

XAKIMOV MUXAMMADAZIZ

FULLSTACK DEVELOPER

+998773151707

khakimovmukhammadaziz@gmail.com

Tashkent/Fergana, Uzbekistan

PROFILE

Motivated FullStack developer with experience in building efficient web applications using Node.js, React.js, NestJS, PostgreSQL, and Prisma. Passionate about algorithms and solving complex problems. Quick to learn new technologies and able to work effectively both independently and as part of a team.

WORK EXPERIENCE

2025, July - Present

Mars IT school, Tashkent

FullStack developer

- Supervised real-world projects involving API integration, routing, and performance optimization in React.
- Teaching and mentoring students in modern frontend development using React.js, JavaScript (ES6+), HTML5, CSS3, and Tailwind CSS.
- Guided students in building scalable React projects using Zustand for state management and TanStack Query for efficient data fetching and caching.
- Implemented i18next for multilingual support and seamless user experience across different languages.
- Instructed best practices in component-based architecture, reusable UI design, and responsive layouts.
- Helped students deploy and maintain production-ready frontend applications following clean code principles.

2024, May - 2025, July

UpWork

BackEnd Developer

- Developed efficient and secure backend systems using Node.js, NestJS, and PostgreSQL.
- Designed and optimized database schemas with Prisma ORM.
- Built and integrated RESTful APIs for front-end applications.
- Wrote unit tests and fixed bugs to enhance code quality and maintainability.
- Implemented new features and optimized existing modules based on client requirements.

EDUCATION

2021 - 2025

FERGANA STATE TECHNICAL UNIVERSITY

- Software Engineering

2024 - 2025

NAJOT TA'LIM

- FullStack developer

SKILLS

- JavaScript
- TypeScript
- Nodejs
- Redis
- Express
- WebSocket
- Nestjs
- Telegram bot
- PostgreSQL/prisma
- MongoDB
- HTML/Css
- System Design
- Nextjs
- ReactJs

PROFESSIONAL LINKS

[GitHub profile](#)

[LinkedIn profile](#)

[LeetCode profile](#)

[YouTube profile](#)

LANGUAGES

- Uzbek(native)
- English (pre Intermediate)
- Russian (elementry)

■ Visitor Track

Visitor Track — Full-stack visitor analytics platform for restaurant chains, tracking real-time foot traffic from IoT people-counting devices across multiple branches. Built with NestJS, PostgreSQL, Prisma ORM, Redis, and React.

- Designed an IoT data ingestion pipeline that receives cumulative in/out counts from entrance sensors, computes hourly deltas with Redis caching, and gracefully handles daily counter resets and device restarts.
- Built a role-based analytics REST API (admin vs. branch access) delivering daily, hourly, and per-branch visitor statistics with optimized Prisma aggregation queries, documented via Swagger.
- Implemented secure JWT authentication with email verification and password recovery flows, request throttling, and automated admin seeding.
- Developed a responsive React + TypeScript dashboard with TanStack Query, Recharts visualizations, dark mode, multi-language support (UZ/EN/RU), and PDF/Excel report export.
- Containerized the full stack (API, PostgreSQL, Redis) with Docker Compose for one-command deployment.

■ Telegram Clone

Telegram Clone — Full-stack real-time messaging application inspired by Telegram, built with Next.js, Express, MongoDB, and Socket.io.

- Developed a secure passwordless authentication flow with email-based OTP verification via Nodemailer and JWT session management using HTTP-only cookies.
- Implemented real-time communication with a dedicated Socket.io service, enabling instant message delivery, live typing indicators, and online status tracking.
- Built rich messaging features including message editing, deletion, emoji reactions, and media sharing with file uploads powered by UploadThing.
- Designed a responsive, modern UI with Next.js 16, React 19, Tailwind CSS, and shadcn/ui, supporting dark/light themes and optimistic updates via TanStack Query and Zustand.
- Structured the backend with a clean MVC architecture, RESTful APIs, and centralized error handling to ensure scalability and maintainability.