

Hariom Pandey

pandeyhariom352@gmail.com | linkedin.com/in/hariom-pandey1 | github.com/HariomPCode

EDUCATION

University of Mumbai

B.E. in Artificial Intelligence & Machine Learning | CGPA: 8.5/10

Mumbai, India

Aug. 2023 – May 2027 (Expected)

- **Coursework:** Data Structures & Algorithms, Object-Oriented Programming, Operating Systems, Database Management Systems, Computer Networks

TECHNICAL SKILLS

Languages: Java, TypeScript, JavaScript, Python

Backend & Data: Node.js, Express.js, REST APIs, Redis, BullMQ (async job processing), MongoDB, PostgreSQL

Infrastructure & Tools: Docker, Git, GitHub, Google APIs, OAuth 2.0, JWT

AI Integration: OpenAI API (GPT-4o-mini), function calling / structured outputs

Frontend: React, Next.js, Tailwind CSS

AI-Assisted Engineering: ChatGPT for debugging, error analysis, implementation reasoning, and iterating on fixes against observed errors and behavior

Practices: Unit and API testing, input and path validation, GitHub pull-request workflow

PROJECTS

Koder – Online Judge Platform | TypeScript, Next.js, Express, MongoDB, Redis, BullMQ, Docker | [GitHub](#) 2026

- Architected a three-service system (Next.js/TypeScript frontend, Express REST API, independent BullMQ worker) that decouples submission handling from code execution via a Redis-backed job queue; the API returns a job ID immediately and clients poll for asynchronous grading results.
- Built a Docker sandbox that runs untrusted submissions in ephemeral node:20-alpine containers with no network access, all capabilities dropped, a read-only filesystem, a 256MB / 1-CPU / 64-process ceiling, and a 3-second timeout enforced via SIGKILL, containing malicious or runaway code.
- Designed a MongoDB schema and grading pipeline that evaluates each submission against multiple test cases and stores per-test execution results; structured to extend to additional programming languages.

Tasky – AI Task Extraction Platform | Next.js, MongoDB, Google APIs, OpenAI API | [GitHub](#) 2025

- Designed an AI extraction pipeline that converts unstructured Gmail, Google Calendar, and Google Tasks data into structured task objects using OpenAI GPT-4o-mini with function calling for consistent output.
- Integrated the Gmail, Google Calendar, and Google Tasks APIs into a unified dashboard, with Clerk authentication and OAuth 2.0 for secure account access.

EduTest – Mock Exam Platform | React.js, Node.js, Express, MongoDB | [GitHub](#) 2025

- Built a MERN-stack entrance exam simulator with an Express REST backend, JWT authentication, and server-tracked progress state, covering 10+ previous-year papers across 3 subjects.
- Designed a performance-analytics engine that aggregates per-topic accuracy across multiple submissions using MongoDB aggregation.

OPEN-SOURCE CONTRIBUTIONS

career-ops | Open-source AI job-search tool | Node.js, TypeScript, Next.js | [Repository](#) 2026

- Built a read-only Documents dashboard ([PR #4958](#), merged): a discovery module and a `GET /api/documents` endpoint that lists generated CV and cover-letter PDFs and serves them with path-containment checks, symlink protection, and descriptor-based file reads, plus a page with loading, empty, and error states.
- Added automated tests for document discovery and API behavior and iterated through maintainer and automated code review (8 commits): separated missing-file (ENOENT) errors from other read failures, hardened download-filename encoding, and reused the shared manifest-parsing logic.
- Fixed CLI argument handling in `invite-match.mjs` ([PR #2958](#), merged): added a usage block and `--help/-h` support and made the script reject unknown or mistyped flags instead of silently ignoring them, without validating the arguments of scripts that import it.

EXPERIENCE

Frontend Developer Intern, Legalo Technologies Services LLP (Legalo.ai)

Dec. 2025 – Mar. 2026 | Remote

- Built 5+ reusable Next.js components for Legalo.ai's production platform, replacing duplicated UI and improving code reuse across multiple features.
- Refactored client-side state with Redux Toolkit, centralizing shared state and simplifying predictable updates across 3 components.
- Implemented responsive layouts and optimized image delivery with next/image to improve frontend performance.

ACHIEVEMENTS

- Solved 638 LeetCode problems (206 Easy, 323 Medium, 109 Hard) with a 1,671 contest rating (top 16.52% contest-rating percentile); 362 active days and a 287-day maximum submission streak.
- Won three inter-college programming competitions, including two 2nd Prize finishes at Tantrotsav 2024 and 2nd Runner-up at CodeSprint (Vidyavardhini's College of Engineering and Technology).