Software & System Architecture

System Overview

ReUseIt uses a modular, microservices-inspired architecture: ReUseIt uses a modular, microservices-inspired architecture:

- **Mobile App:** Expo React Native client with on-device TensorFlow Lite for waste classification
- **Backend:** NestJS GraphQL API server (business logic, authentication, data processing)
- **Database:** MongoDB Atlas (Prisma ORM, geospatial indexing)
- **Cache:** Redis for performance
- **AI/ML:** Python ML training utilities (YOLOv8, TensorFlow Lite)
- **External:** Firebase (Auth, FCM), Google Maps

Data Flow

1. Mobile app classifies waste on-device, sends results to backend via GraphQL
2. Backend processes requests, queries MongoDB, caches in Redis, integrates with Firebase/Maps
3. AI features use Ollama for enhanced content (Reserved for future AI features)
4. Notifications sent via Firebase Cloud Messaging

Key Design Decisions

- GraphQL for flexible, efficient queries
- Modular backend for maintainability
- On-device ML for privacy/offline support
- Cloud-first DB for scalability

Technical Debt

- Partial test coverage (target 80%)
- CI/CD pipeline incomplete
- Leaderboard UI and educational articles UI pending

Installation & Run Instructions

1. Install APK on Android device
2. Run: `pnpm install` in project root
3. Start backend: `pnpm --filter backend run start:dev`
4. Start mobile: `pnpm --filter mobile run start`