JATIN SINGH
Back-end Developer

SUMMARY

I'm a software developer with a strong foundation in building efficient Back-ends. Specialize in creating smooth CRUD operations, optimizing database queries, and implementing secure authentication methods. I'm experienced in designing RESTful APIs and have a solid grasp of core Back-end technologies. I have build many MVPs , DB schemas and CRUD applications.

SKILLS

- Core Python
- Django, Fastapi, Javascript
- MySQL , Postgresql
- MongoDB , Casandra , Mysql
- JWT OAuth2

- Linux
- RESTful APIs
- Cloud
- Networking
- Github

- Deployment
- Documentation
- Communication
- Teamwork
- Management

WORK EXPERIENCE

• Contributed to Glances (open source) P.R.

July 2025

Co-authored and merged a pull request to the open-source project Glances (Linux system monitoring tool with 25K+ GitHub stars), enhancing performance and maintainability.

E-Commerce MVP (freelance) :

November 2024

Tech Stack: FastAPI, PostgreSQL Stripe/Razorpay
Developed a minimal viable product (MVP) for a clothing e-commerce platform featuring secure user authentication, product listing with database integration, and basic payment gateway functionality.Implemented RESTful APIs using FastAPI for user registration, login, and product management.

PROJECTS

Universal Authenticator:

Tech Stack: FastAPI, Authlib, PostgreSQL, JWT, OAuth2, Email, 2FA
Developed a scalable authentication service supporting email, Google, GitHub, and Facebook login.Implemented JWT-based session management, email verification, and TOTP-based two-factor authentication. Added a secure "Banking Mode" toggle to demonstrate advanced security and modular backend

Payment integration:

Tech Stack: FastAPI, Postgresql , Razorpay
Built a REST API microservice integrating multiple payment gateways with transaction recording, verification via webhooks, and real-time payment tracking. Designed a minimal structured backend integration and fintech API understanding.

QR Generator : QRGEN

Tech Stack: Django Jinja2 Render.
A QR code generator built with Django that accepts a link to any image and generates a downloadable QR code. Users can provide a name for the output so the QR image is saved with the specified filename.