



# ESCUELA POLITÉCNICA NACIONAL

## ESCUELA DE FORMACIÓN DE TECNÓLOGOS

---



### DESARROLLO APP MOVILES

ASIGNATURA: Desarrollo APP Moviles

PROFESOR: Ing. Juan Carlos Gonzalez MSc.

PERÍODO ACADÉMICO: 2025-B

### 2DO BIMESTRE

### TÍTULO

### PRUEBA 2 – DESARROLLO APP MOVILES

Nombres:

**Guerra Lovato Josué Eduard**

## PRUEBA 2 – DESARROLLO APP MOVILES

PetAdopt es una aplicación móvil que desempeña una labor social de gran relevancia: facilitar un segundo hogar para la fauna urbana en situación de abandono.

La plataforma se integra con diversos servicios externos, ofreciendo una experiencia integral y confiable en el proceso de adopción de mascotas.

La arquitectura del sistema se compone de:

- **Framework Móvil:** Aplicación desarrollada en Flutter
- **Backend:** Supabase (Auth, Database, Storage)
- **Sitio Web Auxiliar:** Desplegado en Vercel/Render/Railway(a elección del estudiante podría ser otra herramienta)para despliegue de interfaz de confirmación de cuenta y reseteo de contraseña
- **IA Conversacional:** API de Gemini para asistente virtual
- **Mapas:** OpenStreetMap para geolocalización de refugios

### Alcance del Sistema

PetAdopt es una aplicación móvil multiplataforma que permitirá:

- Registro y autenticación de usuarios (adoptantes y refugios) mediante Supabase Auth
- Gestión completa de mascotas en adopción por parte de refugios (CRUD)
- Visualización y búsqueda de mascotas disponibles para adoptantes
- Sistema de solicitudes de adopción con flujo de aprobación/rechazo
- Asistente virtual con IA (Gemini) para consultas de salud y cuidados
- Localización de refugios cercanos mediante GPS y OpenStreetMap

### Integraciones Externas

#### Supabase

- **Auth:** Gestión de usuarios, sesiones, confirmación de email y reseteo de contraseña
- **Database:** PostgreSQL con Row Level Security para control de acceso
- **Storage:** Almacenamiento de imágenes de las mascotas con políticas de acceso

#### Vercel/Railway/Render

Sitio web auxiliar para:

- Página de confirmación de cuenta
- Formulario de reseteo de contraseña
- Redirección personalizada post-acción

#### Gemini AI

Integración con la API de Gemini para:

- Procesamiento de consultas en lenguaje natural
- Generación de respuestas sobre salud y cuidados
- Mantenimiento de contexto conversacional

## OpenStreetMap

- Tiles de mapa gratuitos y de código abierto
- Integración mediante Leaflet o bibliotecas nativas
- Geocodificación para búsqueda de direcciones

## Criterios de Evaluación

Criterio	Peso
Implementación de Supabase Auth con perfiles y Vercel/Railway/Render	20%
CRUD de mascotas con Supabase database-Storage	20%
Chat con IA (Gemini)	20%
Sistema de gestión de solicitudes de adopción	15%
Mapa con GPS y OpenStreetMap	25%
<b>TOTAL</b>	<b>100%</b>

### Puntos Adicionales (en la nota final sobre 20):

- 2 puntos si logra implementar **autenticación mediante terceros** o más específicamente como **OAuth** con Google.
- **3 puntos** si implementa un sistema de notificaciones push en tiempo real para la gestión de solicitudes de adopción.

### Sanciones(en la nota final sobre 20):

- Si la aplicación no tiene una interfaz intuitiva/responsiva con un diseño moderno y atractivo visualmente , se aplicará sanción de **-6 puntos** a la nota final sobre 20.
- No usar Arquitectura Clean le resta **-2 puntos** a la nota final sobre 20.
- Si la aplicación de un estudiante se parece a la de su compañero automáticamente se asigna la **nota de cero** a los estudiantes involucrados.
- La calificación y asignación de nota del proyecto está condicionada a la demostración de autoría y dominio técnico durante la defensa. El estudiante debe demostrar que comprende completamente el código que presenta. El estudiante DEBE ser capaz de:
  - Demostrar la Arquitectura de la app.

- Explicar la estructura de carpetas.
- El flujo de datos y cómo se comunican los componentes Código
- Describir qué hace cada archivo, función y componente de su proyecto.
- Explicar cómo se conecta la app con Supabase, Gemini, OpenStreetMap y otros servicios.

**En caso de no demostrar dominio técnico, se asigna la nota de cero a la prueba, así la misma cumpla todos los requisitos técnicos.**

- La aplicación debe presentarse instalada en un celular por medio del apk, caso contrario se penaliza con **-10 puntos** menos a la nota final sobre 20.

### **Entregables:**

- Adjuntar el enlace del repositorio de Github en este documento y subir a la plataforma en formato pdf hasta el 06-01-2026 antes de las 15:59.
- Adjuntar el enlace de OneDrive del apk de su aplicación en este documento hasta el 06-01-2026 antes de las 15:59.
- Demostrar en clase al docente la aplicación instalada en el celular y las funcionalidades desarrolladas el día 06-01-2026 a las 16:00.

<b>Link del repositorio Github de la app:</b>	<a href="https://github.com/josdank/Prueba_B2_AppMovile_JosueGuerra.git">https://github.com/josdank/Prueba_B2_AppMovile_JosueGuerra.git</a>
<b>Link OneDrive con el APK de la app:</b>	<p>1. Enlace directo de descarga:  <a href="https://raw.githubusercontent.com/josdank/Prueba_B2_AppMovile_JosueGuerra/7b36d7e3df2dbd9df3f65d652094f007db88dadd/app-release.apk">https://raw.githubusercontent.com/josdank/Prueba_B2_AppMovile_JosueGuerra/7b36d7e3df2dbd9df3f65d652094f007db88dadd/app-release.apk</a></p> <p>2. Enlace del Onedrive:  <a href="#">APK PRUEBA2 BIM2</a></p>

**Nota:** Si no se sube la documentación de entregables solicitada antes del horario indicado 06-01-2026 antes de las 15:59, no se revisa la aplicación en clase, así la tenga completamente terminada el estudiante.