

AVINASH JOSHI

☎ +91-7751075367 ✉ joshiavinash00999@gmail.com 🔗 [linkedin.com/in/avinashjoshi28](https://www.linkedin.com/in/avinashjoshi28) 🐙 github.com/Avinash-2803

Summary

Entry-level Software Development Engineer (B.Tech, 2026) with internship experience in software application development, requirements analysis, programming, testing, debugging, and troubleshooting. Skilled in Python, C++, JavaScript, Django REST, microservices, and REST APIs, with hands-on experience across the software development lifecycle including system design, coding, unit testing, API testing, and maintenance. Experienced in collaborating with engineering and product stakeholders to translate requirements into reliable and scalable applications in Agile environments.

Skills

Programming Languages: C++, Python, Core Java, JavaScript, SQL
Backend: Django, Django REST Framework, REST APIs
CS fundamentals: OOP, Operating Systems, Computer Networks, DBMS, Data Structures and Algorithms.
Databases: PostgreSQL, MongoDB, SQLite, Redis
DevOps: Linux, Git, AWS, Kubernetes, CI/CD, Docker, Prometheus, Grafana.

Experience

Full Stack Developer Intern – InnoByte Services

Nov 2024 – Jan 2025

- Collaborated with product and engineering stakeholders to analyse requirements, define application changes, and develop scalable backend functionality using Django REST APIs and React in an Agile environment.
- Designed, implemented, tested, and debugged REST APIs and application workflows, troubleshooting functional and performance issues to improve software reliability .
- Optimized database queries using **Django ORM and Django REST Framework** by reducing average query execution time by 40%.
- **Reduced API response time by 25%** through **query optimization and caching**, improving backend scalability under concurrent requests.

Data Science Intern – The Developers Arena

Mar 2026 – May 2026

- Built and evaluated **Machine Learning classification pipelines** using **Scikit-learn and Pandas**, achieving **92%+ accuracy**.
- Performed **EDA on large datasets**, reducing preprocessing time by 30%.

Projects

Enterprise AI Customer Support Platform | *Django REST, PostgreSQL, React, Redis, Celery, LLM, Docker* | [Link](#)

- Engineered a distributed multi-tier **AI-powered CRM platform with JWT authentication, RBAC authorization**, and responsive **React** dashboards for Customers, Agents, and Administrators supporting end-to-end ticket management.
- **Integrated LLM capabilities** (Google Gemini) into a production workflow for automated ticket classification and priority prediction, evaluating output accuracy and reducing latency by 30% — improving reliability of an AI-enabled governance/support process.
- Designed event-driven, multi-step workflow orchestration using Celery task queues and Redis, supporting asynchronous cross-system processing and improving throughput by 40%.
- **Designed a Scalable backend architecture with Docker, Prometheus, and Grafana** to monitor distributed services at scale, tracking API latency, throughput, and system health .

devPilot: AI-Powered Developer Engineering Platform | *Django REST, React, PostgreSQL, Redis, Docker* | [Link](#)

- Engineered an **AI-powered developer platform** using **Django REST Framework** and **React**, integrating **Retrieval-Augmented Generation (RAG)**, **OAuth authentication**, and background task processing to automate code understanding, repository analysis, and engineering workflows .
- **Designed a scalable backend architecture** with **PostgreSQL, Redis, Celery**, and **Docker**, implementing asynchronous repository indexing, **AI task orchestration, caching**, and secure REST APIs for high-performance developer services that reduce latency by 30% .
- Implemented **retrieval-augmented generation (RAG)** using embeddings and a vector database (ChromaDB) with LangChain, enabling semantic search and contextual Q&A — designed and tuned prompts to balance accuracy, latency, and cost across LLM calls (Groq API) .
- Implemented comprehensive API testing, and role-based access control to improve system reliability, observability, and maintainability.

collabCode | *React, JavaScript, Node.js, Docker, AWS* | [Live Link](#)

- Built a browser-based collaborative code editor using React and Monaco Editor, allowing multiple **users to edit the same JavaScript document in real time**.
- **Integrated Yjs and Socket.IO** for synchronized editing and live user presence, with an Express server serving the frontend and a health-check endpoint.
- Created a **multi-stage Docker image** that builds the Vite frontend and packages its static assets with the Node.js backend, using Node.js 20 Alpine .
- Deployed the containerized application on AWS [**ECR, ECS**] by provisioning the host, installing Docker, building and running the image, and configuring network access to serve the app.

Education

International Institute of Information Technology Bhubaneswar

B.Tech Electronics And Telecommunication

2022-2026

CGPA: 7.61/10