Zan FinTech Solutions – Development Roadmap

# Milestone 1: Server and Development Environment Setup

* - Update Ubuntu server and install required packages
* - Install Docker and Docker Compose
* - Install Git and clone the project repository
* - Set up Tailscale for remote access (optional)

# Milestone 2: Project Bootstrap

* - Initialize Git repository and project folder structure
* - Setup Django backend with apps (users, videos, news, analytics)
* - Install backend dependencies and create initial migrations
* - Initialize React frontend using Vite
* - Install frontend dependencies (axios, react-router-dom)

# Milestone 3: Local Development & API Integration

* - Create Django models, serializers, views, and routes
* - Implement React components and connect to backend APIs using Axios
* - Test CRUD operations and frontend-backend integration locally

# Milestone 4: Dockerize Application

* - Create Dockerfile for Django backend
* - Create Dockerfile for React frontend
* - Write docker-compose.yml to run backend, frontend, and Nginx
* - Test full stack locally using Docker Compose

# Milestone 5: CI/CD Setup with GitHub Actions

* - Create GitHub repository and push local code
* - Create GitHub Actions workflow (deploy.yml)
* - Configure SSH and GitHub secrets for server access
* - Enable auto-deploy on push to main branch

# Milestone 6: Nginx and Reverse Proxy Setup

* - Create nginx/default.conf to serve frontend and proxy API requests
* - Integrate Nginx into docker-compose.yml
* - Test end-to-end deployment using localhost or domain

# Milestone 7: Domain, SSL, and Security Hardening

* - Point domain to server IP or Tailscale IP
* - Install Certbot and configure SSL certificates
* - Configure UFW firewall and Fail2ban for brute-force protection
* - Install Supervisor (optional) to ensure services remain up

# Milestone 8: Monitoring, Logs, and Optimization

* - Set up log rotation for Nginx, backend, and frontend
* - Add monitoring tools like Prometheus, Grafana, or uptime robot
* - Review performance bottlenecks and optimize queries, frontend bundles