

PROYECTO PROGRAMACIÓN 6C 2025

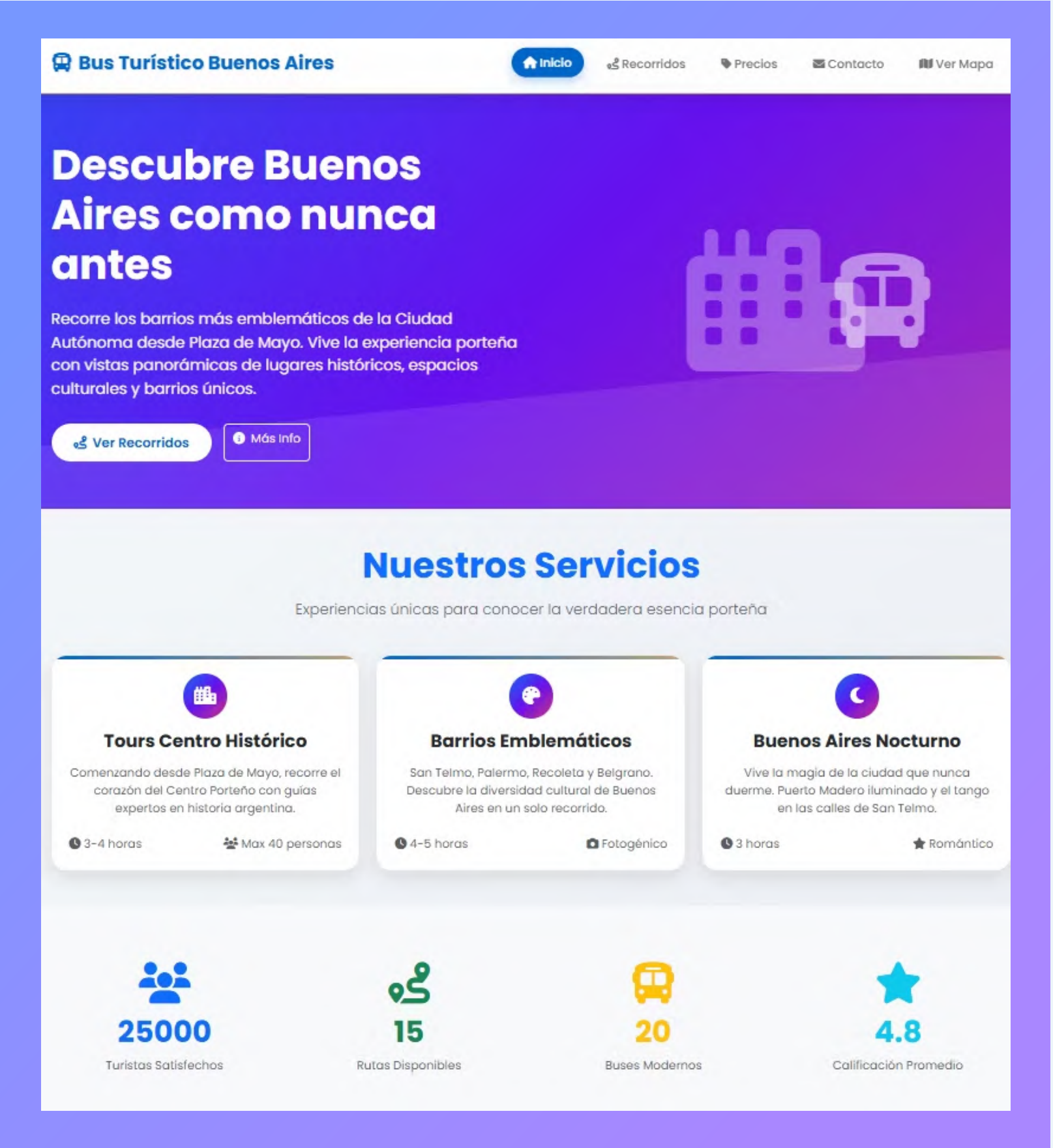
GRUPO N°7: OLMEDO VALENTINO, NUÑES TOMÁS, CARRERA
MATÍAS Y BENAVIDEZ LAUTARO

BUS TURÍSTICO



DESCRIPCIÓN GENERAL:

Creamos un sitio web informativo del Bus Turístico de Buenos Aires. Los usuarios pueden ver los recorridos, precios, horarios, paradas y sus correspondientes atractivos, además de poder contactar fácilmente por WhatsApp, teléfono o email.



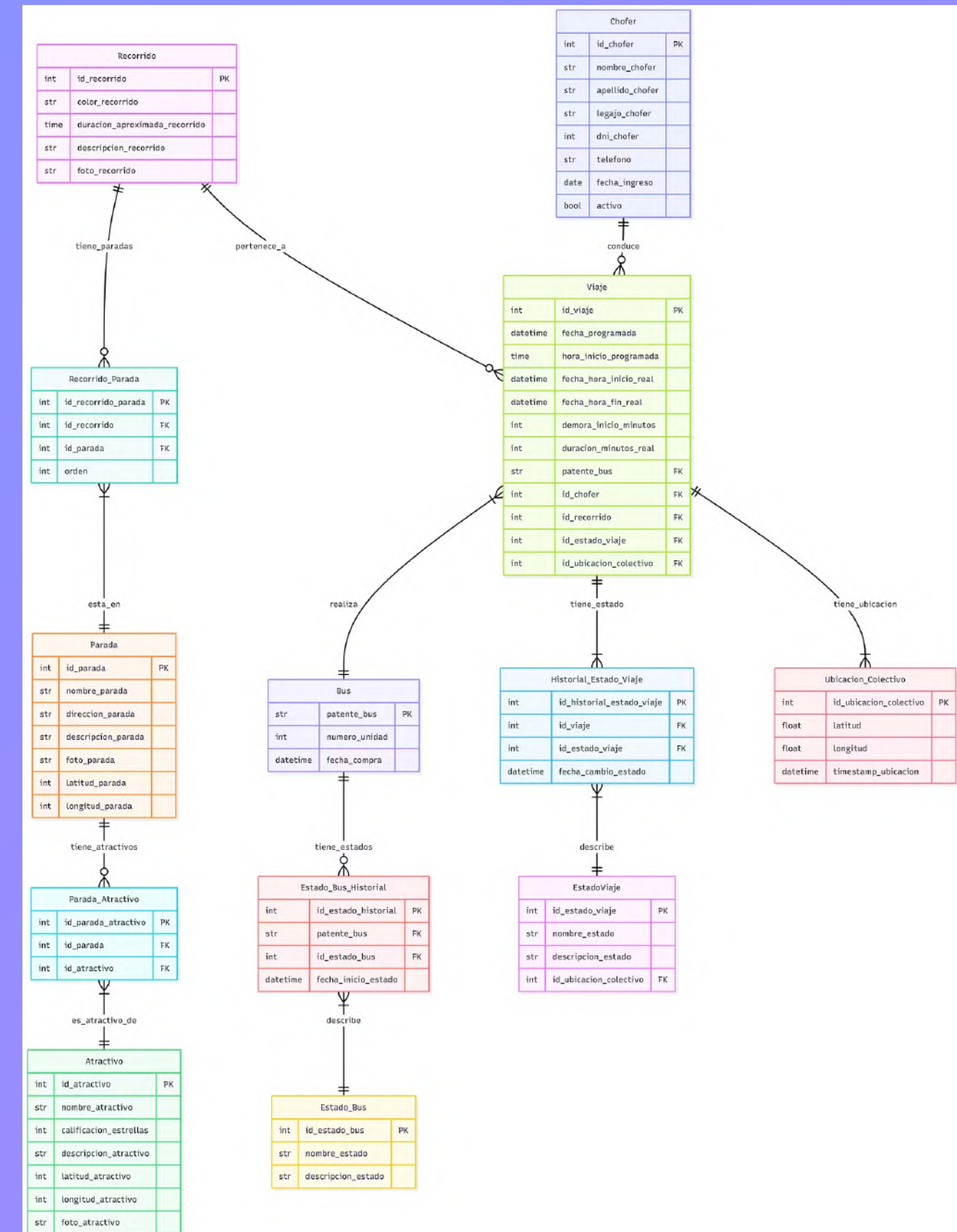
OBJETIVOS

Gestionar de forma eficiente los viajes de recorridos turísticos.

Automatizar el proceso de registro de choferes y recorridos, evitando tareas manuales y reduciendo errores. Brindar una interfaz intuitiva para que los usuarios puedan realizar sus consultas de manera rápida y los administradores puedan visualizarlas y responderlas fácilmente.

Análisis del dominio

- Modela recorridos, paradas y atractivos del bus turístico.
- Cada viaje asigna bus, chofer y recorrido con fecha y estado.
- Se registran historiales de estado y ubicación de buses y viajes.
- Permite conocer itinerarios, atractivos y seguimiento de viajes.



Tecnologías utilizadas



Django en Python



HTML



CSS



Bootstrap



Sqlite3



WSL Ubuntu

TRABAJO EN EQUIPO / CONTROL DE VERSIONES

Github

FUNCIÓN:

Trabajar en equipo de manera organizada, evitando conflictos entre versiones del proyecto. Registrar los cambios de cada integrante a traves de commits y branches.

LINK DEL REPOSITORIO

<https://github.com/Villada-PG3/trabajo-practico-integrador-bus-turistico.git>



Valentino Olmedo
DESAROLLADOR



Tomás Nuñez
DESAROLLADOR



Matias Carrera
DESAROLLADOR



Lautaro Benavidez
DESAROLLADOR

DIFICULTADES

Cada uno estaba tocando el mismo proyecto y al hacer git push tiraba error porque el remoto tenía cambios que no estaban en la máquina local.

También subimos cosas que no deberían ir al repo, como db.sqlite3, y había conflictos.

Queríamos seguimiento en vivo del bus, con rutas reales y paradas reales, sin depender de servicios pagos ni internet estable.

SOLUCIONES

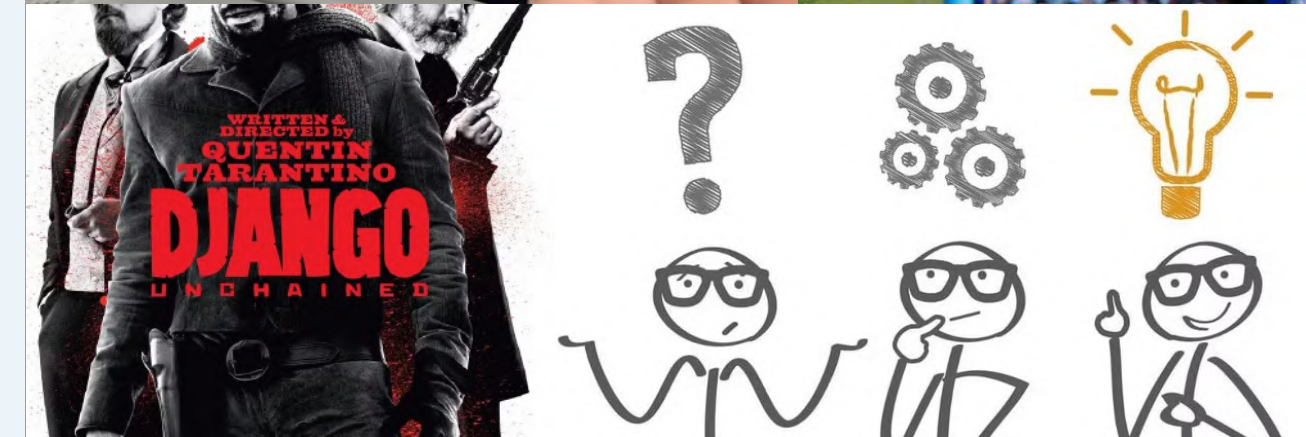
Definimos que todos trabajen en ramas y no directamente en main.

Agregamos .gitignore para evitar subir archivos basura (migraciones rotas, base de datos local, etc.).

Levantamos nuestro propio motor de rutas (OSRM) dentro de Docker, y lo publicamos con ngrok para que cualquiera pueda ver el mapa y el movimiento del colectivo.

CONCLUSIÓN

- Integración completa de Django: modelos, vistas y templates.
- Trabajo colaborativo con Git y organización por tareas.
- Diseño responsive e interfaz clara para el usuario.
- Aprendizaje práctico y resolución de problemas reales.



GRACIAS