios. Estas interfaces están construidas utilizando diversos elementos UI como `TextView`,

ImageView, RecyclerView, etc., que se encargan de mostrar datos y recibir interacciones de los usuarios. (ver Imagen 1 hasta Imagen 43 del documento Anexos).

4.4.2. *Back End*

Contiene las herramientas de desarrollo como Android Studio y Kotlin, que se utilizan para programar la lógica de la aplicación. Además, utiliza Firebase para integrar la autenticación de usuarios, la gestión de datos con Firestore y el almacenamiento de archivos con Firebase Storage. Glide es utilizado aquí para optimizar la carga y el manejo de imágenes en la aplicación.

4.4.3. *Firebase*

Sirve como puente entre la aplicación y los servidores de backend, manejando datos en tiempo real, autenticación de usuarios y almacenamiento de archivos, asegurando que los datos sean accesibles y seguros. (ver Imagen 44 hasta Imagen 50 del documento Anexos).

4.4.4. *Internet*

Es crucial para la conectividad y el acceso remoto a los servicios de Firebase, permitiendo que la aplicación funcione de manera eficiente en un entorno conectado

5. IMPLEMENTACIÓN

5.1. Desarrollo de funcionalidades clave

5.1.1. *Proceso de registro e inicio de sesión.*

El proceso de registro e inicio de sesión en una aplicación es crucial para proporcionar una experiencia personalizada y segura a los usuarios. A través del registro, los usuarios pueden

crear cuentas personalizadas que les otorgan acceso a funciones exclusivas y permiten el almacenamiento seguro de su información. En este caso para el registro, tanto de clientes como administradores se piden datos como: nombre, apellido, correo electrónico, verificación de contraseñas, teléfono y cédula (ver Imagen 3 del documento Anexos), con la intención de tener información que se pueda necesitar en futuro, misma que se almacena en la base de datos una vez el registro del usuario haya sido exitoso.

El inicio de sesión, por su parte, brinda a los usuarios un acceso rápido y cómodo a la aplicación, permitiéndoles disfrutar de sus servicios de inmediato. Para iniciar sesión se solicita al usuario que ingrese su cuenta de e-mail y contraseña creada (ver Imagen 2 del documento Anexos), en caso de que alguno de estos datos tenga errores se despliega un mensaje de error solicitando verificar la información previamente ingresada.

Un proceso de registro y acceso bien diseñado es intuitivo y fácil de usar, además de ofrecer sólidas medidas de seguridad, como la verificación de identidad y la autenticación de dos factores, para proteger la información sensible de los usuarios.

5.1.2. *Gestión de usuarios.*

Inicialmente, cada usuario nuevo será asignado con un rango 0, designando su rol como "padre" o "cliente". Estos usuarios pueden registrar y gestionar la información de sus hijos, así como inscribirlos en clases disponibles dentro del centro vacacional (ver Imagen 14, ver Imagen 18, e Imagen 19 del documento Anexos). Los usuarios con rango 1 son administradores, quienes tienen la capacidad de visualizar las actividades asignadas e hijos inscritos, además de poder visualizar imágenes relacionadas con el centro (ver Imagen 41 del documento Anexos).

Finalmente, los usuarios con rango 2, conocidos como "super administradores", tienen privilegios adicionales, como la capacidad de cambiar el rango de otros usuarios, crear y eliminar clases, y gestionar imágenes de uso general y personal (ver Imagen 24 del documento Anexos). Dependiendo del rango del usuario en cuestión, éste será dirigido a la pantalla principal correspondiente.

5.1.3. Sistema de inscripciones para actividades extracurriculares.

El sistema de inscripciones para actividades extracurriculares en la aplicación permite a los clientes inscribir a sus hijos en las actividades propuestas por centro vacacional. Esta funcionalidad útil para instituciones educativas, clubes y organizaciones que ofrecen talleres, cursos, deportes y otras actividades.

Los clientes tendrán la posibilidad de:

1. Acceder al listado de actividades: los clientes pueden explorar actividades disponibles, con detalles como el nombre, el instructor, la ubicación, el horario, el límite de edad y el número de cupos (ver Imagen 13 del documento Anexos).
2. Se ha implementado una funcionalidad que permite a los padres registrar a sus hijos en las actividades disponibles, filtrando las opciones en función de la edad de cada niño. Esta implementación se realiza mediante una pantalla que muestra un listado de las actividades, utilizando un RecyclerView para desplegar las clases que están disponibles para el rango de edad del hijo seleccionado (ver Imagen 19 e Imagen 20 del documento Anexos).

3. Manejo de cupos: el sistema controla el número de plazas y evita inscripciones duplicadas o no autorizadas así evitando errores en el registro.
4. Eliminar la inscripción del hijo en caso de ser necesario (ver Imagen 21 e Imagen 22 del documento Anexos).

5.1.4. Visualización y gestión de horarios.

Para la funcionalidad de visualización y gestión de horarios, la aplicación ha implementado características que permiten tanto al super administrador como a los administradores gestionar y visualizar imágenes relevantes para el centro vacacional.

Super administrador:

1. El super administrador tiene una pantalla que permite visualizar y gestionar las imágenes generales que son visibles para todos los administradores (ver Imagen 40 del documento Anexos).
2. Esta pantalla presenta un ImageView y dos pares de botones que permiten al super administrador seleccionar y subir dos imágenes principales desde la galería (ver Imagen 38, ver Imagen 38 e Imagen 40 del documento Anexos).
3. Las imágenes se almacenan en Firebase Storage y se visualizan utilizando la biblioteca Glide, permitiendo una gestión eficiente y rápida de las imágenes (ver Imagen 48 del documento Anexos).
4. Gestión de administradores:

- a. El super administrador también tiene una pantalla que muestra una lista de todos los administradores, utilizando un RecyclerView (ver Imagen 36 del documento Anexos).
- b. En esta pantalla, el super administrador puede subir una foto personal para cada administrador (ver Imagen 37 del documento Anexos).

Administradores:

1. Los administradores tienen una pantalla que permite visualizar tres imágenes, utilizando un ImageView y tres botones etiquetados "Ver Imagen General 1", "Ver Imagen General 2" y "Ver Imagen Personal 3" (ver Imagen 43 del documento Anexos).
2. Las primeras dos imágenes son las imágenes generales subidas por el super administrador, y la tercera imagen es la imagen personal del administrador, la cual fue subida por el super administrador a cada perfil de administrador de manera individual.

5.2. Integración con Firebase:

5.2.1. Manejo de datos en tiempo real.

Para garantizar la integridad y coherencia de los datos dentro de la aplicación, se ha implementado el uso de transacciones atómicas en Firebase Firestore. Este enfoque es fundamental para el manejo de datos en tiempo real, especialmente en un entorno donde múltiples usuarios pueden intentar modificar la información simultáneamente.

Implementación de transacciones atómicas:

1. **Consistencia de datos:** las transacciones atómicas aseguran que todas las operaciones dentro de una transacción se completen con éxito antes de actualizar la base de datos. Si alguna operación falla, la transacción se revierte completamente, previniendo así cualquier inconsistencia de datos.
2. **Escenarios de uso:** un ejemplo práctico es la inscripción de estudiantes en clases. Al registrar a un estudiante en una clase, la transacción verifica primero la disponibilidad de cupos. Si hay cupo, se procede a inscribir al estudiante y reducir el número de cupos disponibles. Todo este proceso se maneja como una sola transacción para evitar que se inscriban más estudiantes de los que la clase puede admitir.
3. **Sincronización en tiempo real:** las transacciones atómicas son esenciales para mantener los datos sincronizados en tiempo real entre todos los usuarios. Cuando se efectúa una transacción, los cambios se reflejan instantáneamente en todos los dispositivos conectados, asegurando que todos los usuarios vean la información más actualizada sin demoras.
4. **Prevención de condiciones de carrera:** este método es particularmente útil para prevenir condiciones de carrera, donde múltiples procesos o usuarios intentan cambiar la misma información al mismo tiempo. Las transacciones atómicas ayudan a gestionar estas situaciones al asegurar que solo una operación pueda completarse a la vez, protegiendo así la integridad de los datos.

5.2.2. *Firebase Storage y Glide*

Firestore se utiliza para el almacenamiento y manejo de imágenes en la aplicación, proporcionando una solución robusta y escalable para gestionar archivos multimedia. Esta integración permite a los usuarios subir, almacenar y recuperar imágenes de manera eficiente y segura. La utilización de Firestore asegura que las imágenes estén disponibles en cualquier momento y desde cualquier dispositivo, facilitando un acceso global y en tiempo real.

Integración con Glide:

1. Carga y visualización de imágenes: Glide se utiliza en conjunto con Firestore Storage para optimizar la carga y visualización de imágenes dentro de la aplicación. Glide facilita la manipulación de imágenes, incluyendo el caché automático y la gestión de la memoria, lo que mejora significativamente el rendimiento de la aplicación al reducir la latencia y el uso de recursos.
2. Proceso de carga de imágenes: cuando un super administrador necesita subir los nuevos horarios, tanto horarios generales como personales de administradores, seleccionan el archivo desde su dispositivo (ver Imagen 38 del documento Anexos), que se carga a Firestore Storage. Una vez almacenada, el URL de la imagen se guarda en Firestore para su fácil acceso y referencia (ver Imagen 48e Imagen 50 del documento Anexos),
3. Visualización de imágenes: las imágenes almacenadas en Firestore Storage se recuperan y muestran en la aplicación mediante Glide, que automáticamente ajusta el tamaño de la imagen y la presenta de manera adecuada en la interfaz de usuario (ver Imagen 43 del documento Anexos). Glide también maneja

situaciones de baja conectividad o dispositivos con recursos limitados, proporcionando siempre la mejor experiencia visual posible.

4. Seguridad y permisos: Firebase Storage ofrece configuraciones detalladas de seguridad y permisos, lo que permite especificar quién puede acceder o modificar las imágenes. Esto es crucial para garantizar que solo los usuarios autorizados puedan cargar o alterar las imágenes en la aplicación.

5.3. Creación y manejo de clases necesarias

Para la aplicación se utilizan tres clases principales diseñadas para estructurar y manejar eficientemente los datos: User, Hijo, y Clases. Cada una de estas clases desempeña un papel crucial en el funcionamiento general de la aplicación.

Clase User:

1. La clase User gestiona toda la información relacionada con los usuarios de la aplicación. Incluye atributos como nombre, apellido, correo electrónico, contraseña, teléfono, cédula, club (ver Imagen 3 del documento Anexos), así como un rango que determina el nivel de acceso del usuario (padre, administrador, superadministrador).
2. Contiene también una lista de objetos Hijo, lo que permite asociar cada usuario con sus hijos registrados en el sistema.
3. Por último, la clase User incorpora un campo ImageUri, específicamente utilizado para almacenar el URL de una imagen de perfil en el caso de los administradores. Este atributo facilita la personalización de cada perfil de los administradores(ver Imagen 3 del documento Anexos).

Clase Hijo:

1. La clase Hijo está diseñada para almacenar información detallada sobre los hijos de los usuarios. Incluye campos como nombre, apellido, fecha de nacimiento y edad (ver Imagen 14 del documento Anexos), que son esenciales para el proceso de inscripción en actividades adecuadas según su edad.
2. Esta clase facilita la gestión de inscripciones y la asociación de cada hijo con las clases específicas en las que están registrados.

Clase Classes:

1. La clase Classes encapsula todos los detalles necesarios para la gestión de las clases ofrecidas por el centro vacacional. Contiene información como el nombre de la clase, el nombre del profesor, el lugar, la capacidad total de alumnos (cupos), los días de la semana en que se imparte la clase, y las horas de inicio y fin (ver Imagen 13 del documento Anexos).
2. También especifica el rango de edad permitido para asegurar que las inscripciones sean apropiadas para el grupo de edad de los niños.
3. Cada instancia de Classes tiene asociada una lista de Hijo para rastrear los niños inscritos en cada clase, lo que ayuda en la gestión de la disponibilidad y asignación de espacios (ver Imagen 49 del documento Anexos).

6. PRUEBAS Y VALIDACIÓN

6.1. Metodología de Pruebas: Cómo se planificaron y ejecutaron las pruebas para asegurar la funcionalidad y usabilidad de la aplicación.

Se realizaron pruebas con el personal del centro vacacional que solicitó el prototipo de esta aplicación, para confirmar la usabilidad de la misma, asegurando que los interfaces y flujos de trabajo fueran intuitivos y eficientes para los diferentes tipos de usuarios, incluidos padres, administradores y super administradores. Además, se llevaron a cabo varias fases de pruebas:

1. Pruebas unitarias: se implementaron para verificar cada componente de la aplicación de forma aislada. Esto incluyó pruebas de las funciones de la base de datos, verificaciones de los procesos de autenticación y seguridad, y tests de las interacciones de la interfaz de usuario.
2. Pruebas de integración: después de las pruebas unitarias, se realizaron pruebas de integración para asegurar que todos los módulos de la aplicación funcionaran juntos como se esperaba. Estas pruebas fueron cruciales para identificar problemas de interfaz entre los distintos componentes de la aplicación.
3. Pruebas de sistema: se ejecutaron para evaluar la aplicación completa y asegurarse de que funcionaba correctamente en su conjunto. Estas pruebas incluyeron escenarios de uso simulado para ver cómo la aplicación manejaba las operaciones típicas dentro del centro vacacional.
4. Pruebas de aceptación del usuario: fueron conducidas con usuarios reales de el centro vacacional para obtener retroalimentación directa sobre la funcionalidad y la experiencia del usuario. Estas pruebas ayudaron a ajustar la interfaz de usuario y las funcionalidades para mejor adaptarse a las necesidades y expectativas de los usuarios finales.

6.2. Resultados de las Pruebas: Análisis de los resultados obtenidos y cómo estos impactan en la calidad final del proyecto.

Los resultados de las pruebas fueron positivos, el personal administrativo del centro vacacional aprobó el funcionamiento del proyecto, destacando su eficiencia y la mejora significativa en la gestión de actividades y recursos. Los tests demostraron que la aplicación era capaz de manejar eficientemente todas las operaciones clave.

1. Eficacia en la gestión de usuarios: la funcionalidad de gestionar distintos niveles de acceso y permisos para padres, administradores y superadministradores fue especialmente valorada, pues permitió una administración de roles clara y segura.
2. Integridad y seguridad de los datos: las pruebas confirmaron que las medidas de seguridad implementadas eran robustas, con transacciones de datos seguras y consistentes, minimizando los riesgos de errores o pérdidas de información.
3. Respuesta a la carga de trabajo: la aplicación respondió bien bajo cargas de uso simuladas, manteniendo un rendimiento estable y rápido incluso con múltiples usuarios accediendo simultáneamente, lo que demuestra su capacidad para soportar el ambiente dinámico del centro vacacional.

Para proporcionar una evidencia concreta del éxito de estas pruebas y la aceptación del sistema por parte del personal administrativo, se adjunta documentación de la validación realizada por el personal del centro vacacional, en la que certifican la eficacia y la aceptabilidad del sistema, asegurando que la solución propuesta cumple con los requisitos operativos y las expectativas del cliente.

7. CONCLUSIONES Y TRABAJOS FUTUROS

7.1. Conclusiones

1. Los objetivos planteados al inicio del proyecto fueron alcanzados de manera satisfactoria. La aplicación móvil desarrollada proporciona una plataforma robusta y eficiente para la gestión de inscripciones y actividades en el centro vacacional. Las funcionalidades implementadas, como el registro de usuarios, la gestión de clases y la visualización de horarios, han sido integradas con éxito, mejorando la eficiencia operativa del centro.
2. Las necesidades específicas del centro vacacional que motivaron el desarrollo de esta aplicación, como la mejora en la gestión del registro de las familias inscritas y la administración de actividades, se han solventado efectivamente. La aplicación ha demostrado ser una herramienta valiosa en la automatización de procesos que anteriormente eran manuales y propensos a errores, lo que ha resultado en una gestión más fluida y menos laboriosa.
3. La interfaz intuitiva y las funciones avanzadas de gestión han mejorado significativamente la experiencia de los usuarios, tanto administradores como clientes del centro vacacional. La facilidad de uso y la accesibilidad de la información han aumentado la satisfacción del usuario, lo que se refleja en las respuestas positivas recibidas durante las fases de prueba.
4. El proyecto ha incorporado tecnologías de vanguardia como Kotlin para el desarrollo de la aplicación y Firebase para la gestión de bases de datos y autenticación de usuarios. Esta adopción no solo ha fortalecido la infraestructura técnica del centro

sino que también ha establecido un precedente para la integración de soluciones digitales más avanzadas en el futuro.

5. A pesar de los éxitos, el proyecto enfrentó varias limitaciones, como la dependencia de la conectividad a Internet para el funcionamiento óptimo de la aplicación y la limitación del alcance a usuarios de dispositivos Android. Estas limitaciones destacan áreas para mejoras futuras y expansión de la compatibilidad del sistema.

7.2. Evaluación del Proyecto: Reflexión sobre el cumplimiento de los objetivos, problemas enfrentados y soluciones implementadas.

La evaluación del proyecto revela un alto grado de cumplimiento de los objetivos iniciales. La aplicación diseñada para gestionar las actividades y operaciones del centro vacacional ha mejorado significativamente la eficiencia y la comunicación entre los usuarios. Sin embargo, durante el desarrollo y la implementación del proyecto, se enfrentaron varios desafíos que requirieron soluciones creativas y técnicas ajustadas.

Dificultades y Soluciones:

1. Manejo de una base de datos NoSQL: el uso de Firebase Firestore, una base de datos NoSQL, presentó desafíos iniciales en términos de diseño del esquema y consultas eficientes, especialmente dada la estructura dinámica y el volumen de datos. La solución implicó la adopción de prácticas recomendadas para NoSQL, como la de normalización de datos y el uso intensivo de índices para acelerar las consultas, lo cual ayudó a superar las limitaciones de rendimiento y facilitó una gestión de datos más fluida y escalable.

2. Adaptación al lenguaje de programación (Kotlin): la implementación utilizando Kotlin, a pesar de sus muchas ventajas, requirió un tiempo de adaptación y reconocimiento en la estructura y funcionamiento de la aplicación.
3. Experiencia del usuario: asegurar una experiencia de usuario intuitiva y satisfactoria fue un reto continuo. Se realizaron múltiples rondas de pruebas de usabilidad, que permitieron recoger feedback valioso de los usuarios reales. Este feedback fue esencial para iterar el diseño de la interfaz, haciendo ajustes que simplificaron la navegación y la interacción, mejorando así la accesibilidad y la aceptación de la aplicación.

7.3. Recomendaciones para futuras mejoras: Identificación de áreas de oportunidad y sugerencias para el desarrollo futuro de la aplicación.

Si bien la aplicación cumple con los requisitos iniciales, siempre hay espacio para mejoras y futuras investigaciones. Algunas recomendaciones para el desarrollo futuro de la aplicación incluyen:

- 7.3.1.** Mejorar la comunicación: agregar funciones como chat en tiempo real entre padres y administración, así como foros de discusión, podría potenciar la comunicación y el sentido de comunidad.
- 7.3.2.** Incorporar pagos: integrar un sistema de pagos dentro de la aplicación permitiría a los padres pagar por el programa completo o servicios adicionales directamente a través de la aplicación, aumentando la comodidad.

- 7.3.3.** Optimizar el rendimiento: continuar optimizando el rendimiento de la aplicación, especialmente en el manejo de grandes volúmenes de datos o usuarios concurrentes, asegurará una experiencia de usuario fluida y satisfactoria.
- 7.3.4.** Ofrecer soporte multiplataforma: desarrollar versiones de la aplicación para otras plataformas, como iOS, ampliaría el alcance y la accesibilidad de la solución, beneficiando a una base de clientes más amplia.

8. REFERENCIAS

- Android Developers. (s.f.). *Cómo crear diseños XML para Android*.
<https://developer.android.com/codelabs/basic-android-kotlin-training-xml-layouts?hl=es-419#0>
- Ardito, L., Coppola, R., Malnati, G., & Torchiano, M. (2020). Effectiveness of Kotlin vs. Java in Android App Development Tasks. *Information and Software Technology*, 127.
<https://doi.org/10.1016/j.infsof.2020.106374>
- Bunk1. (s.f.). *Stay Connected to the Camp Experience*. <https://www.bunk1family.com/our-services>
- Cheng, F. (2018). *Build Mobile Apps with Ionic 4 and Firebase*. Apress.
<https://doi.org/10.1007/978-1-4842-3775-5>
- ClassDojo. (s.f.). *Acerca de nosotros*. <https://www.classdojo.com/es-mx/>
- Hagos, T. (2018). *Learn Android Studio 3*. Apress, https://doi.org/10.1007/978-1-4842-3156-2_2
- Islam, R., Islam, R., & Mazumder, T. (2010). Mobile application and its global impact. *International Journal of Engineering & Technology*, 10(6), 72-78.
<https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=9f842c8177eee41fd96348eb4a0ae8ad45807076>
- Khawas, C., & Shah, P. (2018). Application of firebase in android app development-a study. *International Journal of Computer Applications*, 179(46), 49-53.
<http://dx.doi.org/10.5120/ijca2018917200>
- Miya, T., & Govender, I. (2022). UX/UI design of online learning platforms and their impact on learning: A review. *International Journal of Research in Business and Social Science*, 11(10), 316-327. <https://doi.org/10.20525/ijrbs.v11i10.2236>

- Moskala, M., & Wojda, I. (2017). *Android Development with Kotlin*. Packt Publishing Ltd.
https://books.google.com.ec/books?hl=es&lr=&id=PJZGDwAAQBAJ&oi=fnd&pg=PP1&dq=kotlin+programming+language+android+app+development&ots=3LdceDPXsD&sig=ZCVS8JD92VnUvPK5aSz5vzRXCf4&redir_esc=y#v=onepage&q&f=false
- Moroney, L. (2017). *The Definitive Guide to Firebase*. Apress, https://doi.org/10.1007/978-1-4842-2943-9_1
- myCAMPapp. (s.f.). *Keep Everyone Informed and Happy*. <https://mycampapp.com/your-summer-camp-app>
- Panchal, R., & Patel, A. (2017). A comparative study: Java vs Kotlin programming in android. *International Journal of Innovative Trends in Engineering & Research*, 9(2), 4-10.
<https://doi.org/10.26483/ijarcs.v9i3.5978>
- Phongtraychack, A., & Dolgaya, D. (2018). Evolution of mobile applications. MATEC Web of Conferences, 155. <https://doi.org/10.1051/mateconf/201815501027>