

## **Etapas de Implementación**



**BusTrack**

## 1. Lenguaje de Programación

Se utilizará **Dart** con el framework **Flutter** para el desarrollo de la aplicación móvil, ya que permite crear aplicaciones multiplataforma (Android e iOS) con un único código. Para el backend y la lógica de negocio, se usará **Python** o **Node.js**, dependiendo de las necesidades específicas del proyecto, con el objetivo de gestionar la API de comunicación con los datos GPS.



*Ilustración 1 Dart & Flutter*

## 2. Frameworks y Herramientas

- **Flutter:** Para el desarrollo de la interfaz de usuario intuitiva y dinámica.
- **Firebase:** Gestión de la base de datos en tiempo real, autenticación de usuarios y notificaciones push.
- **Google Maps API o Mapbox:** Para la integración del mapa interactivo que mostrará la ubicación en tiempo real de los buses.
- **Git y GitHub:** Control de versiones, organización del código y colaboración en el repositorio del proyecto.
- **Postman:** Para probar y documentar las APIs desarrolladas.

## 3. Hardware

- Se planea que el sistema sea compatible con **dispositivos GPS instalados en los buses**, que enviarán datos en tiempo real al servidor.
- Pruebas iniciales se realizarán simulando dispositivos GPS con herramientas de depuración y emulación para validar la precisión y latencia del sistema.



*Ilustración 2 GPS*

#### 4. Base de Datos y Almacenamiento

- Se usará **Firestore Realtime Database** o **Firestore** para almacenar:
  - Información de usuarios, buses, rutas y tarifas.
  - Datos temporales relacionados con la ubicación GPS de los buses.
- **Google Cloud Platform (GCP)** o **AWS** podrían emplearse si la escala del proyecto requiere mayor capacidad o procesamiento.

#### 5. Metodología de Desarrollo

Se adoptará una **metodología ágil como Scrum** para garantizar el desarrollo iterativo y colaborativo:

- El proyecto se dividirá en **sprints** de dos semanas, priorizando las funcionalidades críticas, como:
  - **Sprint 1:** Configuración inicial del proyecto y pruebas con Google Maps API.
  - **Sprint 2:** Desarrollo del backend básico para gestionar rutas, líneas y precios.
  - **Sprint 3:** Implementación de la funcionalidad de rastreo en tiempo real.
  - **Sprint 4:** Integración de notificaciones personalizadas y filtros dinámicos.
- Retroalimentación constante permitirá ajustes rápidos antes de pasar a las fases finales.