

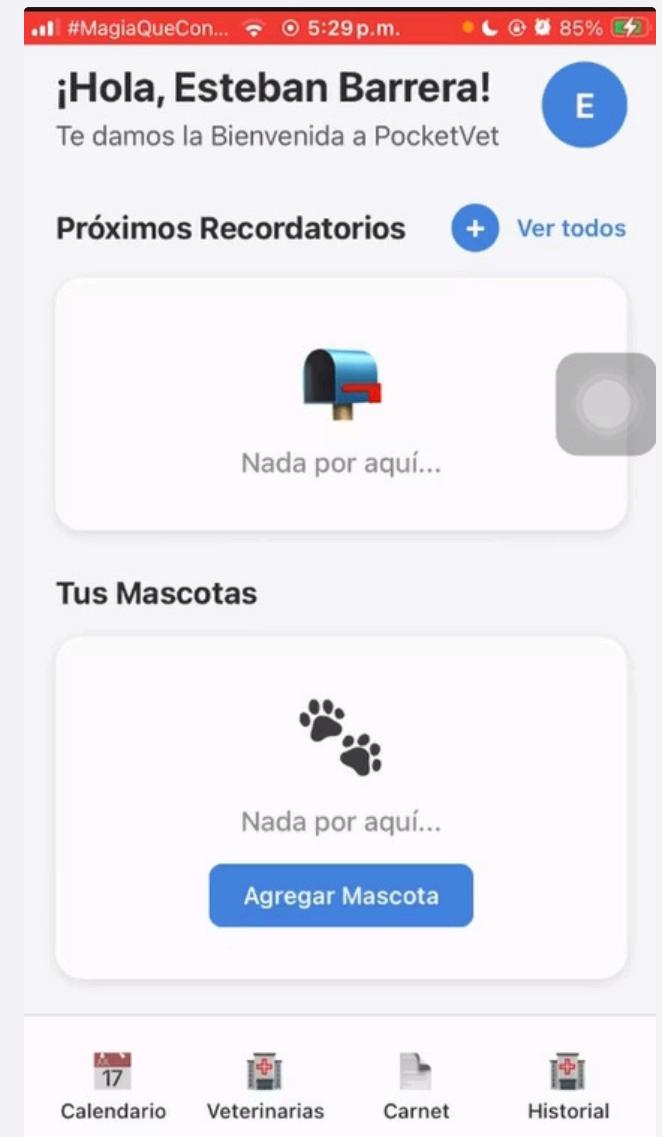
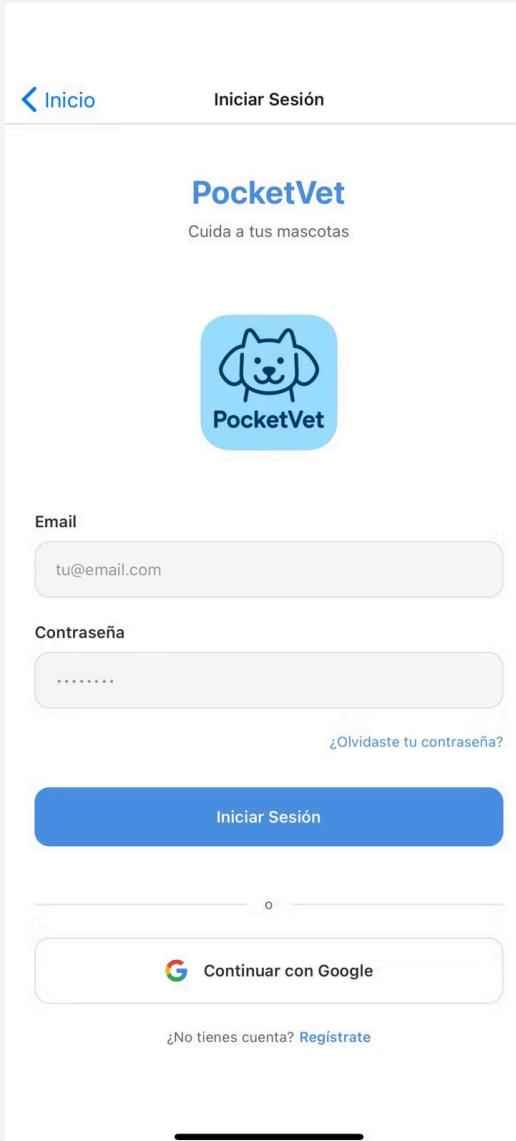


PocketVet, aplicación de  
gestión de mascotas y  
búsqueda de veterinarias.

# Planteamiento

## PocketVet

PocketVet se pensó como una aplicación para dispositivos móviles (Android/ios) que busca organizar y facilitar la vida de dueños de mascotas al poder tener en un mismo entorno integrados los servicios fundamentales de cuidado de mascotas, desde la carnetización hasta la búsqueda de centros de atención y grooming o recordatorios personalizados de citas o eventos que tenga con su mascota.



# Levantamiento del proyecto

1

## Creación del Proyecto

Se decidió hacer en Typescript con  
Expo react

Plantilla oficial de Expo:

```
npx create-expo-app -t blank  
(typescript)
```

2

## Creación base de datos

Inicialmente se pensó para firebase.  
Terminó siendo postgresql con  
prisma studio.



3

## Desarrollo

Uso de estrategias proporcionadas  
por el profesor a lo largo del  
semestre



# Librerías y APIs Clave



## Expo SDK

Base multiplataforma para iOS, Android y Web.



## OpenStreetMap

Gestión de mapas.



## MapTiler API

Dibujado de Tiles para los mapas.



## React Navigation

Manejo avanzado de rutas y navegación.



## Google Auth

Uso de Google auth para gestión de cuentas.



## Testing

Jest, Eslint/prettier

# Estructura y Estilo del Código

- **Componentes Funcionales**

Uso de interfaces TypeScript para props.

- **Tipado Estricto**

Activado para prevenir errores en tiempo de ejecución.

- **Modularización**

Separación clara de componentes, hooks y utilidades.

- **Consistencia del Código**

Uso de Prettier y ESLint para mantener alta calidad.



# Retos Técnicos y Soluciones



## Migración de firebase a Postgresql

Reestructuración completa del backend del proyecto.



## Autenticación de usuarios

Configuración de Google Auth por correo electrónico.



## Dibujado de mapas

Uso de OpenStreetMaps.



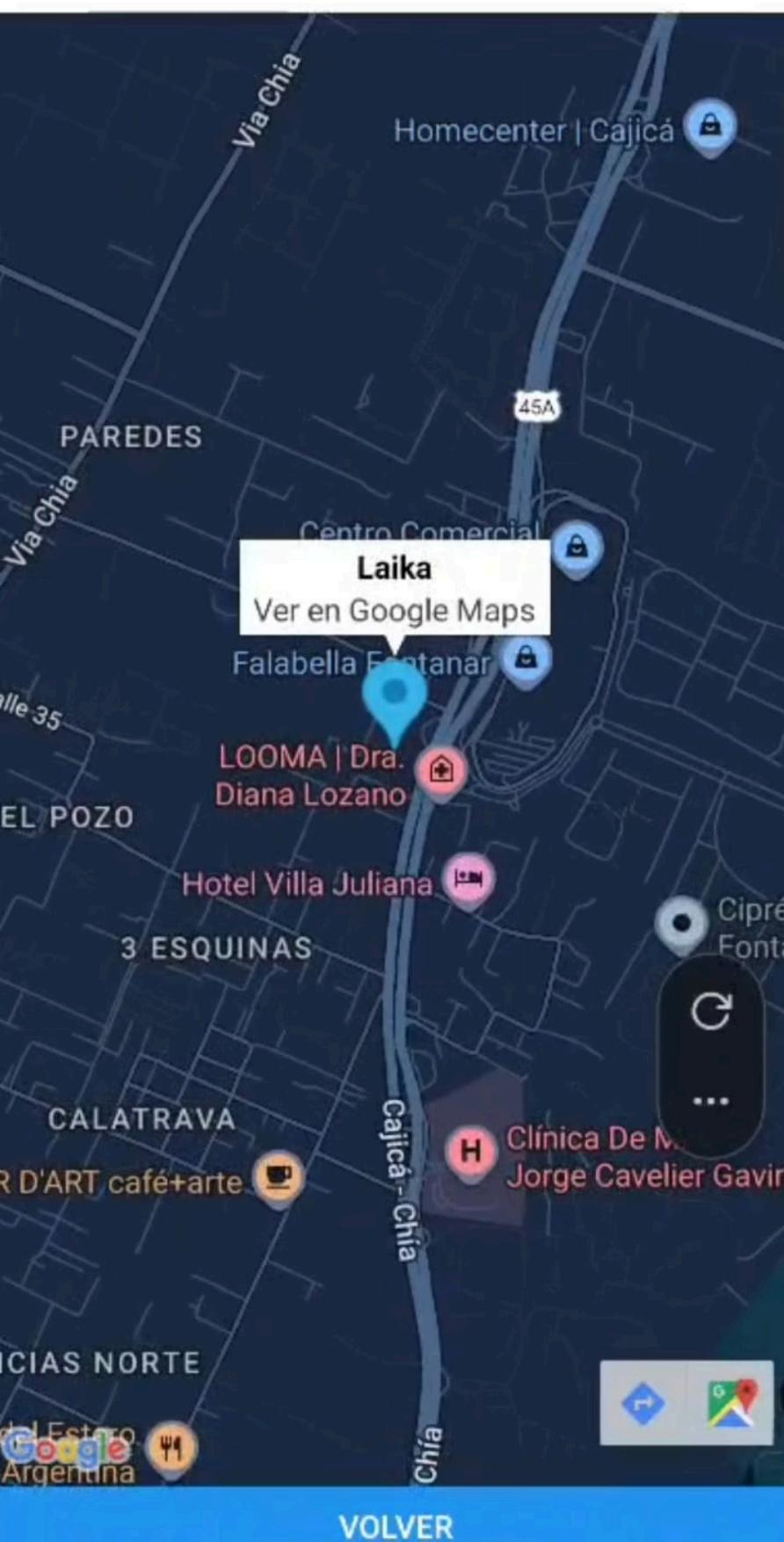
## Gestión de notificaciones

Notificaciones personalizadas por usuario.



## Compatibilidad

SafeAreaView para distintos dispositivos.



# Herramientas de Desarrollo y Debugging



## Expo CLI

Desarrollo rápido y testing en dispositivos reales.



## TypeScript

Autocompletado y detección temprana de errores.



## OpenStreetMaps

Mapeado optimizado para dispositivos móviles.



## Pruebas

Unitarias con jest y uso de linters con ESLint.

# Reflexiones Finales

“

## Calidad y Escalabilidad

Expo + TypeScript: potencia y robustez.

”

“

## Curva de Aprendizaje

Un poco inclinada al inicio.

”

“

## Buenas Prácticas

Tipado estricto, modularidad y testing: claves del éxito.

”

“

## Aprendizajes

Rutas, optimización y gestión de estados.

”

