

Maharshi Nahar

maharshinahar10@gmail.com | +91 97136 00511 | maharshinahar.in | GitHub | LinkedIn | Codeforces | CodeChef

EDUCATION

Indian Institute of Information Technology, Pune
B.Tech in Computer Science & Engineering

Aug 2024 – May 2028

EXPERIENCE

Software Development Intern Jun 2026 – Jul 2026

Maa Narmada Mahavidyalaya, Dhamnod (M.P.)

- Joined without a spec; spent the first week with office staff mapping their workflows, then designed and shipped three systems that are still in daily use after handover.
- Built an AI grievance redressal bot on WhatsApp answering the questions the office fielded dozens of times a day, escalating to an administrator with the conversation history attached.
- Made its learning safe: after each escalation the system drafts an answer and sends it to an admin for approval over WhatsApp before using it, so nothing enters the knowledge base unapproved. Answers are grounded in a fixed knowledge base; with no source it escalates rather than inventing one.
- Evaluated against a test set built from real parent messages, re-run on every knowledge base change; logged production failures back into routing rules.
- Shipped a bulk WhatsApp platform sending notices and reports addressed by department and role, with attendance for 1,000+ students a day feeding it directly. Queued dispatch with backoff and per-recipient delivery state so failures retry without resending to everyone.
- Built an inventory system for institutional assets, plus RBAC with per-module and per-database grants, single-device sessions, CSRF protection on every mutating request, and idempotent schema migrations.

Tech: Node.js, Express, MySQL, WhatsApp Business API, LLM APIs, HITL approval, guardrails, evaluation

PROJECTS

RAION — Self-Hosted Multi-Agent Assistant Python · LangGraph

- Self-hosted AI assistant (FastAPI + LangGraph) where a supervisor agent spawns parallel workers, each restricted to an explicit tool allowlist rather than the full toolset.
- Bounded every run against runaway loops with hard caps on recursion depth, agent count, tool calls and wall-clock time.
- Built a permission layer classifying each tool call auto or ask; an "ask" calls interrupt() to checkpoint the graph mid-run to SQLite and wait for approval in the UI, then resumes from the same point. Destructive shell commands are pattern-matched and always gated.
- Integrated WhatsApp, Telegram, Gmail, Drive and Calendar, plus MCP connectors, a local vector store for retrieval, and natural-language automations compiled to cron, Gmail or filesystem triggers.

Tech: Python, FastAPI, LangGraph, SQLAlchemy, SQLite, MCP, RAG, OAuth2, Fernet

BitTorrent Client from Scratch Systems · Networking

- Full BitTorrent client in pure Python, standard library only. Downloaded a complete Ubuntu ISO from a live swarm: 24,868 pieces, every one SHA-1 verified.
- Wrote the whole stack by hand: bencode parser and encoder, HTTP tracker announce with compact peer parsing, the 68-byte handshake, and the peer wire message protocol over raw sockets.
- Ran 30 concurrent peer threads against one shared piece queue guarded by a single lock, with failed pieces released back to the pool so a bad peer never costs a piece of the file.

Tech: Python (socket, struct, hashlib, threading), binary protocols, concurrency

Deep Q-Learning Snake Agent Reinforcement Learning

- Trained an agent to play Snake with no movement rules in the code, using an eleven-value state of danger, heading and food-direction flags, all relative to the snake's own heading so one lesson transfers across all four orientations.
- Implemented the Bellman update and replay buffer directly in PyTorch, training on both the fresh transition and a random batch sampled from history to break temporal correlation.

Tech: Python, PyTorch, Q-learning, Pygame, NumPy

Duplicate Question Classifier NLP · Deep Learning

- Semantic duplicate-question classifier on the Quora Question Pairs dataset using SBERT embeddings and a custom PyTorch network, reaching 88%+ accuracy.
- Engineered feature vectors combining embeddings, absolute differences and dot products, behind an NLP preprocessing pipeline for contractions, HTML and punctuation.

Tech: Python, PyTorch, SBERT, Scikit-learn, NLTK, HuggingFace

TECHNICAL SKILLS

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

Frameworks: FastAPI, LangGraph, LangChain, Node.js, Express, PyTorch, Scikit-learn

Applied AI: LLM applications, multi-agent systems, RAG, guardrails, evaluation, human-in-the-loop design, prompt engineering

Databases: MySQL, PostgreSQL, SQLite, ChromaDB

Tools: Git, Docker, Postman, WhatsApp Business API, MCP

Coursework: Data Structures & Algorithms, OOP, DBMS, Operating Systems, Computer Networks, Software Engineering

ACHIEVEMENTS & CERTIFICATIONS

- Codeforces Pupil (peak 1306) · CodeChef 3-Star
- Machine Learning Specialization — DeepLearning.AI / Coursera
- Natural Language Processing Specialization — DeepLearning.AI / Coursera