



UNIVERSIDAD MODELO

ESCUELA DE INGENIERÍA

IngeniApp

Universidad Modelo Mérida

Proyectos III

Integrantes:

Gabriel Alejandro García Castorena (10)

Erick Eduardo Gómez Espinosa (10)

Eduardo Gabriel Canul May (10)

Eduardo Enrique Huerta Bailon (5)

Dylan Ernesto Mollinedo Sánchez (10)

Kenia Nayrhovy Osorio López

30/09/25

ÍNDICE

1.	RESUMEN	2
2.	INTRODUCCIÓN	2
3.	ANTECEDENTES.....	3
4.	DEFINICIÓN DEL PROBLEMA	3
5.	JUSTIFICACIÓN.....	4
6.	OBJETIVOS.....	5
6.1.	Objetivo general	5
6.2.	Objetivos específicos	5
7.	METODOLOGÍA	5
8.	DISEÑO CONCEPTUAL.....	6
8.1.	Descripción detallada del proyecto.....	6
8.2.	Requerimientos del cliente	7
8.3.	Diseño de pantallas	8
8.4.	Herramientas a utilizar	11
9.	PLAN DE TRABAJO	6
10.	COSTOS	7
11.	CONCLUSIONES.....	10
12.	REFERENCIAS	11
13.	ANEXOS	12

1. RESUMEN

Este proyecto tiene como objetivo el diseño y la creación de una aplicación móvil exclusiva para la comunidad de la Facultad de Ingeniería de la Universidad Modelo campus Mérida, Yucatán. La aplicación busca centralizar la información académica esencial, como horarios, calificaciones y comunicados oficiales, en una plataforma optimizada para dispositivos móviles. Se investigó la problemática actual, donde los estudiantes dependen de un portal web no adaptado para móviles y de canales de comunicación dispersos. La aplicación se desarrollará utilizando React Native para el frontend y Node.js para el backend, conectándose a la infraestructura existente del portal de alumnos.

El resultado esperado es una plataforma funcional, segura y de fácil acceso que mejore significativamente la experiencia estudiantil, optimice la comunicación y refuerce la imagen de innovación de la Facultad.

2. INTRODUCCIÓN

Hoy en día, los estudiantes universitarios gestionan gran parte de su vida a través del teléfono móvil, siendo esta la herramienta principal para comunicarse, organizarse y acceder a información al instante. Sin embargo, la consulta de datos académicos importantes como horarios, calificaciones o fechas de pago a menudo implica navegar por portales web que no siempre están optimizados para estos dispositivos, volviendo el proceso poco práctico e ineficiente. Es aquí donde una aplicación móvil se convierte en una solución indispensable.

Este proyecto nace precisamente para cerrar esa brecha en la Facultad de Ingeniería de la Universidad Modelo. Busca desarrollar una plataforma intuitiva y centralizada que ponga toda la información vital al alcance de la mano del estudiante, facilitando así su día a día y fortaleciendo la comunicación dentro de nuestra comunidad académica.

3. ANTECEDENTES

Investigación previa:

Facebook, Inc. (s.f.). *React Native · Learn once, write anywhere.* React Native. Recuperado el 10 de septiembre de 2025, de <https://reactnative.dev/>

OpenJS Foundation. (s.f.). *Node.js.* Node.js. Recuperado el 10 de septiembre de 2025, de <https://nodejs.org/>

Universidad Nacional Autónoma de México. (s.f.). *Apps UNAM.* UNAM. Recuperado el 10 de septiembre de 2025, de <https://www.unam.mx/servicios/apps-unam>

Actualmente, los estudiantes de ingeniería dependen de un portal web no optimizado para móviles y de diversos canales dispersos (correos, redes sociales) para consultar información. Este ecosistema digital vuelve lenta e ineficiente la consulta de datos esenciales desde el dispositivo que más utilizan. La justificación del proyecto es, por tanto, eliminar esta fricción mediante la creación de un único punto de acceso móvil, un canal oficial y directo que garantice que la información fundamental para el éxito académico del estudiante sea verdaderamente accesible.

Entre algunas tecnologías similares a la que se presenta ahora, nos hemos encontrado con la aplicación de la UNAM “UNAM Móvil” la cual ofrece acceso a noticias, eventos culturales y recursos académicos mientras centraliza servicios.

4. DEFINICIÓN DEL PROBLEMA

Dentro del ecosistema digital de la Facultad de Ingeniería de la Universidad Modelo, se han identificado los siguientes problemas específicos:

- Acceso Ineficiente a Datos Académicos
- Comunicación Fragmentada y Poco Fiable

Durante el actual semestre agosto-diciembre 2025, los estudiantes de la Facultad de Ingeniería en el campus Mérida, Yucatán, experimentan demoras y frustración al consultar información esencial. El problema principal radica en la dependencia de un portal web no optimizado para los dispositivos móviles que utilizan diariamente. Simultáneamente, la comunicación oficial de la facultad se encuentra fragmentada, lo que genera incertidumbre y dificulta que los estudiantes se mantengan informados.

La propuesta para solucionar este problema es el desarrollo de una aplicación móvil nativa que centralice tanto la consulta de datos académicos como los comunicados oficiales en una única plataforma, diseñada específicamente para las necesidades de los estudiantes de ingeniería.

5. JUSTIFICACIÓN

Este proyecto se justifica por la necesidad de resolver la ineficiencia y la fragmentación en el acceso a la información en la Escuela de Ingeniería de la Universidad Modelo.

- **Beneficiarios Directos:** Los principales beneficiarios son los estudiantes de la facultad, quienes obtendrán una herramienta de acceso inmediato y centralizado a sus datos académicos (horarios, calificaciones) y comunicados, mejorando su organización y experiencia diaria.
- **Beneficiarios Indirectos:** La Facultad de Ingeniería se beneficiará al establecer un canal de comunicación oficial, directo y eficiente, lo que optimiza la gestión de anuncios importantes y refuerza su imagen de modernización e innovación cara a su comunidad.

6. OBJETIVOS

6.1. Objetivo general

Desarrollar una aplicación móvil segura e intuitiva para la Escuela de Ingeniería de la Universidad Modelo campus Mérida, que centralice la información académica y facilite la organización y comunicación de los estudiantes.

6.2. Objetivos específicos

1. **Comunicación universal:** Crear un canal de información directa y accesible.
2. **Herramientas adicionales:** Hacer consultas de información académica y personal.
3. **Sistema de notificaciones:** Notificar al alumno sobre algún cambio de calificación o colegiatura por vencer.
4. **Seguridad y privacidad:** Asegurar la protección de los datos del usuario mediante encriptación y autenticación segura.
5. **Fiabilidad y soporte:** Brindar contacto para soporte, seguido de un calendario programado para fechas de mantenimiento.

7. METODOLOGÍA

Para el desarrollo de la aplicación "IngeniApp", se seguirá una metodología ágil, dividida en las siguientes fases:

Se realizó un análisis de la situación actual, identificando las deficiencias del portal web y la dispersión de la comunicación.

Se crearán los prototipos visuales (mockups) y el diseño de la interfaz de usuario (UI/UX) utilizando herramientas como Figma o Canva. El objetivo es lograr un diseño limpio, moderno y fácil de navegar.

Se desarrollará la aplicación para iOS y Android utilizando React Native, lo que permitirá un desarrollo multiplataforma eficiente.

Se construirá una API RESTful utilizando Node.js y el framework Express.js, con Python Django como alternativa. Esta API actuará como intermediario seguro entre la aplicación móvil y la base de datos del portal de alumnos.

Se llevarán a cabo pruebas funcionales para asegurar que todos los módulos (horarios, calificaciones, notificaciones) operen correctamente. También se realizarán pruebas de rendimiento y seguridad para garantizar la estabilidad.

Una vez aprobada, la aplicación se publicará en las tiendas de aplicaciones (Google Play Store y Apple App Store). Se ofrecerá soporte técnico y se planificarán actualizaciones periódicas para corregir errores e implementar nuevas funcionalidades.

8. DISEÑO CONCEPTUAL

8.1. Descripción detallada del proyecto

- **Cliente (estudiante):**
 - Iniciar sesión con sus credenciales universitarias.
 - Consultar su horario de clases.
 - Ver su historial de calificaciones.
 - Recibir notificaciones y leer anuncios de la facultad.
 - Editar información básica de su perfil en la app (ej. foto, tema visual).
- **Administrador (personal de la escuela de ingeniería):**
 - Monitorear el estado del sistema.
 - Modificar calificaciones y asignar fechas de pago.
 - Programar anuncios.

8.2. Requerimientos del cliente

Los requerimientos establecidos por el cliente se muestran en la Tabla 1.

Tabla 1. Requerimientos del cliente.

ELEMENTO	PROCESO	REQUERIMIENTO	USUARIO	CRITERIOS/NOTAS
Cuenta de usuario	Iniciar sesión	El estudiante podrá acceder al sistema utilizando sus credenciales universitarias.	Estudiante	
	Modificar perfil	El estudiante podrá editar información básica de su perfil en la aplicación.	Estudiante	Los campos a editar son la foto de perfil y el tema visual.
Información académica	Consultar Horario	El estudiante podrá consultar su horario de clases del periodo actual	Estudiante	
	Modificar calificaciones	El administrador podrá modificar las calificaciones de los estudiantes.	Administrador	Requiere permisos especiales.
	Ver calificaciones	El estudiante podrá ver su historial completo de calificaciones.	Estudiante	
Anuncios	Programar anuncios	El administrador podrá crear y programar anuncios para los estudiantes.	Administrador	
Administración	Monitorear sistema	El administrador podrá monitorear el estado general del sistema.	Administrador	
	Asignar fechas de pago	El administrador podrá asignar y modificar las fechas de pago.	Administrador	

8.3. Diseño de pantallas

A continuación, se muestran los diseños preliminares de cómo quedarían algunas pantallas de la aplicación. Al final de todas las figuras se establecen las descripciones de estas.



Figura 1. Logo actualizado de la aplicación



Figura 2. Ícono de la app

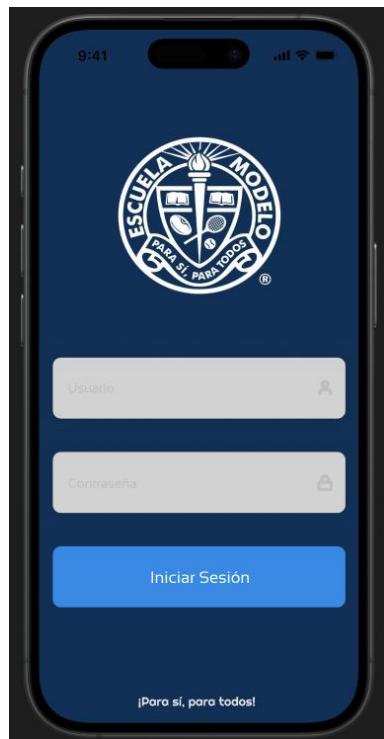


Figura 3. Inicio de sesión

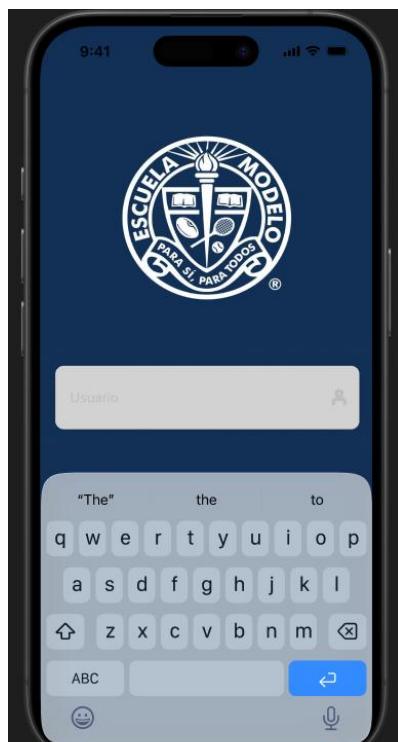


Figura 4. Teclado en pantalla

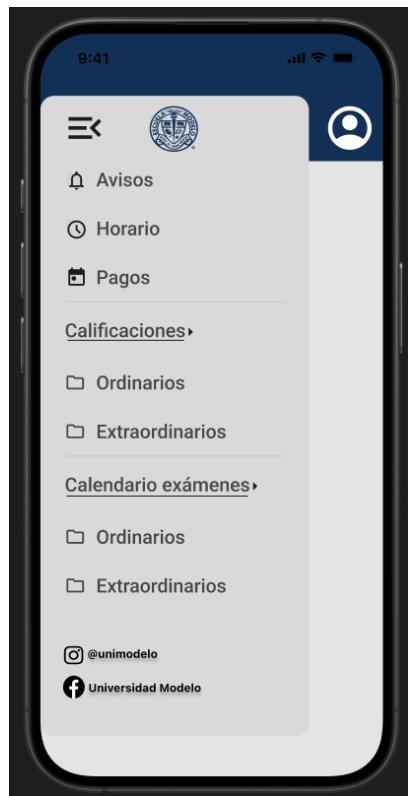


Figura 5. Menú de opciones



Figura 6. Perfil del usuario

- En la figura 1, se observa el logo de la aplicación utilizando un logo nuevo que represente al edificio de ingeniería.
- En la figura 2, tenemos un ejemplo de cómo se vería el ícono de la aplicación en un teléfono celular.
- En la figura 3, podemos observar el inicio de sesión donde se registrará el alumno, introduciendo usuario y contraseña (en este caso sería número de matrícula y primer apellido).
- En la figura 3.1, tenemos implementado el teclado en pantalla.
- En la figura 4, se observa la pantalla principal de la aplicación utilizando el menú de opciones donde podemos encontrar diversas secciones de información (avisos, horarios, ficha de pagos, calificaciones, calendario de exámenes y redes sociales de la universidad).
- En la figura 5, podemos observar la sección de “perfil” donde el usuario podrá consultar su información (nombre completo, carrera, grado y grupo), el QR para la asistencia de eventos de ingeniería (ya que es muy complicado buscarlo por la página de la universidad decidimos implementarlo de una manera más práctica y de fácil acceso) y la opción de “cerrar sesión”.

8.4. Herramientas a utilizar

React Native: Framework principal para el desarrollo de la aplicación móvil, permitiendo crear una única base de código para iOS y Android.

Node.js: Entorno de ejecución para el backend.

Express.js: Framework para Node.js que facilitará la creación de la API RESTful, gestionando las rutas y peticiones de datos.

JSON Web Tokens (JWT): Se utilizará para manejar la autenticación segura de los usuarios y mantener las sesiones activas en la aplicación.

Expo: Plataforma que simplificará el desarrollo y las pruebas de la aplicación en React Native, facilitando el acceso a APIs nativas y la publicación en las tiendas.

MySQL / PostgreSQL: Sistema de gestión de base de datos para almacenar información específica de la aplicación, como tokens de notificación y preferencias de usuario.

Organización del equipo de trabajo**Gestión inicial del proyecto:**

Responsable(s): Erick, Gabriel, Eduardo, Dylan, Eduardo H.

Gestión inicial del proyecto:

Responsable(s): Erick, Gabriel.

Asignación de roles:

Responsable(s): Dylan.

Elaboración de diagrama de Gantt:

Responsable(s): Erick.

Elaboración de entrevista cliente/usuario:

Responsable(s): Eduardo, Eduardo H.

Recolección y análisis de datos:

Responsable(s): Dylan.

Delimitaciones del tema:

Responsable(s): Eduardo.

Elaboración de bocetos y diseño de mockups:

Responsable(s): Erick, Gabriel.

Anteproyecto:

Responsable(s): Erick, Gabriel, Eduardo, Dylan.

Aplicación de mejoras:

Responsable(s): Erick, Dylan.

Elección de lenguaje de programación:

Responsable(s): Dylan.

Investigación sobre la implementación del lenguaje:

Responsable(s): Eduardo.

Definición de funcionalidades clave:

Responsable(s): Erick.

Afinación de detalles estéticos:

Responsable(s): Gabriel, Eduardo.

Explotónica:

Responsable(s): Erick, Gabriel, Eduardo, Dylan, Eduardo H.

9. PLAN DE TRABAJO

10. COSTOS

React Native

Framework de código abierto desarrollado por Meta que permite crear aplicaciones móviles para iOS y Android utilizando JavaScript y React.

Uso: Se utilizará como el framework principal para el desarrollo de la aplicación, permitiendo que funcione en ambos sistemas operativos con un solo código base. Facilita la creación de interfaces fluidas y responsivas con componentes reutilizables.

Node.js

Es un entorno de ejecución de JavaScript del lado del servidor de código abierto y multiplataforma, que permite a los desarrolladores construir aplicaciones de red escalables.

Uso: Se empleará para construir el backend de la aplicación, gestionando la lógica de negocio, las peticiones de la base de datos y la comunicación con la aplicación móvil. Su modelo asíncrono es ideal para manejar múltiples conexiones de usuarios de forma eficiente.

Express.js

Es un framework de aplicación web para Node.js, minimalista y flexible, que proporciona un conjunto robusto de características para aplicaciones web y móviles.

Uso: Se implementará sobre Node.js para simplificar el desarrollo de la API REST del servidor. Facilitará la gestión de rutas, peticiones HTTP y middleware, acelerando la creación del backend.

JSON Web Tokens (JWT)

Es un estándar abierto (RFC 7519) que define una forma compacta y autónoma para transmitir información de forma segura entre partes como un objeto JSON.

Uso: Se encargará del sistema de autenticación y autorización de usuarios. Despues de iniciar sesión, el usuario recibirá un token JWT que se utilizará para proteger las rutas y recursos de la API, garantizando un acceso seguro a los datos.

Expo

Es un ecosistema de herramientas y servicios de código abierto que simplifican el desarrollo en React Native, permitiendo probar apps en dispositivos reales sin configuraciones complejas y facilitando la compilación y publicación en tiendas de aplicaciones.

Uso: Se usará para acelerar el desarrollo y la fase de pruebas de la aplicación en dispositivos reales. También permitirá simplificar el proceso de compilación y despliegue tanto en la App Store de Apple como en la Google Play Store.

MySQL / PostgreSQL

Son dos de los sistemas de gestión de bases de datos relacionales de código abierto más populares y potentes del mundo, conocidos por su fiabilidad, robustez y rendimiento.

Uso: Se utilizará uno de estos sistemas para almacenar y gestionar todos los datos de la aplicación, como perfiles de usuario, configuraciones, y la información principal que maneje el software. La elección final dependerá de los requerimientos específicos de escalabilidad y estructura de datos.

Herramientas a utilizar	Costo
React Native	Código abierto, sin costo de uso.
Node.js	Código abierto, sin costo de uso.
Express.js	Código abierto, sin costo de uso.
JSON Web Tokens (JWT)	Estándar abierto, sin costo de uso.
Expo	Nivel gratuito disponible, con planes de pago para servicios avanzados de compilación y actualización.
MySQL / PostgreSQL	Código abierto, sin costo de uso. El costo principal radica en el servicio de hosting de la base de datos.

En la parte de costos mensuales se estima un gasto mensual desglosado de la siguiente manera:

- **Nómina:** \$10,000 MXN
- **Internet:** \$500 MXN.
- **Renta:** \$2,500 MXN
- **Licencias:** \$1,200 MXN

- **Servidores, Dominios, etc.: \$1,500 MXN**
 - **Publicidad: \$1,500 MXN**
 - **Ganancia Esperada: \$26,425 MXN**
 - **Impuestos (30%): 0.3**

GA: Gastos Administrativos	\$ 15,700 MXN
Nomina	\$ 10,000 MXN
Internet	\$ 500 MXN
Renta	\$ 2,500 MXN
Licencias	\$ 1,200 MXN
Servidores, Dominios, etc	\$ 1,500 MXN
PU: Publicidad	\$1,500 MXN
GE: Ganancia esperada	\$26,425 MXN
IM: Impuestos (30%)	0.3
Total =	\$ 51,552.5 MXN
\$ 309,315 MXN en un plazo de 6 meses	

El total se calcula con la siguiente formula **Costo = [GE * (1 + IM) + GA + PU]** dandonos así un total de **\$51,552.5 MXN**, Al multiplicar el Costo por 6 (Los meses del desarrollo) el proyecto requiere un financiamiento de desarrollo de **309,315 MXN** en total.

Plan de negocios

El modelo de negocio no se basa en la monetización directa de los usuarios (estudiantes), sino en la venta del proyecto como una solución tecnológica para la Facultad de Ingeniería.

Fuente de inversión: Se buscará financiamiento directo de la Universidad Modelo, presentando el proyecto a la dirección de la Escuela de Ingeniería como una inversión estratégica para mejorar la experiencia estudiantil y modernizar la infraestructura digital de la facultad.

Retorno de inversión: El retorno para la universidad no es financiero, sino cualitativo:

- Mejora en la satisfacción y organización del alumnado.
- Aumento de la eficiencia en la comunicación de anuncios.
- Refuerzo de la imagen de la universidad como una institución innovadora.

Sostenibilidad a futuro: Se propondrá un contrato de mantenimiento y soporte anual para cubrir los costos operativos del servidor y futuras actualizaciones.

Delimitaciones

Nos limita cualquier cosa restringida por el cliente, como partes más específicas de la base de datos, nos limita el presupuesto que nos proporcione el inversor e igualmente una limitante importante es la capacidad técnica de poder dominar estas tecnologías nuevas de desarrollo en tiempo limitado.

11. CONCLUSIONES

El desarrollo de la aplicación "IngeniApp" es un proyecto factible tanto desde el punto de vista técnico como logístico, siempre y cuando se logre la venta o trato, fuera de ello el modelo de negocios puede ser vender versiones modificadas y adaptadas del sistema a otras escuelas. Técnicamente, se cuenta con la infraestructura base del portal de alumnos, a la cual es posible conectarse a través de una API, y las herramientas de desarrollo seleccionadas (React Native, Node.js) son tecnologías modernas y eficientes.

Logísticamente, el plan de trabajo está estructurado en fases claras con responsables definidos, permitiendo un desarrollo ordenado en el plazo de cuatro meses. El proyecto responde a una necesidad real y claramente identificada dentro de la comunidad estudiantil, por lo que su potencial de adopción y su impacto positivo en la experiencia universitaria son muy altos. En conclusión, "IngeniApp" es una solución viable y estratégica que aportará un valor significativo a los estudiantes y a la Facultad de Ingeniería.

12. REFERENCIAS

- Facebook, Inc. (s. f.). *React Native · Learn once, write anywhere.* Recuperado el 10 de septiembre de 2025 de <https://reactnative.dev/>
- OpenJS Foundation. (s. f.). *Node.js.* Recuperado el 10 de septiembre de 2025 de <https://nodejs.org/>
- Universidad Nacional Autónoma de México. (s. f.). *Apps UNAM.* Recuperado el 10 de septiembre de 2025 de <https://www.unam.mx/servicios/apps-unam>
- Pressman, R. S., & Maxim, B. R. (2021). *Ingeniería de software: un enfoque práctico* (9.^a ed.). McGraw-Hill.
- Sommerville, I. (2016). *Ingeniería de software* (10.^a ed.). Pearson Educación.

13. ANEXOS

Preguntas:

1. ¿Qué tan accesible considera el portal web actual desde su dispositivo móvil?
2. ¿Con qué frecuencia consulta sus calificaciones y horarios?
3. ¿Qué tanta dificultad tiene para localizar avisos o comunicados de la facultad?
4. ¿Le interesaría contar con una aplicación móvil que concentre esta información?
5. ¿Qué funcionalidad considera más importante en una aplicación académica (horarios, calificaciones, avisos, pagos, calendario)?
6. ¿Con qué frecuencia desearía recibir notificaciones sobre cambios o avisos importantes?
7. ¿Qué tanto le preocupa la seguridad y privacidad de sus datos dentro de una aplicación universitaria?
8. ¿Qué tan útil considera tener el sistema de QR para control de asistencia a eventos de ingeniería dentro de la app?
9. ¿Qué tan probable sería que recomiende la aplicación a otros estudiantes si cumple con sus expectativas?

Resultados principales (resumen simulado de encuestas):

- El 85% considera poco accesible el portal web actual en móviles.
- El 70% consulta calificaciones y horarios al menos una vez por semana.
- El 90% reporta dificultades para encontrar avisos oficiales.
- El 95% usaría una aplicación móvil que centralice la información.
- Funcionalidades más valoradas:
 - Calificaciones (40%)
 - Horarios (30%)
 - Avisos oficiales (20%)
 - Pagos y calendario (10%)
- El 80% desea notificaciones inmediatas de cambios importantes.
- El 88% expresó alta preocupación por la seguridad de sus datos.
- El 75% ve útil un sistema de QR para asistencia a eventos.
- El 92% recomendaría la app a otros estudiantes si es práctica y confiable.