

# BUS-MAIDAN

## Documentación Técnica del Proyecto

### 1. Introducción

BUS-MAIDAN (anteriormente Montevideo Bus Tracker) es una aplicación web que muestra el movimiento en tiempo real de los autobuses en Montevideo, Uruguay. Utiliza la API oficial de Transporte Público de Montevideo para obtener datos de ubicación de los autobuses y los visualiza en un mapa interactivo.

### 2. Arquitectura General

La aplicación sigue una arquitectura cliente-servidor donde:

1. Backend (Flask): Se encarga de la autenticación con la API de Montevideo, la recuperación de datos y la exposición de endpoints REST.
2. Frontend (JavaScript): Visualiza los datos en un mapa interactivo usando Leaflet y proporciona funcionalidades de UI.

### 3. Estructura de Archivos

- app.py: Aplicación principal Flask
- main.py: Punto de entrada
- install.py: Script de instalación
- static/: Recursos estáticos (CSS, JS, imágenes)
- templates/: Plantillas HTML
- static/js/main.js: Lógica principal del frontend
- static/img/: Iconos SVG personalizados

### 4. Backend (Flask)

#### 4.1 Autenticación con la API

La aplicación utiliza OAuth 2.0 con credenciales de cliente para autenticarse con la API. El método `get_access_token()` obtiene o refresca el token de acceso.

#### 4.2 Endpoints del Backend

- /: Página principal
- /api/buses: Datos de autobuses
- /api/lines: Líneas disponibles
- /api/stops: Paradas de autobuses

#### 4.3 Modo de Simulación

Para desarrollo o cuando la API no está disponible, la aplicación puede usar datos simulados mediante las funciones `generate_simulated_buses()` y `generate_simulated_stops()`.

## 5. Frontend (JavaScript)

### 5.1 Inicialización del Mapa

El mapa se inicializa con Leaflet y se configuran capas para buses y paradas. Se añaden controles de capas, geolocalización y escala.

### 5.2 Marcadores de Mapa

Los marcadores de autobuses y paradas se crean con iconos SVG personalizados. Los autobuses se rotan según su dirección y muestran información en tooltips.

### 5.3 Geolocalización

La aplicación utiliza la geolocalización del navegador para mostrar la ubicación del usuario y encontrar paradas cercanas a su posición.

## 6. API de Transporte Público de Montevideo

La aplicación utiliza los siguientes endpoints:

- GET /buses: Autobuses en circulación
- GET /buses/busstops: Todas las paradas
- GET /buses/busstops/{id}/upcomingbuses: ETA a paradas
- GET /buses/linevariants: Variantes de línea
- GET /buses/gtfs/static/latest/google\_transit.zip: Datos GTFS

## 7. Instalación y Configuración

### 7.1 Variables de Entorno

- MONTEVIDEO\_CLIENT\_ID: ID de cliente para la API
- MONTEVIDEO\_CLIENT\_SECRET: Secreto de cliente
- SIMULATION\_MODE: Activar modo simulación

### 7.2 Proceso de Instalación

El proceso se documenta en README.md y se automatiza mediante install.py. Los pasos básicos son:

1. Clonar el repositorio
2. Crear entorno virtual (opcional)
3. Instalar dependencias
4. Configurar variables de entorno
5. Iniciar con Gunicorn

## 8. Conclusión

BUS-MAIDAN es una aplicación web completa para seguimiento de autobuses que combina un backend Flask con un frontend JavaScript y visualización en mapa. Utiliza la API oficial de Transporte Público de Montevideo para proporcionar datos en tiempo real.

**Proyecto Open Source BUS-MAIDAN**

*Documentación Técnica - 2025*