Delni - Minimum Viable Product (MVP) Summary

# Project Name

Delni

# Main Features

* Advanced Route Finding: Graph-based (BFS) algorithm for optimal bus routes, multi-leg journeys with transfer support, route deduplication, and geographic validation.
* Route Optimization: Fewest walking and fewest transfers options, category filtering, and automatic sorting.
* Geographic Intelligence: Real-time distance calculations (GeoPy), nearest point and transfer point detection, efficient bounding box queries.
* User Experience: Walking vs. taxi recommendations, detailed journey info (bus lines, boarding/alighting, walking/bus times, transfers).
* Technical Excellence: Comprehensive input validation and error handling, performance optimizations (MongoDB projections, caching, indexing), security features (CORS, session security, production hardening), full test suite (unit, integration, performance, edge cases).

# Technology Stack

* Backend: Django 5.2.4, Django REST Framework, Gunicorn
* Database: MongoDB 6.0, PyMongo, Geospatial Indexing
* Geographic Processing: GeoPy, custom BFS algorithms
* Documentation: drf-yasg (Swagger/OpenAPI), interactive API docs
* Deployment: Docker, Docker Compose, Nginx

# Key Endpoints

* GET /api/graph-route/ – Advanced route finding
* GET /api/health/ – System health monitoring
* GET /api/diagnostics/ – Database/system diagnostics
* GET /api/find-route/ – Legacy route finding
* GET /api/suggest-route/ – Route suggestions with preferences
* /swagger/, /redoc/, /swagger.json – API documentation

# System Architecture (Text Diagram)

User/API Client  
 |  
 v  
Django REST API  
 |  
 v  
Route Finding & Optimization (BFS, GeoPy)  
 |  
 v  
MongoDB (Geospatial)  
 |  
 v  
Swagger / OpenAPI Docs  
 |  
 v  
User/API Client