

Hardik Mahajan

+91 72494 61177 | hardikmahajan97@gmail.com | [linkedin.com/in/hardik-mahajan](https://www.linkedin.com/in/hardik-mahajan) | github.com/HardikMahajan97

PROFESSIONAL SUMMARY

Pre-final year B.Tech CS student with **production-grade experience** building distributed backend systems, real-time data pipelines, and AI-integrated services. Fluent across the full stack, seeking to apply cross-domain technical depth to solve large-scale engineering challenges.

EDUCATION

Pimpri Chinchwad College of Engineering, Pune
Bachelor of Technology in Computer Engineering

2023 – 2027
CGPA: 7.91/10.0

TECHNICAL SKILLS

Languages: Python (Advanced), Java, JavaScript, C++, SQL, Bash/Shell

Frontend: React.js, HTML5, CSS3, Tailwind CSS, Bootstrap, Material-UI, Responsive Design

Backend & APIs: Node.js, Express.js, Spring Boot, FastAPI, RESTful APIs, Microservices Architecture

Data Eng. & Streaming: Kafka (consumer/producer), Distributed Pipelines, Stream & Batch Processing, ETL

ML & AI: Scikit-learn, PyTorch, LangChain, Pandas, NumPy, RAG, Autonomous Agents, Vector Embeddings

Databases & Caching: PostgreSQL, MongoDB, MySQL, Redis, Query Optimization, Indexing, NoSQL

Cloud & DevOps: AWS (EC2, ECS, Fargate, S3, RDS, ElastiCache, SES), Docker, Git, Linux CLI, Vercel, Render

Auth & Security: JWT, OAuth, Passport.js, Session Management, OWASP Top 10, RBAC

Core CS: DSA, OOP, System Design, Operating Systems, DBMS, Computer Networking, MVC Architecture

PROFESSIONAL EXPERIENCE

Backend & Data Engineering Intern

May 2025 – Jan 2026

ParikSaathi — EdTech B2B SaaS Platform

Pune, India

- Architected **production-grade microservices** across 20+ modules handling 100,000+ data records; eliminated redundant DB queries via **indexing, pagination, and Redis caching** — achieving **65% reduction in API response time** and **98% performance improvement** under peak traffic.
- Engineered a **multi-stage parallel data pipeline** fetching analytics concurrently per student across multiple sources, reducing per-student fetch latency **from 3s to 1s** — a **70% end-to-end improvement** backed by a distributed caching layer cutting DB read load by **40–60%**.
- Built a **cross-language document processing system** (Python + Java) handling 200+ answer sheets per batch: computer vision for **QR detection at 300 DPI**, fire-and-forget async design for **sub-500ms API responses**, and parallel concurrent uploads to **AWS S3** achieving **5–10× throughput**.
- Built an **end-to-end AI report generation pipeline** via **OpenAI Batch API** — prompt construction, job submission, background polling, result retrieval, and structured parsing — processing **500+ reports per batch at 50% lower cost** with **100% parse success rate**.
- Engineered **RBAC** across 30+ REST endpoints with **JWT authentication**, handling **100+ concurrent requests** at sub-200ms; built bulk email notification system delivering to **1,000+ users** at 100% delivery rate.

Frontend Developer

Sep 2024 – Apr 2025

PCCOE ACM Student Chapter

Pune, India

- Developed fully responsive UI with **React.js + Tailwind CSS** serving 500+ active users; refactored codebase to component-based architecture, reducing code duplication by **10%**.
- Implemented **progressive image rendering with lazy loading**, improving **Lighthouse score from 72 to 87** and cutting page load time by **40%**.

PROJECTS

FinoHelp — Banking Fraud Assist Chatbot | *Python, BART-MNLI, FAISS, XGBoost, SQLite*

- Built a **multi-model fraud detection pipeline** combining zero-shot NLP classification (**BART-large-MNLI**) across **10 fraud categories**, semantic **RAG retrieval** (FAISS + sentence-transformers) over a regulatory knowledge base, and ML anomaly detection (**IsolationForest + XGBoost**) with **SHAP explainability** for per-feature risk breakdowns.
- Engineered a **composite risk-scoring engine** weighting NLP confidence, severity, transaction risk, and behavioural signals into a single score driving **AUTO_BLOCK/HOLD/FLAG** actions; built a stateful **dialogue manager** with structured evidence collection across **30+ evidence fields** and session persistence in SQLite.
- Designed a **rule-augmented anomaly detector** flagging just-under-threshold amounts, off-hours transactions, new beneficiaries, and geo-velocity anomalies; delivered a **Rich terminal CLI** with live risk-tier dashboards, SHAP breakdowns, and unique escalation reference IDs at **sub-500ms interactive turns**.

Real-Time Credit Card Fraud Detection | *Python, PyTorch, GNN, Transformer, Graph-RAG*

- Architected a **fusion fraud detection system** combining a heterogeneous **GNN** (card/merchant/device entity graph, 3 edge types) and a **Temporal Transformer** (self-attention over per-card sequences) into a unified Fusion MLP, achieving **97.1% accuracy** and **0.75 ROC-AUC** on 200,000 synthetic transactions.

- Engineered a **streaming inference pipeline** with rolling per-card feature windows, **Platt-calibrated** probabilities, and F1-optimised decision thresholds (tuned to 0.228) delivering **sub-100ms latency** per event.
- Implemented **Focal loss and Weighted Random Sampler** to handle extreme class imbalance (~2% fraud rate) with early stopping on PR-AUC; built a **Graph-RAG explainability layer** retrieving top-K similar fraud embeddings by cosine similarity to generate human-readable investigation summaries for every flagged event.

PlayWise — MERN Marketplace | playwise-frontend.vercel.app | *React, Node.js, Express, MongoDB*

- Full-stack marketplace with **Passport.js dual-strategy auth**, Twilio OTP, and Nodemailer notifications; deployed across **Vercel + Render** with secure CORS and HTTPS session cookies.
- Optimised MongoDB **aggregation pipelines** eliminating N+1 queries; reduced API response times by **60%** via pagination and field projection.

Central Placement Server Application | user-store-app.vercel.app | *React 18, Node.js, MongoDB, Cloudinary, JWT, Vercel*

- Built full-stack placement portal with **Student and Admin dashboards**; engineered a **serverless-optimised global DB connection cache** cutting warm-start latency **from ~500ms to under 5ms**.
- Integrated **stream-based Cloudinary uploads** bypassing local disk/memory; smart middleware kept **health-check responses under 2ms** even during DB downtime.

CERTIFICATIONS & ACHIEVEMENTS

Certifications: GeeksforGeeks — Generative AI: 21 Days, 21 Projects | SimpliLearn — ML with Python | Apna College — Full Stack Web Dev | Udemy — Java Masterclass

Copyright: *Circular Economy Incentive Platform* — Registered platform incentivising donation, recycling, and exchange activities.

Research Paper: *Credit Card Fraud Detection using H-GNN and Temporal Transformer using Graph Augmented Retrieval* — Paper accepted at ICICV 2026 conference..