



## **MIGRACIÓN DE LA APLICACIÓN MÓVIL SION ESTUDIANTES DE LA CORPORACIÓN UNIVERSITARIA ADVENTISTA**

Yineth Alejandra Valderrama Jaramillo  
Stephania Osorio Granada

Trabajo de grado para optar al título de  
Ingeniería Sistemas

Director

Asesor Temático: Edna Katherine Conde Vega Esp.

Asesor Metodológico: Walter Hugo Arboleda Mazo Ph. D.

Corporación Universitaria Adventista de Colombia

Facultad de Ingeniería

Ingeniería de Sistemas

Medellín

2023



**UNAC**  
CORPORACIÓN UNIVERSITARIA ADVENTISTA  
COLOMBIA

Personería Jurídica según Resolución del Ministerio  
de Educación No. 8529 del 6 de junio de 1983  
Carrera 84 No. 33AA-1 Medellín, Colombia  
PBX: 60 4 480 55 90 / NIT: 860 403 751-3  
www.unac.edu.co

**FACULTAD DE INGENIERÍA**  
**CENTRO DE INVESTIGACIONES**

**NOTA DE ACEPTACIÓN**

Los suscritos miembros de la Comisión Asesora del proyecto de grado titulado: **“Migración de la aplicación móvil SION Estudiantes de la Corporación Universitaria Adventista Colombia”**, elaborado por los estudiantes: **Yitned Alejandra Jaramillo Valderrama y Stephania Osorio Granada**, del programa de **Ingeniería de Sistemas**, nos permitimos conceptuar que éste cumple con los criterios teóricos y metodológicos exigidos por la Facultad de Ingeniería y por lo tanto se declara como:

Aprobado Bueno

Medellín, 9 de noviembre de 2023

Saray Galeano

**Mg. Saray Galeano Ospino**  
Presidente

Edna Katherine Conde Vega

**Esp. Edna Katherine Conde Vega**  
Secretaria

Walter Hugo Arboleda Mazo

**Phd(c). Walter Hugo Arboleda Mazo**  
Vocal

Y. Alejandra Jaramillo V.

**Yitned Alejandra Jaramillo Valderrama**  
Estudiante

Stephania Osorio Granada

**Stephania Osorio Granada**  
Estudiante

## Contenido

<b>1. Introducción.....</b>	<b>7</b>
<b>CAPÍTULO 1 Propuesta de Investigación .....</b>	<b>9</b>
1.1 Planteamiento del Problema .....	9
1.2 Justificación .....	10
1.3 Antecedentes.....	11
1.4 Objetivos de la Investigación .....	12
<b>CAPÍTULO 2 Marco Teórico.....</b>	<b>13</b>
2.1 Marco Conceptual.....	13
2.2 Marco de referencia.....	14
2.3 Marco Institucional.....	16
2.4 Marco Legal.....	17
<b>CAPÍTULO 3 Marco Metodológico.....</b>	<b>18</b>
3.1.1 Fase 1: Planificación y evaluación: .....	18
3.1.2 Fase 2: Análisis de documentación:.....	18
3.1.3 Fase 3: Diseño de la nueva arquitectura.....	19
3.1.4 Fase 4: Desarrollo y adaptación .....	20
3.1.5 Fase 5: Pruebas y depuración .....	20
3.1.6 Fase 6: Ajustes .....	21
3.1.7 Fase 7: Cierre del proyecto.....	21
<b>CAPÍTULO 4 Resultados y Análisis de Resultados .....</b>	<b>22</b>
4.1 Resultados.....	22
4.2 Análisis de resultados.....	33
<b>CAPÍTULO 5 Conclusiones y Recomendaciones.....</b>	<b>43</b>
5.1 Conclusiones.....	43
5.2 Recomendaciones .....	44
<b>2. Referencias Bibliográficas.....</b>	<b>45</b>

## Lista de Tablas

Tabla 1: Requerimiento del software	19
Tabla 2: Requerimientos del hardware	20

## Lista de Figuras

Ilustración 1: Metodología en Cascada	22
Ilustración 2: Visual Story Map	23
Ilustración 3: Inicio de Sesión	24
Ilustración 4: Validaciones	25
Ilustración 5: Pantalla principal	26
Ilustración 6: Perfil de usuario	27
Ilustración 7: Cursos y calificaciones	28
Ilustración 8: Horarios	29
Ilustración 9: Carné estudiantil	30
Ilustración 10: Cerrar sesión	31
Ilustración 11: API Rest	32
Ilustración 12: Datos de autenticación en API Rest	33
Ilustración 13: Datos de perfil con API Rest	33
Ilustración 14: Marcos móviles multiplataforma	35
Ilustración 15: Métricas de rendimiento con Performance Monitor	37
Ilustración 16: Tareas asíncronas	38
Ilustración 17: Métricas de rendimiento con React Native DevTools	39
Ilustración 18: Carga en pantalla principal	40
Ilustración 19: Diferencias entre pantallas de inicio de sesión	42

## Resumen

La migración de aplicaciones se ha convertido en un fenómeno global de gran importancia en las últimas décadas, debido a nuevas y mejores tecnologías impuestas en el mercado actual, es por esta razón que diferentes organizaciones han optado por migrar sus desarrollos de software, con el fin de mejorar su estructura de costo, obtener mayor seguridad, rendimiento, escalabilidad adaptable y capacidad de actualizar y responder a cambios continuos de acuerdo con sus necesidades.

El objetivo del proyecto fue migrar la aplicación móvil SION Estudiantes, donde se recrea de una primera versión ya desarrollada, funcionalidades relevantes como el acceso al aplicativo, visualización de cursos matriculados, calificaciones y horario asignado a cada materia.

Esta migración se llevó a cabo desde la plataforma de código abierto Xamarin al framework de código abierto React Native, además se modificaron los estilos de acuerdo con la colorimetría que la Corporación Universitaria Adventista (UNAC) tenía en el momento.

Como resultado final se espera sumar y animar a los estudiantes para que hagan uso del aplicativo, ofreciéndoles a través de esta comodidad y sencillez al momento de realizar el control de su información académica.

**Palabras claves:** Migración, aplicación móvil, Android, SVGA, sistema de información académica, React Native.

## Abstract

The migration of applications has become a global phenomenon of great importance in recent decades, due to new and better technologies imposed on the current market, it is for this reason that different organizations have chosen to migrate their software developments, in order to improve their cost structure, obtain greater security, performance, adaptable scalability and ability to update and respond to continuous changes according to their needs.

The objective of the project was to migrate the SION Students mobile application, where relevant functionalities such as access to the application, visualization of enrolled courses, grades and timetable assigned to each subject were recreated from a first version already developed.

This migration was carried out from the open source platform Xamarin to the open source framework React Native, also the styles were modified according to the colorimetry that the Corporación Universitaria Adventista (UNAC) had at the time.

As a final result it is expected to add and encourage students to make use of the application, offering them through this convenience and simplicity at the time of making the control of their academic information.

**Key Words:** Migration, Mobile App, Android, SVGA, Academic Information System, React Native.

## 1. Introducción

El ingreso de las nuevas tecnologías en los diferentes ámbitos de la vida (social, laboral y formativo), ha hecho que diversas organizaciones educativas, tanto áreas administrativas como no administrativas (estudiantes), tengan que sumergirse en el amplio mundo de las Tecnologías de la información y las comunicaciones (TIC), cabe decir que esta forma de trabajo se ha introducido desde mucho tiempo atrás y su proceso ha sido satisfactorio, con resultados favorecedores que fortalecen la comunicación y propician una mayor motivación en los educandos (Gellibert Merchán et al., 2021). Sin embargo, con el pasar de los tiempos muchas organizaciones han tenido que construir un plan estratégico para la migración de sus aplicativos, debido a nuevas implementaciones y actualizaciones, un claro ejemplo es la reconocida empresa Netflix quienes tuvieron que migrar su infraestructura a la nube, debido a un fallo en las bases de datos, pasando así de un enfoque basado en centro de datos al reconocido servidor Amazon Web Services (Izrailevsky et al., 2016), pasando por lo mismo miles de empresas reconocidas.

Es por esto que la Corporación Universitaria Adventista (UNAC) ha iniciado una reforma en SION (sistema de información), poniendo en práctica y utilizando nuevos productos de desarrollo para el mejoramiento de todo el sistema, dicho sistema se encarga de gestionar el perfil del estudiante, sistemas misionales (Test de psicología), sistemas de apoyo (Labor educativa, proceso de la UNAC como inscripciones y matrículas, salud ocupacional, informes de signos Covid-19 y votaciones), también se encarga del SVGA (Sistema Virtual de Gestión Académica), que en general es la plataforma donde el estudiante puede ver sus cursos, notas y todo el portafolio asignado al estudiante.

El sistema actualmente contiene una aplicación móvil llamada "SION Estudiantes", que replica algunas de las funcionalidades presentes en el sistema principal de SION. Esta aplicación se ha desarrollado utilizando la plataforma Xamarin Android. Dado que el sistema principal está experimentando cambios significativos y una reestructuración tecnológica, es pertinente que esta aplicación móvil también sea migrada para que sea coherente con los nuevos desarrollos tecnológicos del sistema en su conjunto. El objetivo de realizar este proyecto es proveer un valioso respaldo al equipo de desarrollo en la ejecución de esta migración tecnológica. Esto involucra el cambio de la tecnología existente, a una nueva plataforma (React Native) que sea más acorde con las necesidades tecnológicas actuales de la organización.

La realización del proyecto está compuesta por los siguientes apartados:

CAPÍTULO 1. En el primer capítulo se identifica y analiza la problemática que enfrenta la aplicación actual. Esto se hace a través de la justificación y definición de los objetivos del proyecto, con el fin de encontrar soluciones que sean factibles y que sean implementadas de manera oportuna.

CAPÍTULO 2. En el segundo capítulo, se recopila diferente información teórica sobre los elementos que forman parte de la investigación. Esta información es importante para comprender el tema en profundidad y determinar las direcciones actuales de la migración de aplicaciones.

CAPÍTULO 3. En el tercer capítulo se describen y explican las diferentes fases y metodologías que se utilizaron para el desarrollo del proyecto

CAPÍTULO 4. En el cuarto capítulo, se da a conocer los resultados alcanzados con la metodología utilizada en el desarrollo del proyecto, así demostrando que se alcanzó nuestro objetivo general y específicos.

CAPÍTULO 5. En el último capítulo, se expone las conclusiones y recomendaciones sobre la evidencia obtenida en el aplicativo móvil "SIONEstudiantes".

## **CAPÍTULO 1 Propuesta de Investigación**

En este primer capítulo, se plantea la problemática y solución a través de la justificación y los objetivos del proyecto, se describen antecedentes, teorías, conceptos y normas que soportan el estudio previo de la investigación.

### **1.1 Planteamiento del Problema**

En el panorama actual de la tecnología móvil, la aplicación móvil SION Estudiantes se ha mantenido fiel a su plataforma original, Xamarin Android con lenguaje de programación C#. Aunque esta elección ha permitido que la aplicación funcione de manera consistente durante un periodo de tiempo prolongado y sus funciones siguen siendo mantenibles, es necesario que la aplicación se actualice a una infraestructura tecnológica actual.

La migración de la aplicación móvil SION Estudiantes a una tecnología más actualizada se convierte en una necesidad importante debido a varias razones fundamentales, en primer lugar, el sistema SION en su versión web ha experimentado una actualización significativa, adoptando el código fuente en JavaScript para su interfaz de usuario. Por lo tanto, es crucial que el aplicativo móvil también se actualice y se alinee con la misma tecnología, permitiendo una integración más coherente y efectiva en todo el sistema.

Además, la complejidad del código actual desarrollado en C# plantea desafíos considerables, especialmente para las nuevas generaciones de desarrolladores que tienden a utilizar lenguajes de programación más modernos y versátiles. La migración hacia una tecnología actualizada no solo simplificará el código, sino que también facilitará su comprensión y permitirá una implementación más eficiente. Esto, a su vez, acelerará la incorporación de nuevas funcionalidades y mejoras, aprovechando el conocimiento y la experiencia de los desarrolladores en las tecnologías más recientes.

La migración a una tecnología actualizada no solo garantizará la cohesión del sistema SION en su conjunto, sino que también mejorará la eficiencia en el desarrollo y la escalabilidad de la aplicación móvil, alineando con las referencias y competencias de los desarrolladores actuales. Esta transición representa un paso fundamental para mantener la aplicación competitiva y en sintonía con las últimas tendencias tecnológicas.

## 1.2 Justificación

La migración de la aplicación móvil “SION Estudiantes” es justificada por diferentes razones fundamentales que impactan en la eficiencia operativa de la institución educativa como en la experiencia de los usuarios.

Esta aplicación será implementada para el beneficio de la UNAC, ya que va de la mano con la información que proporciona el SVGA, al momento de usarla facilita la manera en que los estudiantes podrán ver sus calificaciones o hacer consultas de las materias matriculadas en semestres anteriores, docentes asignados, historial de notas y horario de clases. El aplicativo será construido para el uso en dispositivos Android, la cual es la plataforma de mayor uso a nivel mundial. Según cifras de marzo de 2023, Android mantiene en la actualidad su posición como sistema operativo móvil líder a nivel mundial, con una cuota de mercado del 71% (Mena, 2023).

Como ya se ha mencionado anteriormente la complejidad del código actual dificulta la incorporación de nuevas características y la corrección de errores, también migrar a una plataforma más moderna permitirá simplificar el código y seguir mejores prácticas de desarrollo, proporcionará una arquitectura escalable que permita el crecimiento sostenido de usuario y datos, lo que facilitará el mantenimiento y acelerar el desarrollo de nuevas funcionalidades. Además, ayudará a abordar posibles vulnerabilidades y debilidades de seguridad en el código actual, asegurando que la aplicación cumpla con los patrones de seguridad actuales y protegiendo la información confidencial de los usuarios, aprovechar las últimas tecnologías garantizará que los usuarios tengan acceso a una experiencia de usuario moderna y eficaz, esto no solo mejora la satisfacción del usuario, sino que también genera compromiso con la innovación tecnológica.

La migración contribuirá a que los estudiantes puedan acceder de manera más sencilla y eficiente a su información académica, lo que es esencial para el éxito en el ámbito educativo, generando además una curva de aprendizaje a nivel técnico frente al equipo de desarrollo.

### 1.3 Antecedentes

El proceso de migrar una aplicación se ha vuelto esencial en su mismo contexto, ya que permite tanto a propietarios como a desarrolladores mantener las funcionalidades de sus productos en un ecosistema de constante evolución. En este apartado se proporciona una visión general de los antecedentes de migración de aplicaciones móviles que abarcan una amplia gama de industrias y contextos, resaltando así su importancia en el mundo y la tecnología actual.

Entre las principales migraciones de aplicaciones se encuentra el sector tecnológico, empresas como microsoft y facebook, han hecho migraciones importantes de aplicaciones y servicios móviles, un ejemplo de esto es el aplicativo de Facebook, que para el año 2012 migró su aplicación móvil de HTML5 a tecnologías nativas, con el fin de mejorar el rendimiento y la experiencia de usuario (Pavan, 2012), sin embargo, aunque se creyera que este sector maneja un mayor flujo de migraciones, otros sectores como la banca móvil también le apuestan a mejorar sus productos, uno de ellos fue el banco financiero BBVA quien migró de servidores locales a servidores en la nube, reduciendo así tiempos de espera y respuesta, mejorando la satisfacción de los usuarios (Innovacion, 2022), por otro lado se encuentran también las aplicaciones gubernamentales, el gobierno que es una de las instituciones más grandes a nivel mundial, migran constantemente sus aplicativos con el fin de brindar servicios y una experiencia de usuario más fluidos, esto puede incluir la migración de aplicaciones de servicios públicos, salud y educación, un ejemplo en este caso es el proceso de actualización de la aplicación móvil del Servicio Nacional de Salud (NHS) en el reino unido, se realizó con el objetivo de brindar servicios de salud más accesibles y eficientes a los ciudadanos, lo que refleja el enfoque de muchos gobiernos en mejorar sus aplicativos para beneficio de la población (Hughes, 2019).

Muchos ejemplos como estos pueden definirse en este contexto, empresas del sector educativo y entretenimiento como Netflix, Spotify, Coursera, los mencionados anteriormente y entre otros, demuestran que la migración de aplicaciones es una práctica común en la industria de desarrollo de software y tecnologías. Las razones para esas migraciones varían, pero suelen estar relacionadas con la mejora de la experiencia del usuario, la seguridad, el rendimiento y la capacidad de mantenerse al día con las tecnologías emergentes.

## 1.4 Objetivos de la Investigación

Migrar la aplicación móvil del sistema SION Estudiantes, al framework de código abierto React Native, con el fin de favorecer el rendimiento, estabilidad y usabilidad del aplicativo.

Para formular la estrategia se proponen los siguientes objetivos específicos:

- Objetivo 1: Analizar los requisitos técnicos y de negocio de la aplicación, así como los requisitos de la nueva plataforma.
- Objetivo 2: Definir dentro del código que partes se pueden reutilizar, qué partes necesitan ser reescritas y cuáles deben ser eliminadas.
- Objetivo 3: Desarrollar una primera versión de la aplicación con React Native.
- Objetivo 4: Validar la aplicación de acuerdo con los requisitos definidos en la parte inicial del proyecto.
- Objetivo 5: Documentar todo el proceso del proyecto de desarrollo como evidencia e información para nuevos desarrolladores.

## **CAPÍTULO 2 Marco Teórico**

En este capítulo se recopila información teórica de los componentes que hacen parte de la investigación, esta información será importante para tener un mejor conocimiento del tema y definir cuáles son las tendencias de la migración de aplicaciones.

### **2.1 Marco Conceptual**

A continuación, podrán encontrar la definición de conceptos utilizados en el proyecto:

#### **SION**

(Sistema de Información Organizacional Universitario) Sirve para facilitar los procesos que requieren gestión de información por parte de los administrativos universitarios, como la inscripción, matrícula y otros procesos, incorporando las nuevas tecnologías de información (Proyecto Educativo de Programa, 2018).

#### **SVGA**

(Sistema de Información Organizacional Universitario) Este es un sistema de información académica de cada uno de los programas académicos como el registro de notas, ausencias, planeación académica, asignaciones de las cargas académicas e información de los docentes (Proyecto Educativo de Programa, 2018).

#### **Aplicación Móvil**

También conocida como App móvil, diseñadas para ejecutarse en dispositivos móviles, son diseñadas para darle al usuario un mejor servicio y experiencias de calidad (Herazo, 2022).

Se considera aplicación móvil, a aquel software desarrollado para dispositivos móviles. Móvil se refiere a poder acceder desde cualquier lugar y momento a los datos, las aplicaciones y los dispositivos (Enriquez & Casas, 2014).

#### **Migración de aplicaciones**

Es el proceso de trasladar las aplicaciones ya sean tecnologías o entornos, estas migraciones se realizan para beneficiarse de una mejor estructura, escalabilidad, adaptabilidad y capacidad de actualizar las aplicaciones a los cambios continuos de acuerdo con las necesidades (Vmware, 2023).

Se le llama migración de aplicaciones “al proceso de mover uno o varios programas o aplicaciones de un entorno a otro. Estas migraciones se producen por muchas razones diferentes, tales como cierres de centros de datos, problemas con la seguridad del servidor, recuperación de desastres, actualizaciones del hardware o del funcionamiento del servidor, etc” (Uribe, 2021).

### **Xamarin Android**

Es una plataforma de desarrollo móvil para crear aplicaciones para Android utilizando C# y lenguajes de programación .NET (Bambu mobile, 2023).

Pertenece a Microsoft, utiliza principalmente el lenguaje C#, que es un lenguaje de programación desarrollado y estandarizado por Microsoft, tiene su propio entorno de desarrollo integrado IDE (Jimenez, 2021).

### **React Native**

Es un framework de JavaScript para crear aplicaciones reales nativas para Android y iOS, basado en la librería de JavaScript React (Deloitte, 2023).

Esta plataforma se basa en JavaScript y utiliza la librería de React.JS. utiliza un conjunto de componentes predefinidos que facilita a los desarrolladores la creación de las aplicaciones (Jimenez, 2021).

### **API**

(Interfaz de Programación de Aplicaciones), se refiere a cualquier software con una función distinta se define la manera como se comunican mediante preguntas y respuestas (AWS, 2023).

Es un sistema creado por cualquier tecnología, que permite comunicar distintas aplicaciones (Aguirre, 2022).

## **2.2 Marco de referencia**

Uno de los referentes más importantes que tuvimos en cuenta para la realización de este trabajo y al cual se le está haciendo la migración, fue el proyecto de grado de Daza Acevedo, estudiante de la facultad de Ingeniería de Sistemas de la UNAC, el objetivo del proyecto fue generar una aplicación móvil nativa, para el sistema operativo Android, donde los estudiantes pudieran hacer seguimiento a su proceso académico, consultar información institucional sobre

eventos y actividades, y realizar la evaluación de los docentes a cada materia matriculada, el motivo por el cual se realizó el trabajo fue debido que la interfaz gráfica del aplicativo web no se ajustaba en los dispositivos móviles, lo que impedía tener una experiencia de usuario agradable (Daza et al., 2015).

Para solventar este problema definieron requisitos funcionales como: la gestión de acceso, gestión de grupos, gestión de publicidad y anuncios, gestión de horarios, gestión de cursos, gestión de notas, entre otros, utilizando una interfaz gráfica acorde con cada funcionalidad; para el desarrollo del proyecto utilizaron la metodología Extreme Programming (XP) que consiste en crear sistemas de alta calidad, basados en una estrecha colaboración con el cliente, pruebas constantes y ciclos de desarrollo cortos. Plantearon además las siguientes fases: fase de exploración, fase de planificación del proyecto, fase de iteración y fase de mantenimiento, para cada una de las fases comparten un amplio desarrollado documental que se encuentra publicado en el repositorio de la universidad, la aplicación fue desarrollada con éxito y los resultados satisfactorios (Daza et al., 2015).

Un segundo referente fue el de la Universidad de Stanford, quien en 2019 realizó una migración a su aplicativo móvil con el fin de mejorar la experiencia estudiantil y proporcionar un acceso mucho más fácil a información relevante y recursos en el campus.

Esta nueva versión “Stanford Mobile 3.0” ofrecía en la pantalla principal del aplicativo un diseño nuevo, que consistía en priorizar contenido de interés para los estudiantes, incluyendo en ella una navegación con botones desplazables que facilitan la búsqueda rápida, contenido actual e importante del campus, marcado de los restaurantes más populares e información de su estado (abierto, cerrado) en el momento, además de generar acceso rápido a la información de cada ubicación.

el equipo TI de la Universidad constantemente desarrolla nuevas y mejoradas actualizaciones al aplicativo, generando así una mayor interacción con los estudiantes, quienes a su vez hacen recomendaciones para el crecimiento de la Aplicación (Knox, 2019).

Un tercer referente fue el de la Universidad de Harvard, quienes migraron de una aplicación para iOS, a una aplicación nativa para Android e iOS, además de generar un aplicativo web móvil accesible para cualquier teléfono inteligente.

Este aplicativo brinda acceso a mapas, directorios, menús del comedor del campus, así como a noticias, eventos y catálogos de los cursos de la Universidad. Algunas actualizaciones y nuevas funcionalidades fueron: actualización de la interfaz, nueva información nutricional en el módulo de comidas; actualización en los mapas, una nueva funcionalidad (módulo) donde se generan alertas con números de teléfono clave y un espacio para contenido de alertas, una nueva funcionalidad (módulo) fotográfico que captura escenas de la vida en el campus, también como parte de la actualización, generaron una nueva aplicación móvil de Admisiones.

La migración de esta aplicación también hace referencia al cambio hacia una plataforma de software como servicio, lo que les permite hacer actualizaciones constantes del lado del servidor y permitirles así aprovechar las nuevas funcionalidades y mejoras disponibles (Harvard Gazette, 2013).

### **2.3 Marco Institucional**

La Corporación Universitaria Adventista es una institución universitaria, de carácter privado y confesional que pertenece a la Iglesia Adventista del Séptimo Día, se dedica a *transformar personas en líderes íntegros, innovadores, con responsabilidad social y ambiental, al servicio de Dios y del mundo* (UNAC, 2021a).

Como parte de su crecimiento y transformación la universidad cuenta con diferentes dependencias que ayudan en el buen funcionamiento de la misma, dos de ellas son la vicerrectoría Académica y el Departamento de Sistemas e Informática (DSI), la primera es responsable de todo lo relacionado con el quehacer académico de la universidad, así como es responsable de liderar la creación, renovación y acreditación de programas académicos, y la segunda es una dependencia que innova, desarrolla y aplica nuevas tecnologías de la información y comunicación, su propósito es administrar y desarrollar eficientemente todos los recursos TIC de la universidad (UNAC, 2021a).

El proyecto al ser una aplicación académica hace parte de las dependencias mencionadas anteriormente, entrando a apoyar no solo en estas áreas sino también al plan estratégico de la universidad quienes proponen indicadores en desarrollo a cumplir desde el año 2021 al año 2025 (Barrios et al., 2020) teniendo en cuenta esto, se determinó que el proyecto aporta en la matriz estratégica en docencia, su proceso es, *Innovar en el proceso de enseñanza de aprendizaje* y su indicador es, *% de nivel de satisfacción de los estudiantes con las estrategias pedagógicas*. El proyecto apoya en este indicador, debido a que, según el nivel de desarrollo de

la aplicación, los estudiantes pueden quedar o no satisfechos con la elaboración de esta, dando así un porcentaje de pérdida/mejora al indicador.

Otra matriz en la cual el proyecto aporta valor es la matriz estratégica eje de gestión, su proceso es, *Adoptar un plan maestro de infraestructura física y tecnológica acorde a las necesidades institucionales* y su indicador es, *# de nuevos proyectos tecnológicos implementados*. Aunque este proyecto no es nuevo, la migración de la aplicación requiere de la implementación de nuevas tecnologías y funcionalidades, lo que conlleva a generar un nuevo proyecto (UNAC, 2021b).

## **2.4 Marco Legal**

La política de Protección de datos de la UNAC tiene como objetivo aplicar a la Ley 1581 de 2012 y al Decreto 1377 de 2013 sobre protección de datos personales (Mineducación, 2020), un marco legal y normativo que garantice la privacidad y seguridad de los datos manejados por la organización, establece como responsable del tratamiento de la protección de datos personales al Director del DSI, el cual debe atender las peticiones, consultas y reclamos ante la cual el titular de la información puede ejercer sus derechos a: Conocer, actualizar, rectificar y suprimir información revocar la autorización (UNAC, 2013).

Uno de los deberes de la UNAC como responsable del Tratamiento es garantizar al titular, en todo tiempo, el pleno y efectivo ejercicio del derecho de hábeas data, esto bajo la Ley 1266 de 2008, por la cual se regula el manejo de la información contenida en bases de datos personales, en especial, financiera, crediticia, comercial, de servicios y la proveniente de terceros países (Secretaría Jurídica Distrital, 2008).

De acuerdo a lo anterior el actual proyecto se regula bajo las normativas y reglamentos que se respaldan con las leyes gubernamentales del País Colombiano, ya que en algún momento este proyecto tiene conexión a las bases de datos personales de los estudiantes, para esto nos apoyan normas de confidencialidad con políticas de autorización donde podemos tener consentimiento al manejo de información del proyecto o cualquier operación o conjunto de operaciones sobre datos personales, tales como la recolección, almacenamiento, uso, circulación o supresión. Cualquier información vinculada o que pueda asociarse a una o a varias personas naturales determinadas o determinables. Cualquier persona natural o jurídica, pública o privada, que por sí misma o en asocio con otros realicen el tratamiento de datos personales por cuenta del responsable de tratamiento de datos (UNAC, 2013).

## **CAPÍTULO 3 Marco Metodológico**

La metodología adoptada en el proyecto tiene un enfoque cualitativo, además se han considerado diversas etapas de desarrollo para llevar a cabo el estudio. Estas fases incluyen la planificación y evaluación inicial, el análisis de la documentación existente, el diseño de la nueva arquitectura, la fase de desarrollo y adaptación de la aplicación, la etapa de pruebas y depuración para garantizar su funcionamiento óptimo, la fase de ajustes para corregir posibles problemas identificados y finalmente, la fase de cierre del proyecto, que implica la consolidación de los resultados obtenidos. A continuación, se proporciona una descripción detallada de cada una de estas fases

### **3.1.1 Fase 1: Planificación y evaluación:**

Esta primera fase representa el punto inicial del proyecto, en ella se procede a realizar un análisis evaluativo, facilitando una visión detallada y completa sobre la aplicación a la cual se le realizará la migración, se determina el porqué de la migración y se procede en colaboración estrecha con las partes interesadas en identificar los objetivos, permitiendo así una definición clara del propósito y metas que se desean alcanzar. Además, se evalúan los sistemas existentes del aplicativo, es decir, se realiza un análisis sobre la infraestructura tecnológica, su arquitectura y posibles limitaciones o desafíos que puedan surgir. Esta fase proporciona la base principal de planificación y ejecución exitosa del proyecto.

### **3.1.2 Fase 2: Análisis de documentación:**

Después de haber hecho un primer acercamiento con las partes interesadas y responder al porqué de la migración de la aplicación, se procede a realizar una lectura profunda sobre la documentación relacionada con el anterior proyecto (documento de biblioteca y código de programación), con el fin de generar un análisis amplio y lograr entender el funcionamiento a detalle de la aplicación, es importante que al finalizar dicho análisis se pueda responder a preguntas como: ¿Para qué sirve la aplicación? ¿Qué vistas tiene la aplicación? ¿Qué funcionalidades tiene? ¿Es congruente el desarrollo aplicativo con el documento realizado?

En esta fase se debe determinar un plan de migración detallado, es decir, se debe asignar recursos y/o tareas al equipo de desarrollo, así como elaborar un plan de comunicación y entregables del proyecto.

### 3.1.3 Fase 3: Diseño de la nueva arquitectura

En esta fase se consideran aspectos como: la escalabilidad, infraestructura tecnológica y seguridad con el fin de asegurar que la arquitectura sea resistente y cumpla con los estándares requeridos, se detalla la interfaz de usuario asegurando que sea intuitiva, atractiva y capaz de brindar una experiencia de usuario excepcional.

Además, se determinan las funcionalidades existentes y nuevas que se tendrán en cuenta para el desarrollo, y se analizan diferentes herramientas de ayuda para el rendimiento y visualización de problemas. A continuación, se mencionan algunas herramientas y métodos utilizados para monitorear el rendimiento.

React DevTools: esta herramienta permite depurar la aplicación y visualizar el árbol de componentes de la aplicación, ayudando a identificar problemas de renderización innecesarios.

React Native Performance Monitor: esta es una biblioteca que permite medir y visualizar en tiempo real el rendimiento de la aplicación, lo que permite optimizar el código cuando sea necesario.

React.memo: es una función de React, se utiliza para la renderización de componentes funcionales y reducir la cantidad de actualizaciones innecesarias en la interfaz de usuario.

### Requerimientos del software

Tabla 1: Requerimiento del software

Nombre de la aplicación	Migración de la aplicación móvil SION Estudiantes de la Corporación Universitaria Adventista
Herramientas	<p><b>Editor de código:</b> Visual Studio Code</p> <p><b>Lenguaje de programación:</b> JavaScript, TypeScript</p> <p><b>Estructura:</b> React native CLI version 18.2.0, Android Studio</p> <p><b>Dispositivos:</b> Dispositivo físico: SM-A135M/DS versión de Android: 13; Emulador: Pixel_3a_API_33_x86_64, Versión de Android: 13</p> <p><b>Motor Base de Datos:</b> API Rest</p>

Fuente: Propia autoría

### Requerimientos del Hardware

Se lista a continuación los requisitos del hardware que permitirán el buen funcionamiento de la aplicación

*Tabla 2: Requerimientos del hardware*

Nombre de la aplicación	Migración de la aplicación móvil Sion Estudiantes de la Corporación Universitaria Adventista
Herramientas	<p><b>Sistema Operativo:</b> Android, mínimo V.4.0</p> <p><b>Dispositivo Móvil:</b> Smartphone</p> <p><b>Conectividad:</b> WiFi 2G, 3G, 4G, LTE. 802.11 b/g/n</p> <p><b>Memoria RAM:</b> mínimo 126 MB.</p> <p><b>Touchscreen:</b> Capacitiva Multitouch 2 puntos.</p> <p><b>Resolución de la pantalla:</b> mínimo 320 x 480 píxeles</p> <p><b>Java:</b> Compatible</p>

*Fuente: Propia autoría*

### **3.1.4 Fase 4: Desarrollo y adaptación**

Después de determinar los requerimientos tanto del software como del hardware y tener clara la funcionalidad de la aplicación, se procede a generar la implementación de la migración, este proceso implica una programación minuciosa y limpia que refleje los requisitos y especificaciones de la nueva arquitectura, además se genera constantemente pruebas de compatibilidad y rendimiento con el fin de garantizar que dicha aplicación no solo sea funcional sino también ágil y eficiente, se obtiene errores a través de la depuración de código lo que permite analizar y corregir posibles problemas que puedan surgir en el desarrollo.

En esta fase se crea la lógica para validar, enviar y recibir toda la información que corresponde a los datos del estudiante, también se generan protocolos seguros para la recepción y transmisión con el fin de proteger los datos de los usuarios, además del proceso de lógica para los datos, se desarrolla la interfaz con la cual el usuario podrá interactuar y consumir las funcionalidades de esta.

### **3.1.5 Fase 5: Pruebas y depuración**

En esta fase se llevan a cabo las pruebas para evaluar la velocidad y capacidad de respuesta de la aplicación, se generan pruebas donde una alta demanda de usuarios ingrese al

aplicativo e interactúen con la misma, de esta manera se determina que tan lenta puede llegar a ser o experimentar errores con los tiempos de respuesta.

También se deben implementar pruebas de seguridad, esto debido a que se manejan datos sensibles los cuales deben ser protegidos, determinando así que no haya brechas de seguridad o amenazas de ciberseguridad.

Las pruebas de usabilidad también se deben tener en cuenta ya que con estas se determina que tan efectiva será la experiencia de usuario, con estas pruebas se evalúa la facilidad de uso, la navegación y la eficiencia en la interfaz.

### **3.1.6 Fase 6: Ajustes**

Teniendo en cuenta la fase anterior, es posible que se identifiquen posibles errores, problemas de rendimiento, seguridad o usabilidad, por lo que esta fase se centra en analizar estos hallazgos y dar solución a cada uno de ellos. Dentro de estos ajustes pueden surgir diversos aspectos como, implementar soluciones que fortalezcan la seguridad del aplicativo, el cambio del código fuente, mejorar la velocidad y eficiencia, si el aplicativo necesita características adicionales, se puede realizar ajustes en la lógica de la aplicación, además de hacer los ajustes necesarios en la documentación, para mantenerla actualizada y reflejar cada uno de los cambios realizados durante esta fase.

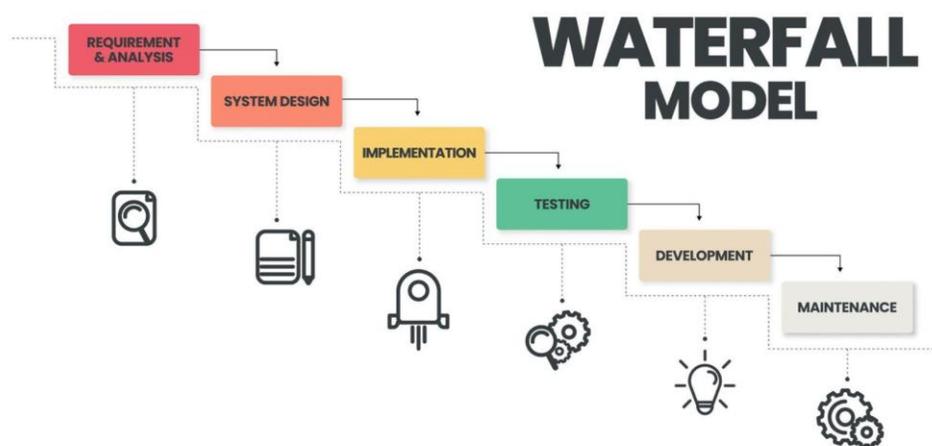
### **3.1.7 Fase 7: Cierre del proyecto**

En esta etapa final, se hace una revisión de todo el proyecto para asegurar que se haya cumplido con los objetivos planteados, se genera un documento donde se plasme todo el desarrollo y se completa la entrega final.

## CAPÍTULO 4 Resultados y Análisis de Resultados

La metodología seleccionada para el proyecto de desarrollo fue la “Metodología en Cascada”, la cual tiene un enfoque en desarrollar proyectos de forma secuencial y estructurado, se adapta a proyectos donde los objetivos se pueden definir claramente desde el principio y se caracteriza por desarrollar el proyecto paso a paso, es decir, cada etapa debe completarse antes de continuar a la siguiente etapa, empezando por las fases de análisis y diseño y finalizando con las de testeo y puesta en producción.

*Ilustración 1: Metodología en Cascada*



*Fuente: (Coworkingfy, 2020).*

### 4.1 Resultados

Este proyecto de migración se ha enfocado principalmente en el desarrollo de la interfaz y funcionalidades actuales de la aplicación móvil, además de integrar los datos de los estudiantes recibidos a través de un API Rest y poder visualizarlos en cada vista. Los resultados de este proyecto sientan las bases para futuros desarrollos implementando el framework de React native.

Para determinar las funcionalidades más importantes del SVGA se tomó como referencia el Visual Story Map (VSM) y el Android Application Package (APK) creado por los desarrolladores de la primera aplicación, a partir de este se crea el siguiente VSM con las funcionalidades a desarrollar en la migración, las cuales son:

usuarios: se gestiona el acceso al aplicativo a través de una pantalla de Login y se validan las credenciales

Cursos: Se gestiona la visualización de cursos matriculados por el estudiante.

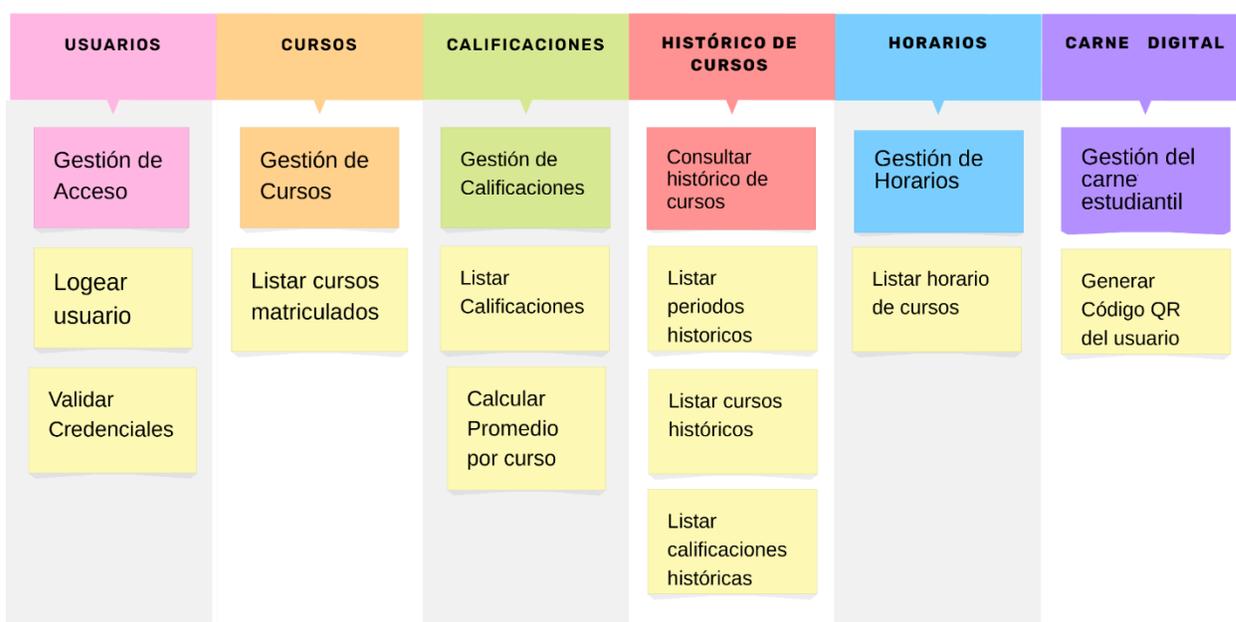
Calificaciones: Se gestiona las calificaciones hechas y se calcula el promedio por cada curso matriculado

Histórico de cursos: Se listan los periodos y los cursos del estudiante según su recorrido académico en la universidad, cada curso con sus respectivas calificaciones.

Horarios: Se listan los horarios de los cursos matriculados en el periodo actual

Carné digital: Se genera un código QR ligado al documento de identificación del usuario.

*Ilustración 2: Visual Story Map*



*fuelle: Propia autoría*

A continuación, se muestra la interfaz de usuario por cada funcionalidad implementada.

En esta interfaz de inicio de sesión, comúnmente llamada “Login” o “Pantalla de inicio” el usuario podrá ingresar sus credenciales las cuales les permitirán acceder al sistema, para ello deberá ingresar con el nombre de usuario y contraseña de la plataforma Sion, además de ingresar el código de estudiante.

Ilustración 3: Inicio de Sesión



2:56 29%

**Iniciar Sesión**

**Bienvenido de Nuevo**  
Ingrese las credenciales del sistema SION y el código estudiantil

Código Estudiantil

Usuario SION

Contraseña SION

**Iniciar**

Copyright 2007 - 2020 | Sistema de información Organizacional SION | Todos los derechos reservados.

*Fuente: Propia autoría*

El usuario al momento de interactuar con la aplicación al dar clic en el botón “iniciar” la aplicación realizará una serie de validaciones automáticamente las cuales garantizaran una autenticación segura. A continuación, se detallan las validaciones que proporciona la aplicación:

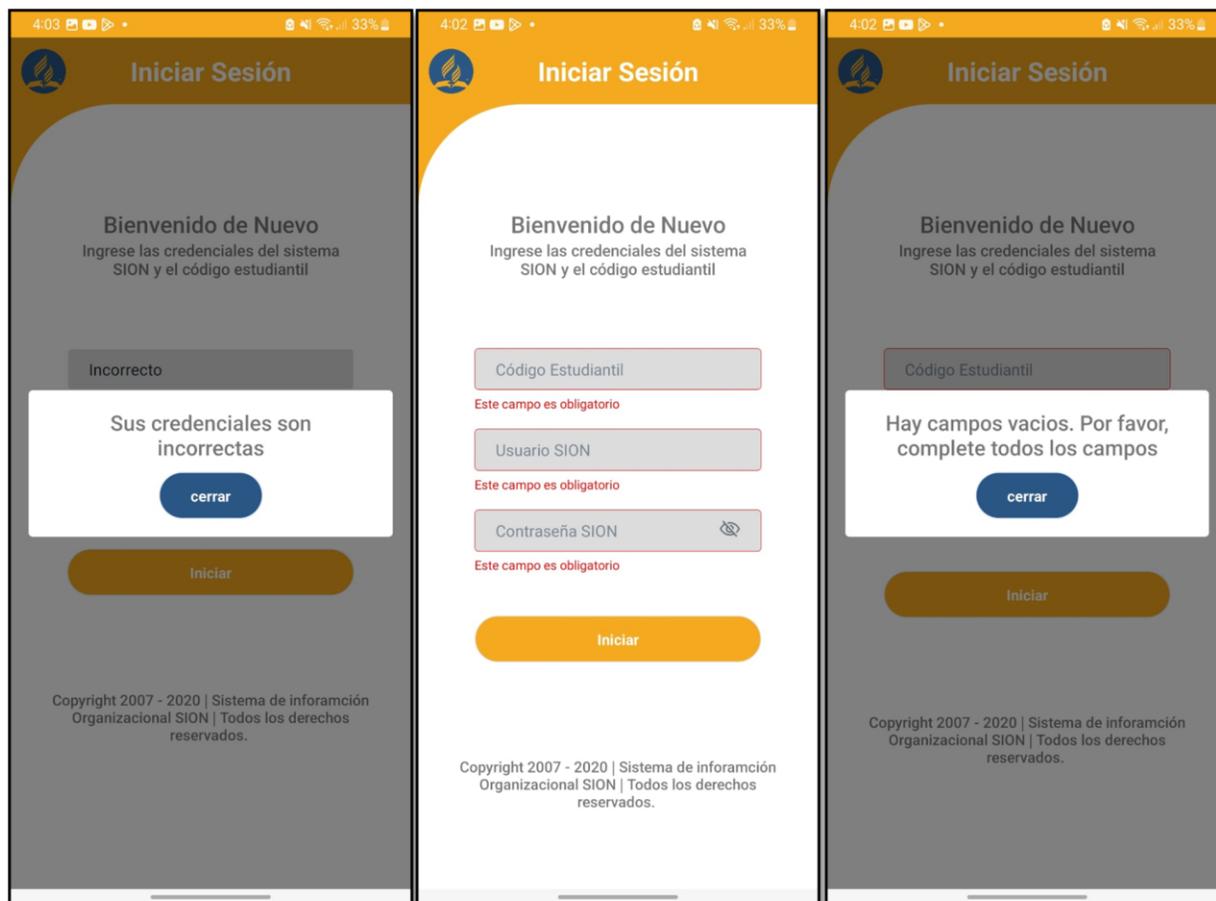
**Validación de campos vacíos:** Si el usuario no ingresa información en uno de los campos o en ninguno de ellos, el sistema entra en acción y en lugar de permitir el acceso, se activan mensajes informativos debajo de cada campo vacío, con el fin de que el usuario se dé cuenta y corrija el error al instante, sin embargo, si el usuario aun así deja los campos sin rellenar el sistema genera una alerta que brinda información sobre la presencia de errores.

**Validación de credenciales incorrectas:** Si el usuario ingresa alguno de los campos de forma incorrecta, como el usuario o contraseña incorrectos, el aplicativo nuevamente generará

una alerta informando las causas del acceso denegado, por lo que el usuario deberá hacer la corrección de sus credenciales.

Validación de red y servicios: En caso de que el usuario no disponga de acceso a Internet o si el servicio que proporciona la información a cada estudiante se encuentra inactivo, el sistema mostrará una alerta para notificar el problema.

*Ilustración 4: Validaciones*

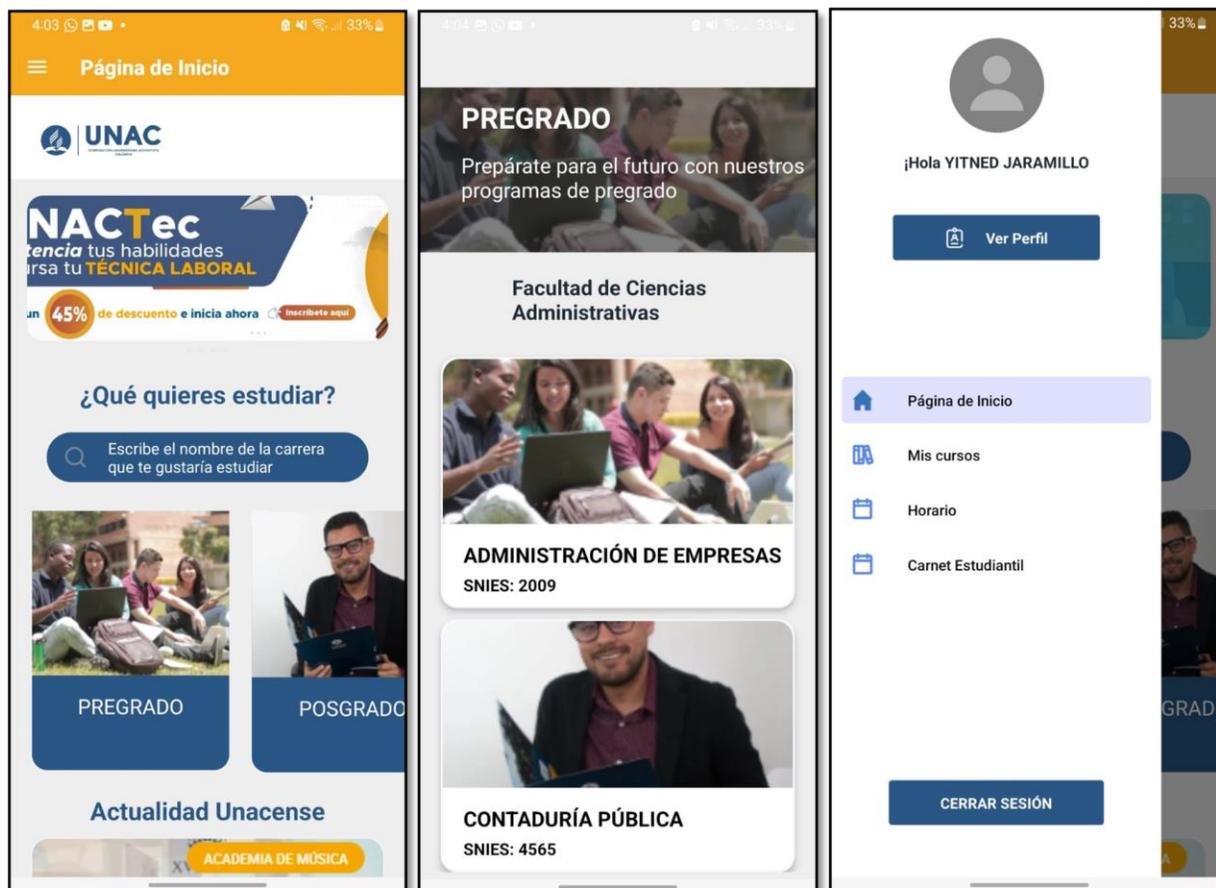


*Fuente: Propia autoría*

Una vez el usuario ingrese de forma exitosa al sistema, se redirigirá a una nueva interfaz, donde podrá ver la cartelera de la universidad, esto incluye anuncios, eventos, noticias y métodos de estudio, una vez allí si el usuario desea ver más información sobre los módulos presentados y obtener información detalla, podrá dar un clic en cada módulo y ser redirigido a la página principal de la Universidad o a la sección específica, además en la parte superior izquierda, el

usuario podrá ingresar a través de un menú lateral a diferentes opciones, como: Ver perfil, Cursos, Horarios, Carnet Digital y Cerrar sesión.

*Ilustración 5: Pantalla principal*



*Fuente Propia autoría*

El menú lateral proporciona información importante y acceso rápido, lo que permite una interacción más efectiva y agradable al usuario, en cada una de las opciones mencionadas anteriormente se visualizará información y funcionalidades diferentes, estas opciones proporcionan contenido que el usuario puede esperar, a continuación, se detalla cada una de las opciones.

**Perfil:** Cuando el usuario de clic en la opción ver perfil se redirige a una nueva pantalla donde se mostrarán datos importantes sobre sí mismos, estos datos incluyen:

**Número de identificación:** Número único que identifica al usuario en el sistema.

**Correo electrónico:** Dirección de correo electrónico asociado a la cuenta estudiantil.

Código estudiantil: Número único asignado por la Universidad.

Nombre completo: Nombre completo del usuario tal como aparece en el registro.

Programa al que pertenece: Programa académico (facultad) al que el usuario pertenece.

Es importante mencionar que en esta pantalla el usuario no tiene permisos para modificar ninguno de sus datos, en caso de ser necesario hacer cambios o actualización de su información personal, deberá dirigirse a la página de Sion, donde podrá hacer las modificaciones de forma segura y con precisión.

*Ilustración 6: Perfil de usuario*

←



Número de documento  
CC - 107863[REDACTED]

Nombre Completo  
Yitned Alejandra Jaramillo Valderrama

Código Estudiantil  
2019103[REDACTED]

Programa  
Ingeniería de Sistemas

Correo UNAC  
yitneda.jamillov@unac.edu.co

*Fuente: Propia autoría*

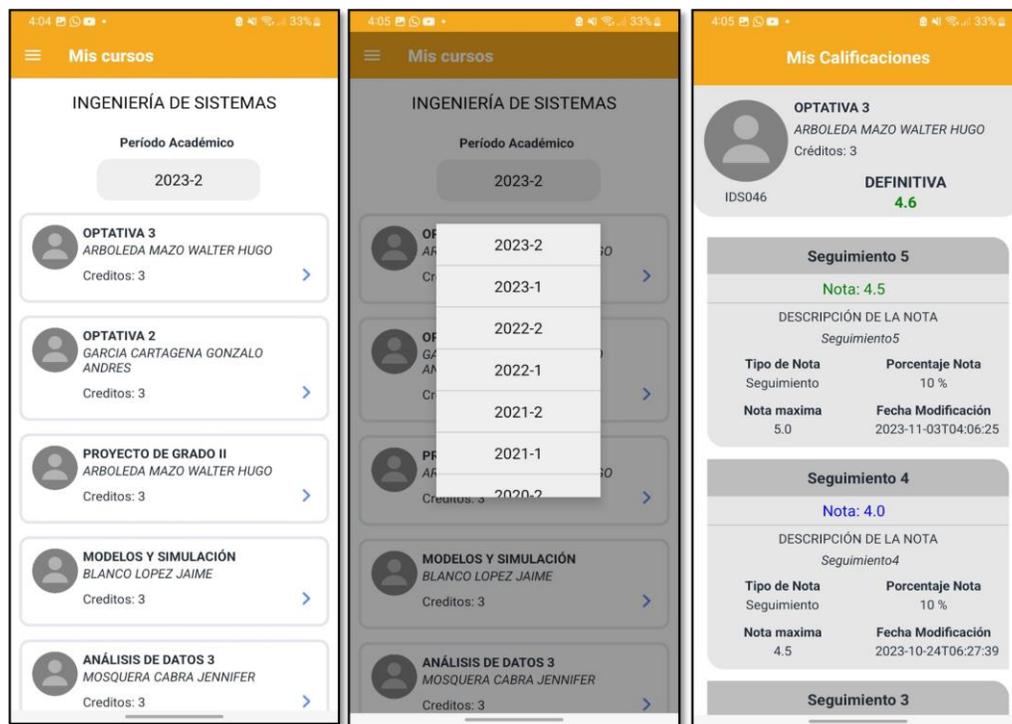
Cursos: cuando el usuario da clic en esta opción se redirige a una nueva pantalla donde se muestra información académica esencial y calificaciones detallada, una vez estando allí, el usuario podrá ver un listado de los cursos a los que esté matriculado, cada curso incluye

información referente al mismo como: Nombre de la materia, profesor que dicta la materia, y créditos.

Además, también tendrá la capacidad de acceder a cursos anteriores organizados por periodo académico, brindando la oportunidad de revisar su historial a lo largo de su trayectoria universitaria, incluyendo la misma información de cursos y calificaciones mencionada anteriormente.

Cuando el usuario da clic en un curso específico, el sistema validará la petición, si hay calificaciones mostrará un mensaje “Cargando calificaciones...” con el fin de dar un tiempo para que las calificaciones se carguen correctamente, de lo contrario si no hay calificaciones mostrará el mensaje “No hay calificaciones disponibles”, una vez hecha la validación el usuario podrá ver sus calificaciones, en el encabezado encontrará información referente al curso en general: nombre de la materia, nombre del profesor, créditos y el promedio general; en el cuerpo de la pantalla se encontrará la lista de calificaciones hechas por el docente con información como: Nombre de la calificación, puntuación obtenida en la calificación, tipo de nota, descripción de la nota, porcentaje de la nota, nota máxima y fecha de modificación.

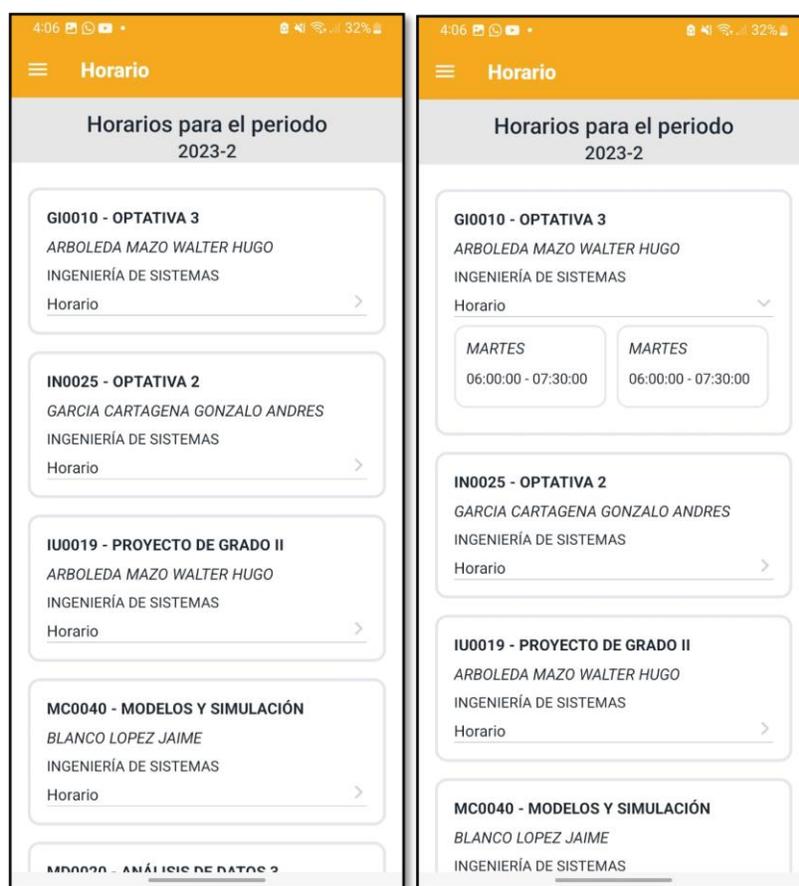
*Ilustración 7: Cursos y calificaciones*



*Fuente: Propia autoría*

**Horario:** Cuando el usuario da clic en la opción de horario, se abrirá una nueva pantalla mostrando los horarios según los cursos matriculados, en el encabezado se visualizará el programa al cual pertenece el usuario y el semestre en el que se encuentra, en el cuerpo de la pantalla se encuentra una lista de cada uno de los horarios, este tiene nombre de la materia, nombre del docente, créditos y posteriormente se encuentra el día y hora en que se dictará la materia. De esta forma el usuario podrá tener una visión organizada y detallada de su programación académica, lo que les permitirá mantener un seguimiento efectivo de sus clases y actividades.

*Ilustración 8: Horarios*

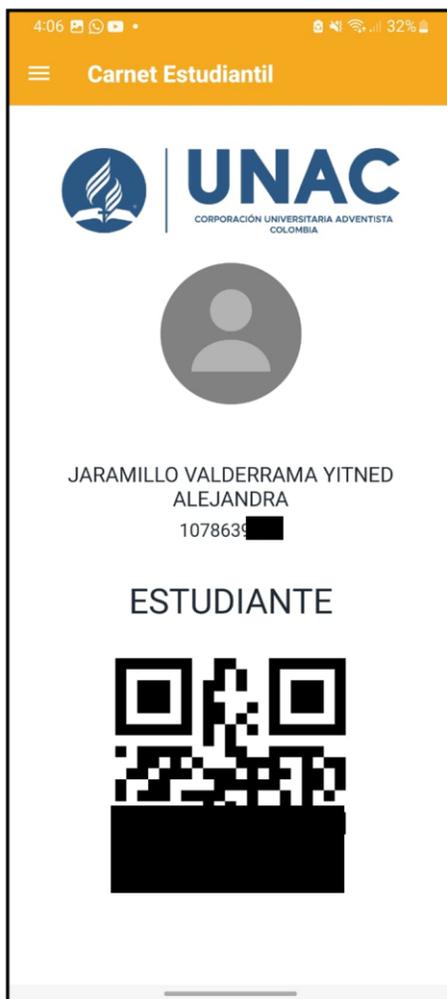


*Fuente: Propia autoría*

**Carnet estudiantil:** Cuando el usuario da clic en esta opción, el sistema lo redirigirá a una nueva pantalla, donde se mostrará información del estudiante, como su nombre, número

de identificación y su rol en la universidad, además encontrará también un código QR el cual le permitirá registrarse y acceder a las instalaciones de la universidad.

*Ilustración 9: Carné estudiantil*

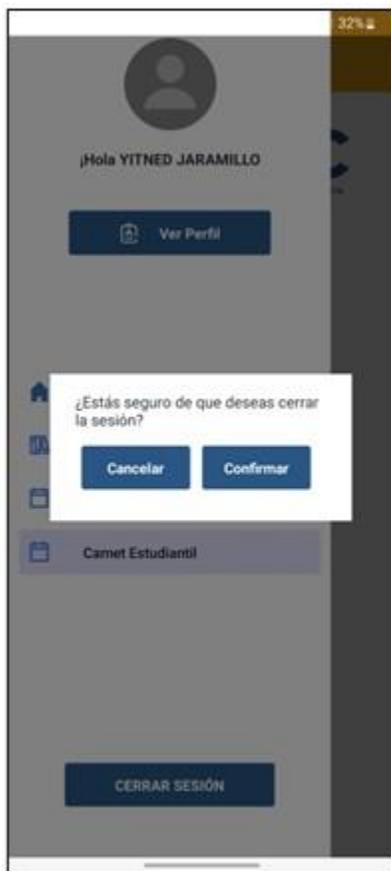


*Fuente: Propia autoría*

Cerrar sesión: Esta última parte, está diseñada con un enfoque de seguridad, es decir, cuando el usuario de clic en la opción “Cerrar Sesión”, el sistema verifica y alerta al usuario sobre la decisión tomada, mostrando así un PopUp que pregunta si realmente está seguro de terminar la interacción por el aplicativo. Esta capa de seguridad y confirmación brinda al usuario la oportunidad de considerar su decisión, presentando en la alerta dos opciones claras: “Cancelar” y “Confirmar” si el usuario decide cancelar la operación, el sistema permite al usuario seguir utilizando el aplicativo. Sin embargo, si el usuario confirma su intención de cerrar sesión, el

sistema actúa de manera inmediata y redirige al usuario a la pantalla de inicio de sesión, borrando tras esta operación las credenciales guardadas al inicio, asegurando así el cierre de sesión exitoso.

*Ilustración 10: Cerrar sesión*

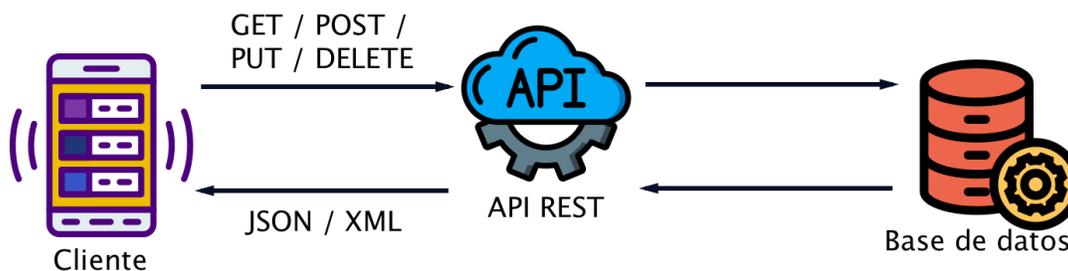


*Fuente: Propia autoría*

Cada institución tiene bajo su responsabilidad, los diferentes datos de todo el personal que hace parte de la organización, estos datos deben ser guardados y protegidos para no vulnerar los derechos de los usuarios tal como se determinó en el marco legal, es por esto que la universidad en favor de asegurar los datos de sus estudiantes y demás personal cuenta con un servidor donde se almacena dichos datos y se protegen de vulnerabilidades de seguridad o brechas informáticas.

Este servidor es manipulado única y exclusivamente por el DSI, por lo que esta área debió proporcionar para el cumplimiento de este proyecto un API Rest con el fin de consumir los datos y utilizarlos en el Frontend de la aplicación.

Ilustración 11: API Rest



Fuente: (DosSetenta, 2021).

Como se observa en la imagen anterior, un API Rest es el intermediario entre la base de datos y el cliente, a través de esta se hacen peticiones de tipo “http” para mandar o recibir información directamente de la base de datos. Es necesario decir que los datos tanto a enviar como a recibir son en formato json.

A continuación, se muestran los datos recibidos y el método que se utilizó

Para gestionar el inicio de sesión, el método utilizado es “POST” esto significa que el API recibirá los datos enviados a través de la aplicación y hará la validación con la base de datos, si el código de estado de respuesta “http” es 200, significa que las credenciales son válidas y correctas, desde este punto se hace la validación en el Frontend para permitir que el usuario acceda al sistema.

El código 400 me advierte que hay un problema en la forma en cómo el usuario envía las credenciales, es decir, a través de este código se maneja la respuesta de credenciales incorrectas. Si por el contrario el código es 500, significa que las credenciales ingresadas no se encuentran registradas en la base de datos, se maneja la respuesta y se le envía la alerta al usuario.

De esta manera se valida el ingreso del usuario al aplicativo, así como toda su navegación en la misma.

*Ilustración 12: Datos de autenticación en API Rest*

```
--header 'Content-Type: application/json' \
--data '{
  "username": "string",
  "studentCode": "string",
  "password": "string"
}'
```

*Fuente: Propia autoría*

Cuando el usuario ingresa sus credenciales, asociado a su código estudiantil se obtienen todos los datos que serán consumidos en el aplicativo, entre ellos perfil, cursos, calificaciones y horarios. todos estos a través del método “GET” que como ya se ha dicho antes, sirve para obtener información desde la base de datos.

*Ilustración 13: Datos de perfil con API Rest*

#### Example Response

```
{
  "documento": "10068**",
  "tipoDocumentoSiglas": "CC",
  "nombre": "NOMBRE COMPLETO DEL ESTUDIANTE",
  "codigo": "202010*****",
  "programa": "CONTADURÍA PÚBLICA",
  "correoUnac": "example.example@unac.edu.co",
  "nombreCorto": "NOMBRE CORTO DEL ESTUDIANTE"
}
```

*Fuente: Propia autoría*

## 4.2 Análisis de resultados

Este apartado tiene como objetivo analizar los detalles de la migración de la aplicación móvil “Sion Estudiantes” la cual tuvo lugar a inicios del presente año. De acuerdo con los resultados obtenidos y la investigación hecha durante el proceso de desarrollo se genera el siguiente análisis:

La migración de esta aplicación implicó hacer una actualización de la plataforma nativa Xamarin Android a la plataforma React Native, por lo que se debió hacer una lectura de comprensión de ambas plataformas y un análisis profundo del código existente, con el fin de entender el motivo por el cual se pretendía hacer la migración.

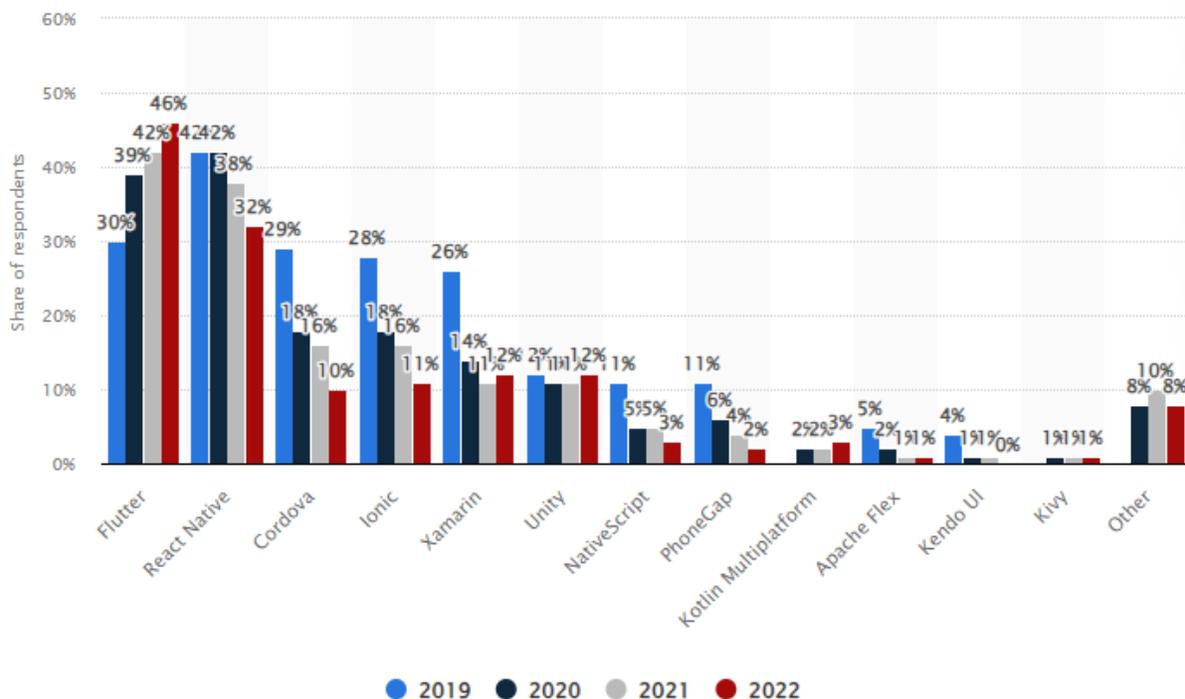
Dado que el proyecto se basa principalmente en la plataforma React native, y teniendo en cuenta que las personas desarrolladoras de este proyecto no tenían conocimiento acerca de dicho framework, se procede a hacer investigación exhaustiva y a comprender los lineamientos, reglas y factores que implica hacer un desarrollo en esta plataforma.

Investigando acerca de esto, se encontró, según estadísticas (ver Figura 13) sobre las respuestas de los usuarios a la mejor plataforma, React Native entre los años 2019 y 2022 se posicionó como uno de los mejores frameworks de código abierto para el desarrollo de aplicaciones multiplataforma (Sujay, 2023), esto debido a que se puede generar aplicaciones tanto para Android como iOS, es una combinación del desarrollo nativo de React y la biblioteca JavaScript, lo que permite crear interfaces de usuario, Se puede utilizar en proyectos que estén en construcción o crearlos desde cero, los cambios que se ejecuten se pueden visualizar tan pronto como se guarden, tiene funciones que se renderizan en la interfaz, entre muchas más cualidades (Bambu, 2022).

Después de indagar un poco más sobre esta plataforma y entender lo que el DSI requería para la migración, se procedió a adquirir conocimiento integral relacionado con la creación y manipulación de código, implementando desde métodos y componentes hasta la creación de funcionalidades más avanzadas, utilizando diferentes librerías y herramientas disponibles. Ese aprendizaje se lleva a cabo a través de diversos recursos educativos, en los cuales se incluyeron la consulta de documentación, la exploración de tutoriales en línea y el aprovechamiento de técnicas de Inteligencia Artificial (IA).

A su vez se iba observando y analizando con detenimiento el código de la aplicación existente, con el fin de obtener los requisitos funcionales y los datos a requerir en la generación de la API.

Ilustración 14: Marcos móviles multiplataforma



Fuente: (Sujay, 2023).

La siguiente etapa involucró la revisión y comparación entre el aplicativo y el código subyacente. Aunque, inicialmente los resultados arrojados después de la evaluación técnica al aplicativo fueron positivos, se descubrió que había diferencias significativas en la documentación técnica existente, específicamente en el VSM, el cual fue definido previamente por desarrolladores anteriores.

Este VSM, quien cumple un papel importante en la comprensión de los requisitos del sistema, reveló que ciertas funcionalidades no se encontraban reflejadas en la aplicación móvil actual. Entre las cuales se encuentran:

Recuperar contraseña: Según esta funcionalidad, la aplicación debería proporcionar un apartado donde el usuario pueda ingresar y recuperar su contraseña si es necesario. Sin embargo, en el aplicativo no se percibe.

Notificación de nuevas calificaciones: De acuerdo con el VSM, esta funcionalidad debería ser implementada al momento que un docente ingrese una nueva calificación. Sin embargo, esta característica no fue encontrada en el aplicativo.

Listado de clases y ausencias: Según el VSM esta funcionalidad lista las clases y a su vez lleva un registro de las ausencias de los estudiantes, en la revisión no fue encontrada.

Cálculo del límite de ausencias: otra funcionalidad registrada en el VSM fue la capacidad de calcular el límite de ausencias permitidas a los estudiantes, a pesar de ello, esta funcionalidad se encuentra ausente.

Notificación de ausencias: Este registro indica que, cuando un usuario estuviera a punto de cumplir el máximo de sus ausencias, debería recibir una notificación. Sin embargo, no se encuentra implementada en dicho aplicativo.

Registro y evaluación de docentes: Se menciona en el VSM la funcionalidad de registrar las evaluaciones que los estudiantes hicieran a los docentes, pero esta no es visible en la aplicación.

Listado de horarios de parciales y finales: Finalmente, se identificó que la funcionalidad que permite a los estudiantes estar al tanto de los horarios de parciales y exámenes finales, tampoco se encuentra implementada en el aplicativo.

Una vez completado el análisis comparativo, se procedió a la creación del VSM que se implementaría en el desarrollo de este proyecto. Este modelo se presenta en la Figura 2, y según este modelo, se planteó una estrategia para cada funcionalidad, detallando la forma y los recursos a utilizar con el fin de lograr un rendimiento óptimo. A continuación, se describen las técnicas y herramientas utilizadas para monitorear y mejorar el rendimiento del proyecto.

FlatList en lugar de ScrollViews para listas de mayor volumen: Para la gestión de listas extensas, se optó por emplear FlatList en lugar de ScrollViews. Esta elección se basa en la eficacia que tienen los FlatList para proporcionar un rendimiento superior al cargar y mostrar datos en listas largas. Esto garantiza una experiencia de usuario más fluida y eficiente al desplazarse por contenido extenso. (React Native, 2023)

React Native Reanimated para animaciones: Para lograr animaciones suaves y eficientes, se implementó la biblioteca React Native Reanimated. Esta herramienta cuenta con capacidad para generar animaciones de alto rendimiento, lo que contribuye a una experiencia de usuario más atractiva y eficaz. (NPM, 2023)

React.memo() para optimización del renderizado: Cada funcionalidad se encapsuló en React.memo(), lo que simplifica y optimiza el proceso de renderizado. Esta técnica permite que los componentes se actualicen sólo cuando cambian sus propiedades, lo que reduce la carga en el sistema y mejora la eficiencia del proyecto en general. (React, 2023)

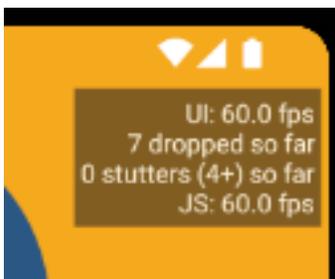
Performance Monitor para seguimiento de rendimiento y fallos: Se incorporó una herramienta de monitoreo de rendimiento, como Performance Monitor, para mantener un registro continuo del desempeño del proyecto. Esto facilita la identificación de posibles problemas de rendimiento y fallos, lo que permite abordarlos de manera oportuna y eficaz.

Fetch para solicitudes eficientes: Se eligió la función 'fetch' para gestionar las solicitudes de datos debido a que tiene la capacidad para minimizar el número de solicitudes innecesarias y reducir el tamaño de las respuestas, lo que a su vez mejora la velocidad de carga de datos y reduce la carga en el servidor.

React Native DevTools: Esta es una extensión específica para React Native que se integra con React DevTools. Esta permite visualizar el árbol de componentes y organizarlos, permite inspeccionar el estado de los componentes y comprender cómo interactúan entre ellos, esta extensión además tiene una funcionalidad que permite grabar y mostrar la duración de ciertas fases durante la renderización de un componente.

Se muestra a continuación algunas métricas de rendimiento tomadas de las herramientas Performance Monitor y React Native DevTools:

*Ilustración 15: Métricas de rendimiento con Performance Monitor*



*Fuente: Propia autoría*

En la anterior imagen se muestran varias métricas que determinan el rendimiento de la interfaz de usuario, en primer lugar, se obtiene un "UI de 60.0 fps" esto representa la velocidad de los fotogramas por segundo, es decir, la velocidad a la que la interfaz de usuario de la aplicación se actualiza o renderiza en la pantalla. En este caso, "60.0 fps" significa que la interfaz de usuario se actualiza a una velocidad de 60 fotogramas por segundo.

Un alto número de fps, como 60 fps, indica un rendimiento óptimo de la interfaz de usuario, esto quiere decir que la aplicación se actualiza suavemente en la pantalla, lo que

proporciona una experiencia de usuario fluida y receptiva. En general, se considera que 60 fps es una tasa de actualización deseable para la mayoría de las aplicaciones, ya que ofrece una experiencia de usuario de alta calidad. (Fernández, 2020)

La línea "7 dropped so far" hace referencia a los fotogramas omitidos, en este contexto, cuando los fotogramas se omiten, significa que la aplicación no se está actualizando tan rápidamente como debería. Lo que conlleva a que la aplicación no sea tan fluida.

La siguiente línea "0 (cero) stutters (4+) so far": Significa que ha habido 0 (cero) momentos en los que la aplicación se ha entrecortado o ha tenido problemas de rendimiento. Los números entre paréntesis "(4+)" sugieren que al menos 4 de esos momentos fueron particularmente notables en término de problemas de rendimiento.

Por último, la línea "js: 60.0 fps" indica que, a pesar de los problemas anteriores, la velocidad de fotogramas de la parte JavaScript de la aplicación es de 60.0 fps, lo cual es una tasa de actualización óptima. Esto sugiere que, a nivel de JavaScript, la lógica de la aplicación se está ejecutando a una velocidad adecuada.

Para manejar los problemas de rendimiento y evitar que se omitan fotogramas, se utilizaron técnicas como 'async' y 'await' para gestionar tareas asíncronas, con el fin de que estas se ejecuten en segundo plano y no bloquear el hilo principal, además como ya se ha dicho anteriormente, se utilizó React.memo() para renderizar solo las partes donde surgen actualizaciones y no todo el componente.

#### *Ilustración 16: Tareas asíncronas*

```
const Login_Service = async (username, studentCode, password) => {  
  const response = await fetch("http://apivestudiante.unac", {  
    method: "POST",  
    headers: {
```

*Fuente: Propia autoría*

La herramienta React Native DevTools también proporciona información relevante que ayuda a determinar qué áreas pueden requerir optimización y mejorar el rendimiento.

Ilustración 17: Métricas de rendimiento con React Native DevTools

The screenshot shows the React Native DevTools performance profiler. On the left, a component stack is listed: `RootComponent (Sion_Estudiantes)`, `PerformanceLoggerContext.Provider`, `AppContainer`, `RootTagContext.Provider`, `View`, `TextAncestorContext.Provider`, `View key="1"`, `TextAncestorContext.Provider`, `App`, and `UserProvider`. On the right, the following metrics are displayed:

- Priority:** Immediate
- Committed at:** 1.5s
- Durations:**
  - Render:** 31.3ms
  - Layout effects:** 0.5ms
  - Passive effects:** 4.1ms
- What caused this update?** Wrapper

Fuente: Propia autoría

La información presentada se obtiene de la interfaz de usuario “Login o Inicio de Sesión” la cual muestra la duración de ciertas fases durante la renderización del componente. Se detalla cada elemento a continuación:

“Priority Immediate” indica la prioridad de la tarea. “Inmediatamente” sugiere que la tarea tiene una alta prioridad y se realizó de manera inmediata, sin demoras notables.

“Committed at: 1.5s” indica cuando se completó la tarea en segundo. En este caso la tarea se completó a los 1.5 segundos después de comenzar la grabación, esto es útil para identificar el momento en que se ejecuta una tarea en relación con otras secciones en la aplicación.

Durations: a continuación, se desglosa la duración de diferentes fases de la renderización

“Render: 31.3ms” es la duración del proceso de generar la representación visual del componente en la vista. En este caso, la fase de renderización llevó 31.3 milisegundos, lo que se puede considerar como el tiempo que lleva construir la interfaz de usuario del componente.

“Layout effects: 0.5ms” son los efectos de diseño que se ejecutan después de que la fase de renderización se haya completado, y se encargan de cálculos relacionados con el diseño de la interfaz. En este caso los efectos de diseño tomaron 0.5 milisegundos.

“Passive effects: 4.1ms” son efectos pasivos que se ejecutan después de que los efectos de diseño se hayan implementado. Estos efectos suelen estar relacionados con tareas menos críticas que pueden ejecutarse de manera retrasada sin afectar significativamente el rendimiento de la interfaz de usuario.

Después de completar la creación de la nueva aplicación, se realizó un análisis de rendimiento comparativo con respecto a la versión anterior. Aunque no se disponía de métricas precisas de la aplicación anterior, se pudo observar una clara diferencia en el tiempo de carga y la eficiencia de la nueva aplicación en base a observaciones prácticas.

En la versión anterior de la aplicación, al ingresar a la página principal después de iniciar sesión, se experimentaba un notable retraso en la carga de imágenes y la visualización del carrusel, con un tiempo de aproximadamente 7 segundos. Por otro lado, la aplicación migrada, desarrollada siguiendo las mejores prácticas y técnicas de optimización mencionadas anteriormente, mostró una mejora significativa en el rendimiento. En este caso, la nueva aplicación cargaba y mostraba las imágenes y el carrusel de manera casi inmediata después de iniciar sesión. Se muestra a continuación, cómo se visualiza la pantalla principal del anterior aplicativo.

*Ilustración 18: Carga en pantalla principal*



*Fuente: Propia autoría*

Estos resultados prácticos evidencian que la nueva aplicación ha logrado una mejora en el rendimiento con respecto a su predecesora. La implementación de técnicas como el uso de

FlatList, React.memo(), Performance Monitor y otras optimizaciones, ha dado como resultado una experiencia de usuario más ágil y eficiente, lo que contribuye a un mayor nivel de satisfacción de los usuarios y al éxito general de la aplicación.

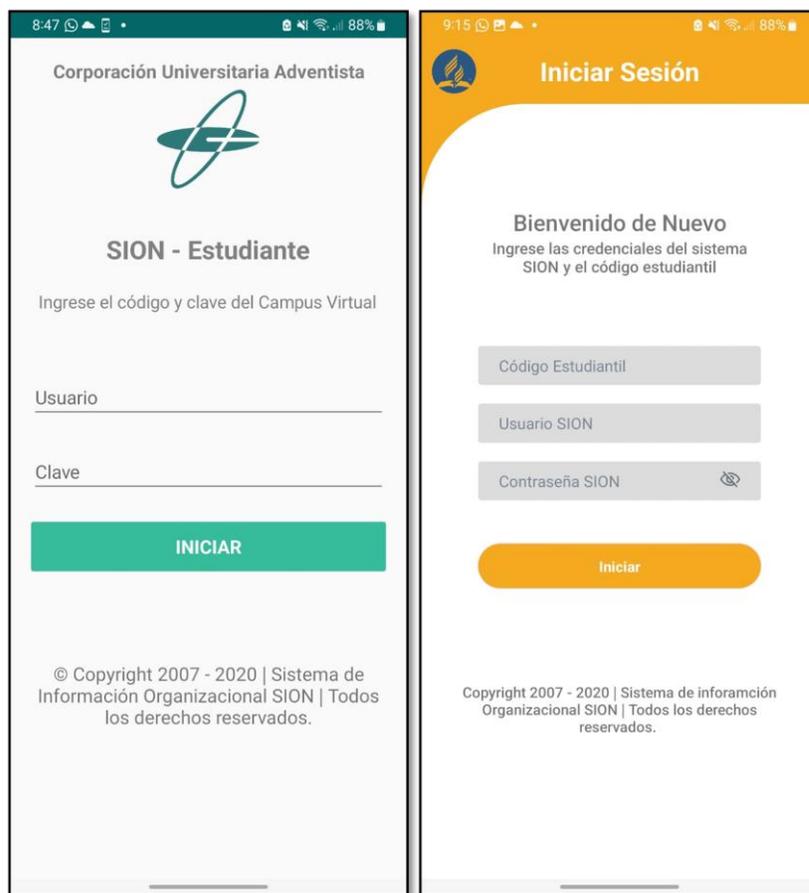
Finalmente, se procede a comparar las interfaces de ambas aplicaciones. Aunque ambas ofrecen las mismas funcionalidades esenciales, se han realizado cambios significativos en la interfaz de usuario para mejorar la experiencia de usuario.

En la Figura 19, se puede apreciar un diseño renovado que respeta la paleta de colores de la Universidad, lo que contribuye a una identidad visual más consistente y profesional. Se introdujo un nuevo campo para verificar y validar la información de los estudiantes, lo cual fortalece la seguridad y la precisión de los datos.

Además, se implementaron nuevas validaciones que no estaban presentes en la aplicación anterior. Ahora, el usuario recibe notificaciones sobre campos vacíos en el inicio de sesión y se le informa si la contraseña ingresada es incorrecta. También se agregó una función que solicita la confirmación del usuario antes de cerrar la sesión lo que evita cierres accidentales de sesión durante la navegación.

En el lateral izquierdo (drawer), se incorporó una opción para acceder al carné digital, proporcionando una funcionalidad adicional y útil. Además, se incluyeron mensajes informativos en los cursos para informar al usuario sobre la disponibilidad de calificaciones, lo que mejora la comunicación y facilita el acceso a información relevante.

Ilustración 19: Diferencias entre pantallas de inicio de sesión



Fuente: Propia autoría

## CAPÍTULO 5 Conclusiones y Recomendaciones

### 5.1 Conclusiones

De acuerdo con el objetivo planteado en este documento y la realización del proyecto, se llegan a las siguientes conclusiones:

**Análisis y diseño efectivo:** La comprensión de los elementos teóricos relacionados con la migración de aplicaciones móviles desempeñó un papel fundamental en la definición de la solución propuesta para abordar las necesidades específicas. La aplicación de la metodología en cascada proporcionó una estructura sólida para todo el proceso de desarrollo, lo que facilitó la elección de las herramientas adecuadas y culminó en una solución que se alineó con los requisitos iniciales.

**Migración exitosa de funcionalidades clave:** Aunque no se logró migrar completamente toda la aplicación, se logró implementar las funcionalidades más relevantes. Esto resultó en una herramienta que puede proporcionar a los estudiantes los recursos necesarios en función de su gestión académica. El monitoreo constante del rendimiento de la aplicación, con la ayuda de diversas herramientas, garantizó el funcionamiento adecuado de cada característica implementada y permitió abordar de manera efectiva cualquier error que surgiera durante la ejecución.

**Ventajas de React Native:** La elección de React Native como plataforma de desarrollo se destacó debido a su capacidad para crear aplicaciones multiplataforma. Aunque en este proyecto no se implementó la versión para iOS, se sientan las bases para lograrlo en el futuro. Esto ofrece un mantenimiento más eficiente y económico, permitiendo satisfacer las necesidades de todos los usuarios en distintos sistemas operativos.

**Crecimiento profesional y lecciones aprendidas:** Desde una perspectiva personal, la ejecución de este proyecto ha impulsado el crecimiento profesional tanto para los participantes como para el equipo de desarrollo. Las lecciones aprendidas y el conocimiento adquirido han resaltado la importancia de la planificación detallada, una gestión de riesgos efectiva y una comunicación constante y transparente. Además, han enfatizado el compromiso y la colaboración del equipo de desarrollo como elementos fundamentales para el éxito del proyecto.

En resumen, el proyecto de migración ha cumplido en su mayoría con los objetivos establecidos y ha proporcionado valiosas lecciones y recomendaciones para proyectos futuros. Las mejoras en el rendimiento y la implementación de nuevas funcionalidades respaldan la efectividad de la migración y allanan el camino para un mantenimiento proactivo y futuras

mejoras. El conocimiento adquirido y las prácticas destacadas son activos importantes para futuros proyectos de desarrollo de aplicaciones.

## 5.2 Recomendaciones

Las siguientes recomendaciones se presentan como puntos clave para enriquecer el aplicativo:

**Publicación del aplicativo:** Se sugiere la publicación de la aplicación para que los estudiantes puedan aprovechar los beneficios de esta. La disponibilidad temprana brinda acceso a las funcionalidades existentes y establece una base para futuras mejoras, además darle la importancia que realmente merece, esto implica destacar su utilidad y promover su adopción entre los estudiantes.

**Completar funcionalidades faltantes:** Se recomienda priorizar la finalización de las funcionalidades que aún no se han implementado en el aplicativo, para garantizar una mejor experiencia de usuario y generar valor agregado

**Agregar nuevas funcionalidades:** Se recomienda considerar la adopción de nuevas funcionalidades integrando la aplicación con diferentes departamentos, al ser una aplicación educativa, esta podría aliarse con la biblioteca Universitaria para proporcionar acceso a recursos académicos de forma conveniente. Además, generar roles para los estudiantes que son internos y externos, con la finalidad de mostrar contenido diferenciado, por ejemplo, brindando a los estudiantes internos la capacidad de ver el menú del restaurante e información proveniente de los hogares.

También una funcionalidad que agregaría mucho valor es la de poder ver el mapa de la Universidad y la locación actual, con el fin de que tanto antiguos como nuevos estudiantes se puedan ubicar de forma fácil en el campus.

**Mejoras en la interfaz:** Es recomendable hacer cambios en la interfaz que aporten y atraigan la interacción de los estudiantes, cambios mínimos pueden hacer la diferencia y conllevar a que la aplicación tenga un mejor auge, es necesario mencionar que se debe hacer seguimiento en el rendimiento y seguridad de la aplicación, generar un mantenimiento constante e informar a la población estudiantil cada que se haga un cambio significativo.

## 2. Referencias Bibliográficas

- Aguirre, S. (2022). *Crea tu API*. Libro. [https://books.google.com.co/books?hl=es&lr=&id=XxF1EAAAQBAJ&oi=fnd&pg=PP2&dq=que+es+una+API&ots=ssZijxA5UB&sig=a3TzDvK0LrGhOSyo2GxBYZzmC54&redir\\_esc=y#v=onepage&q=que%20es%20una%20API&f=false](https://books.google.com.co/books?hl=es&lr=&id=XxF1EAAAQBAJ&oi=fnd&pg=PP2&dq=que+es+una+API&ots=ssZijxA5UB&sig=a3TzDvK0LrGhOSyo2GxBYZzmC54&redir_esc=y#v=onepage&q=que%20es%20una%20API&f=false)
- AWS. (2023). *¿Qué es una API? - Explicación de interfaz de programación de aplicaciones - AWS*. <https://aws.amazon.com/es/what-is/api/>
- Bambu mobile. (2023, May 15). *¿Qué es Xamarin?* <https://www.bambu-mobile.com/que-es-xamarin/>
- Coworkingfy. (2020). *Metodología Waterfall | Cómo aplicar la gestión de proyectos en cascada*. <https://coworkingfy.com/metodologia-waterfall/>
- Daza, L., Arboleda, W., & Ruiz, F. (2015). *Aplicacion Movil Para Android del Sistema Virtual de Gestion Academica de la Corporacion Universitaria Adventista*. Tesis. <http://repository.unac.edu.co/bitstream/handle/11254/200/Trabajo%20de%20grado.pdf?sequence=1&isAllowed=y>
- Deloitte. (2023). *¿Qué es React Native? | Tecnología | Deloitte España*. Artículo. <https://www2.deloitte.com/es/es/pages/technology/articles/que-es-react-native.html>
- DosSetenta. (2021). *¿Qué es una api rest? ¿Cómo funciona? ¿En que tipo de web utilizarlas?* <https://dossetenta.com/que-es-una-api-rest/>
- Enriquez, J. G., & Casas, S. I. (2014, June 11). *Vista de Usabilidad en aplicaciones móviles*. Artículo. <https://publicaciones.unpa.edu.ar/index.php/ICTUNPA/article/view/581/560>
- Fernández Y. (2020). *Cómo saber en tiempo real la tasa de refresco o los FPS de tu móvil Android*. <https://www.xataka.com/basics/como-saber-tiempo-real-tasa-refresco-fps-tu-movil-android#:~:text=Por%20ejemplo%2C%20si%20tu%20m%C3%B3vil,soltura%20alg%C3%BAAn%20juego%20o%20aplicaci%C3%B3n>

- Gellibert Merchán, S. J., Zapata Mora, S. E., & Díaz Vera, J. P. (2021). *Vista de Las TIC en la educación superior durante la pandemia de la COVID-19*. Revista Sinapsis. Vol 1, Nro 19. <https://revistas.itsup.edu.ec/index.php/sinapsis/article/view/405/802>
- Harvard Gazette. (2013, January 25). *Harvard Mobile expands*. CAMPUS & COMMUNITY. <https://news.harvard.edu/gazette/story/2013/01/harvard-mobile-expands/>
- Herazo, L. (2022). *¿Qué es una aplicación móvil? – Anincubator*. <https://anincubator.com/que-es-una-aplicacion-movil/>
- Hughes, O. (2019, October 25). *Digital leaders debate best of breed approach to NHS IT*. <https://www.digitalhealth.net/2019/10/digital-leaders-debate-best-of-breed-approach-to-nhs-it/>
- Innovacion. (2022, June 21). *BBVA migra a la nube los servicios de atención al cliente*. <https://www.bbva.com/es/innovacion/bbva-migra-a-la-nube-los-servicios-de-atencion-al-cliente-para-mejorar-el-servicio/>
- Izrailevsky, Y., Vlaovic, S., & Meshenberg, R. (2016, February 11). *Netflix se muda a la nube*. <https://about.netflix.com/es/news/completing-the-netflix-cloud-migration>
- Jimenez, S. (2021). Desarrollo de aplicaciones para dispositivos móviles mediante plataformas de desarrollo cruzado. *Proyecto*. [https://oa.upm.es/70594/1/TFG\\_SERGIO\\_JIMENEZ\\_TAMAYO.pdf](https://oa.upm.es/70594/1/TFG_SERGIO_JIMENEZ_TAMAYO.pdf)
- Knox, J. (2019, October 28). *Introducing the New Stanford Mobile App | University IT*. <https://uit.stanford.edu/blog/introducing-new-stanford-mobile-app>
- Mena, Mo. (2023, March 30). *Gráfico: El mapa mundial de Android e iOS | Statista*. El Mapa Mundial de Android e IOS. <https://es.statista.com/grafico/29620/sistema-operativo-movil-con-la-mayor-cuota-de-mercado-por-pais/>
- Mineducación. (2020, November 30). *Protección de Datos Personales*. <https://www.mineducacion.gov.co/portal/micrositios-institucionales/Modelo-Integrado-de-Planeacion-y-Gestion/Data/387771:Proteccion-de-Datos-Personales>

NPM. (2023). *React-native-reanimated*. <https://www.npmjs.com/package/react-native-reanimated>

Pavan, B. (2012, December 13). *Facebook presenta aplicación nativa para Android*. <https://hipertextual.com/2012/12/facebook-presenta-aplicacion-nativa-para-android>

Proyecto Educativo De Programa. (2018). *Actualizacion Aprobada por el Consejo de Facultad de Teologia de la Corporacion Universitaria Adventista , Acta O14*. <http://repository.unac.edu.co/bitstream/handle/11254/1006/PEP%20Lic%20en%20Teologi%CC%81a.pdf?sequence=1&isAllowed=y>

Secretaría Jurídica Distrital. (2008). *Ley 1266 de 2008 Congreso de la República de Colombia*. <https://www.bogotajuridica.gov.co/sisjur/normas/Norma1.jsp?i=34488>

Sujay, L. (2023, June 1). *Cross-platform Mobile Frameworks Used by Software Developers Worldwide from 2019 to 2022*. <https://www.statista.com/statistics/869224/worldwide-software-developer-working-hours/>

React Native. (2023). *FlatList*. <https://reactnative.dev/docs/flatlist>

React. (2023). *API de Alto Nivel de React*. <https://es.legacy.reactjs.org/docs/react-api.html#reactmemo>

React Native. (2023). *FlatList*. <https://reactnative.dev/docs/flatlist>

UNAC. (2013). *2.-Politica-de-Proteccion-de-Datos-Personales.pdf* - Google Drive. <https://drive.google.com/file/d/1Fq-FW6RqYzRlelHwCZ3jJDmLmKzGB6wk/view>

UNAC. (2021a). *09. Estructura Organica 2022.pdf* - Google Drive. <https://drive.google.com/file/d/1jO-TNqmtujLXXFdFC-Xd-TB12nD3saof/view>

UNAC. (2021b). *UNAC Plan Estrategico3.pdf* - Google Drive. [https://drive.google.com/file/d/19TBpuAWyPBpEKvFUPw52y0a\\_VqlALSuw/view](https://drive.google.com/file/d/19TBpuAWyPBpEKvFUPw52y0a_VqlALSuw/view)

Uribe, L. (2021). Migracion de Aplicaciones Institucionales. *Informe*.  
[https://bibliotecadigital.udea.edu.co/bitstream/10495/19658/5/UribeLina\\_2021\\_MigracionAplicacionesInstitucionales.pdf](https://bibliotecadigital.udea.edu.co/bitstream/10495/19658/5/UribeLina_2021_MigracionAplicacionesInstitucionales.pdf)

Vmware. (2023). *¿Qué es la migración de aplicaciones? | Glosario de VMware | ES*.  
<https://www.vmware.com/es/topics/glossary/content/application-migration.html>