

Avadhut Kaskar

Mumbai, India | avadhutkaskar@gmail.com | +91-9152415268 | github.com/Avad05

EDUCATION

Smt. Indira Gandhi College of Engineering

BE in Computer Engineering

August 2023 – May 2027

GPA: 8.7 / 10

- Relevant Coursework: Data Structures & Algorithms, Operating Systems, DBMS, Object-Oriented Programming

EXPERIENCE

Open Source Software Engineering — Odysseus (Ai Agent Workflow)

June 2025 – Present

Remote

- Implemented Backend optimization for UI timing by replacing static delays with dynamic character-length formulas, improving UX and readability across high-frequency notifications.
- Enhanced Codebase maintainability by navigating complex PR workflows and multi-round reviews in a large-scale codebase with strict architectural standards.
- Caught and reverted an unrelated security issue (suspicious docker-compose.yml change) during code review before submitting the PR.

PROJECTS

PR Synthesizer [GitHub](#)

2025

Node.js · Python · FastAPI · PostgreSQL · pgvector · Redis · BullMQ · React · Docker · A2A Protocol · Gemini · OpenRouter

- Architected a production-grade GenAI orchestration system using A2A protocol to automate security and performance audits, resolving 10+ critical issues across multiple repositories.
- Engineered a high-fidelity RAG pipeline with pgvector (HNSW indexing, 3072-dim embeddings), enabling agents to perform deep codebase analysis beyond generic LLM warnings.
- Designed a framework-agnostic agent network across Node.js and Python/FastAPI, connected via JSON-RPC 2.0, allowing any agent to be swapped without modifying the orchestrator.
- Optimized Asynchronous systems using BullMQ and Redis, ensuring sub-100ms response times for GitHub webhooks regardless of intensive agent processing loads.
- Added automatic OpenRouter → Gemini fallback with a 20 s AbortController timeout per agent, maintaining 100% review completion rate even when the primary LLM provider is unavailable.
- Packaged all 7 services (orchestrator, 3 agents, Postgres + pgvector, Redis, React dashboard) in a single docker-compose.yml, reducing setup from 30+ minutes to a single command.

Emergency Response Coordination System [GitHub](#) | [Demo](#)

2025

React · Node.js · Express · Render

- Applied rigorous System Design principles to build a scalable emergency dispatch MVP, managing complex incident lifecycles from citizen reporting to real-time unit resolution.
- Designed a Scalable architecture featuring priority-based selection algorithms for optimized unit allocation across diverse emergency categories and geographic locations.
- Built a real-time incident status tracker (Pending → Dispatched → Completed) with an operator dashboard for live review and manual dispatch control.

Blog-API [GitHub](#) | [Client](#) | [Author](#)

2025

Node.js · Express · PostgreSQL · Prisma · Cloudflare Pages · JWT · bcrypt

- Architected and deployed a full-stack blogging platform with 7 RESTful API endpoints across reader and author clients, live on Cloudflare Pages.
- Implemented JWT-based authentication with bcrypt password hashing and role-based access control, separating public reader and protected author routes.
- Designed a relational schema with Prisma ORM, supporting nested comment threads, draft/published post states, and file upload integration.

BattleShip [GitHub](#) | [Live Demo](#)

2024

JavaScript · DOM API · Jest

- Engineered a turn-based Battleship game with a 10×10 grid, applying OOP to model ships, boards, and player actions with full placement validation and win detection.
- Achieved full test coverage of core game logic using Jest, validating hit/miss detection, ship sinking, and boundary conditions before any UI integration.

TECHNOLOGIES

Languages: JavaScript, Python, SQL, C++, C

Frontend: React.js, Vite, Tailwind CSS, Recharts, SSE

Backend: Node.js, Express.js, FastAPI, BullMQ, JWT, bcrypt, Passport.js

Databases: PostgreSQL, pgvector, Redis, Prisma ORM

AI / ML: Gemini API, OpenRouter, A2A Protocol, RAG, Vector Embeddings

Infrastructure: Docker, Docker Compose, Cloudflare Pages, Git, GitHub, Postman, Linux