

Aditya Chaudhary

+91 9327748938 | aditya123905@gmail.com | LinkedIn | GitHub | Portfolio

Technical Skills

Languages: Python, JavaScript

Frameworks: FastAPI, Django, Django REST Framework, React.js, Frappe (ERPNext), Flask

Database & Caching: PostgreSQL, MySQL, Redis, Elasticsearch, MongoDB, SQLite

DevOps & Infrastructure: Docker, AWS, CI/CD, GitHub Actions, Prometheus, Grafana, Nginx

Experience

Upforce Tech July 2025 – Present

Python Backend Developer

- Designed and developed **REST APIs using FastAPI, Flask, and Django REST Framework** across microservices, applying **DDD principles** to structure services around business capabilities.
- Implemented **CQRS architectures** with **Elasticsearch-based denormalized read models**, improving read performance and reducing complex relational queries and database load.
- Built **event-driven asynchronous workflows** using **Celery, RQ, and Amazon SQS**, including bulk email processing for **10,000+ messages** with worker scaling, retries, and dead-letter queues.
- Optimized backend performance using **PostgreSQL query optimization and Redis caching/rate limiting**, including tenant-aware configuration caching for multi-tenant services.
- Implemented **RBAC and secure service-to-service communication** using token-based validation; maintained **OpenAPI contracts** and automated unit, integration, and API testing with **Pytest and pytest-asyncio**.
- Containerized and deployed microservices using **Docker and Kubernetes (AWS EKS)** with **GitHub Actions CI/CD pipelines**, monitoring services with **Prometheus, Grafana, and Kubernetes logs**

Techseria Pvt Ltd May 2024 – June 2025

Python & React Developer

- Built Frappe ERP modules for a Singapore client, including custom workflows, third-party API integrations, and automated data pipelines — reducing manual operational effort by 70%.
- Developed an Azure Backup Module with configurable scheduling, credential management, and retention policies, fully eliminating manual backup processes.
- Designed RESTful APIs using Django REST Framework with JWT authentication and PostgreSQL-backed data handling to support mobile and client-side applications.
- Implemented RBAC, protected routes, and dynamic dashboards in React with Material UI for seamless Frappe–React integration.

BrainyBeam Technologies Pvt. Ltd Aug. 2023 – April 2024

Python Developer

- Built a Hotel Booking Web App using Django and MySQL, integrating Razorpay for secure end-to-end payment processing.
- Designed RESTful APIs for booking workflows and implemented JWT-based user authentication, order history tracking, and session management.
- Wrote unit tests and maintained modular, documented code; deployed the application with structured version control via GitHub.

Projects

Vine Supplier Chain Observability | Professional project

- Built backend services for **SLR Consulting / RCS Global's supplier-chain observability platform**, supporting analysis and visualization of critical-mineral supply chains.
- Led the migration from a legacy backend to Python/FastAPI, modernizing the application architecture while preserving existing business workflows and functionality.
- Redesigned and implemented REST APIs with FastAPI, migrating legacy endpoints and introducing new APIs for supplier-chain features and application workflows.
- Implemented data-processing pipelines and backend integrations for supplier-chain data, including validation, transformation, and workflow processing.
- Executed database migration and data-model evolution, adapting legacy schemas and data to the new backend while maintaining data integrity and application compatibility.
- Built comprehensive unit, integration, and API test coverage using Pytest and pytest-asyncio, supporting safer migration and continued feature development. Containerized and operated backend services with Docker, maintaining OpenAPI/Swagger contracts and supporting deployment, integration, and production reliability.

XPRIZE Learning Platform | Professional project

- Contributed to a large-scale multi-tenant learning platform within the XPRIZE ecosystem using a microservices architecture.
- Implemented schema-based tenant isolation and tenant-specific configuration for secure multi-tenant data handling.
- Developed learning workflows, progress tracking, content management, and RBAC-based access control.
- Implemented secure service-to-service communication using structured service layers and token-based validation.
- Built event-driven asynchronous workflows to decouple background processing from synchronous application requests.
- Optimized request handling with Redis tenant-aware configuration caching and enhanced security through file validation and region-specific access controls.
- Contributed to containerized infrastructure using Docker and Kubernetes (AWS EKS), with Codefresh CI/CD pipelines across staging, demo, and production environments.

VideosForChange | Professional project

- Contributed to a production-scale platform serving 10,000+ users for video-based social engagement and content discovery.
- Implemented CQRS, separating read/write models and using Elasticsearch as a denormalized read store to eliminate complex joins and improve query performance.
- Implemented background Elasticsearch reindexing on data mutations to keep denormalized read models synchronized.
- Built event-driven asynchronous workflows using Celery and message queues for notifications, CSV imports, cleanup, and search reindexing.
- Developed bulk email delivery for 10,000+ messages using Amazon SQS and worker-based processing with dynamic upscaling and downscaling.
- Implemented retry queues and dead-letter queues with a maximum of three attempts for controlled failure handling.
- Improved reliability and operational visibility across distributed workflows through asynchronous processing, retries, and decoupled execution.

FordgeIndia | Personal project — [Live](#)

- Built a multi-tenant SaaS platform enabling local businesses to create branded storefronts with custom domains, product catalogs, collections, business information, portfolio themes, and direct WhatsApp ordering.
- Developed the core backend with Django REST Framework and React + Vite admin dashboard, implementing authentication, onboarding, product/collection management, portfolio customization, publishing, access control, and subscriptions.
- Designed a distributed read architecture where AWS Lambda generates versioned vendor-specific SQLite databases from PostgreSQL, stores datasets in S3, and publishes Redis events to synchronize new versions with the FastAPI read layer.
- Built a lightweight FastAPI read service over SQLite consuming Redis events and performing atomic database-file replacement, isolating public portfolio traffic from the transactional Django application.
- Implemented version-based caching with Nginx and S3 + CloudFront CDN delivery for the public React application, reducing repeated backend requests and improving read performance.
- Implemented Razorpay subscription/payment processing with verification, subscription tracking, and webhooks; deployed the platform using Docker, Nginx, GitHub Actions, EC2, S3, CloudFront, Lambda, PostgreSQL, Redis, and CloudWatch for application/infrastructure logging.
- **Tech Stack:** Python · Django · Django REST Framework · FastAPI · React · Vite · PostgreSQL · SQLite · Redis · AWS Lambda · Amazon S3 · CloudFront · EC2 · CloudWatch · Nginx · Docker · GitHub Actions · Razorpay

NoteStudio | Personal project — [Live](#)

- Built a full-stack note-taking platform with JWT authentication, token refresh, and blacklist handling for secure session management.
- Implemented public shareable links with visibility controls, allowing notes to be toggled between public and private without requiring recipient accounts.
- Deployed backend on AWS EC2 with Nginx as a reverse proxy, database on NeonDB, and frontend on Netlify; automated deployments via GitHub Actions CI/CD.
- **Tech Stack:** Django, DRF, React, PostgreSQL, Redis, AWS EC2, Nginx, Netlify

Thought Exchanger | Personal Project — [Code](#)

- Developed a social media platform supporting tweets, likes, comments, and a follow system with real-time activity notifications.
- Implemented complete user authentication (register, login, logout, password reset), profile management, and media uploads with a responsive UI.
- **Tech Stack:** Django, Sqlite3, Jinja Template, WebSockets

Education
