Jagadish Sai Ram Kancharla Palli

Vijayawada, Andhra Pradesh, India — +91 7893767639 — jagadeeshsairam007@gmail.com
github.com/ZagguChad — linkedin.com/in/jagadish-sai-ram

Summary

Computer Science undergraduate with hands-on experience building GPU-accelerated AI inference systems, embedded C firmware on 32-bit microcontrollers, blockchain-backed applications, distributed real-time backends, and multi-service AI pipelines. Skilled in integrating and debugging software across the full stack – from hardware abstraction layers and OS-level concurrency to frontend interfaces and cloud deployment.

Education

Amrita Vishwa Vidyapeetham – B.Tech, Computer Science and Engineering **Aug 2023 – Present**
CGPA: 8.34 / 10.0 (No Active Backlogs) *Amaravati, AP, India*

- Coursework: Computer Architecture and Organization, Operating Systems, Compiler Design, Digital Logic Design, Data Structures and Algorithms, DBMS, Computer Networks, Distributed Systems, Machine Learning, Discrete Mathematics
- Self-study: x86 instruction set architecture, memory hierarchy and cache design, GPU execution model (CUDA), pre-silicon and post-silicon software development concepts

Sri Srinivasa Gravity Higher Secondary – Mathematics, Physics, Chemistry **Jun 2021 – Apr 2023**
Percentage: 96.6% *Vijayawada, AP, India*

Technical Skills

Systems and Embedded: C, C++ (systems-level), Python, Assembly fundamentals, ESP32 (ARM MCU), GPIO, interrupts, firmware development, memory management, CUDA GPU programming model, OS concepts (scheduling, concurrency, memory hierarchy)

Computer Architecture: x86 ISA fundamentals, cache hierarchy, CPU/GPU execution models, ONNX Runtime (CUDA/cuDNN), multi-threading (ThreadPoolExecutor, QThread, QMutex), dynamic library loading, pre/post-silicon awareness

AI and Machine Learning: ONNX Runtime, InsightFace (ArcFace/SCRFD), Ollama (Qwen3 30B), Gemini API, OpenAI Whisper, Claude API, RAG pipelines, ChromaDB, Scikit-learn, XGBoost, PyTorch, CNN/CNN-LSTM, prompt engineering

Blockchain: Solidity, Hardhat, Ethers.js, smart contract design and local deployment

Compilers and Tools: LLVM (analysis), Git, GitHub, Docker, Linux, Bash, VS Code, GDB

Full-Stack Development: Node.js, Express, FastAPI, React.js, TypeScript, REST APIs, WebSockets, Socket.IO, SQLite, MongoDB, Firebase, Playwright

Projects

FaceFinder – GPU-Accelerated AI Inference Engine *Python, ONNX Runtime, CUDA, cuDNN, PySide6*

- Built a GPU-accelerated face recognition system integrating InsightFace (ArcFace + SCRFD) via ONNX Runtime, managing GPU memory allocation, execution provider selection, and automatic CPU fallback
- Engineered a multi-threaded compute pipeline using ThreadPoolExecutor for parallel I/O, QThread for GPU inference isolation, and QMutex/QWaitCondition for thread-safe synchronization

BlockRx – Blockchain-Based Digital Prescription System *React, Node.js, MongoDB, Solidity, Hardhat, Gemini API*

[GitHub]

- Built a decentralized prescription management platform with Solidity smart contracts deployed on a local Hardhat blockchain to make prescription issuance and dispensing tamper-proof and auditable
- Integrated Google Gemini 1.5 Flash for voice-to-form prescription entry, letting doctors dictate patient and medication details that are parsed and auto-filled via natural language processing
- Designed role-based dashboards for doctors (issuing) and pharmacies (dispensing) on a full MERN-style stack connected to the chain via Ethers.js

Traffic Accident Severity Prediction – ML/DL Replication Study *Python, Scikit-learn, XGBoost, PyTorch*

[GitHub]

- Replicated and extended a published research methodology for predicting traffic accident severity on a 27,540-record real-world dataset (Leeds City Council, 2009-2019), addressing severe class imbalance between fatal/serious and slight-injury outcomes
- Implemented and benchmarked six models spanning classical ensembles (Decision Tree, Balanced Random Forest, XGBoost, Hard/Soft Voting Classifiers) and deep architectures (1D-CNN, CNN-LSTM)
- Outperformed the original paper's published accuracy across every model tested, reaching 88.2% accuracy with a 1D-CNN versus the paper's 87.4% baseline

Secure RFID Authentication on ESP32 *C, ESP32 (32-bit ARM MCU), RSA Cryptography, GPIO*

- Wrote embedded firmware in C for a 32-bit ARM microcontroller: configured GPIO registers, implemented interrupt-driven RFID communication, and managed manual heap allocation
- Validated RSA key exchange at the byte level and debugged timing failures between RFID reader and MCU I/O, verifying correctness against a formal FSM specification

Compiler Bug Clustering – LLVM Subsystem Analysis *Python, Scikit-learn, TF-IDF, K-Means, PCA*

[GitHub]

- Applied unsupervised ML (TF-IDF + K-Means + PCA) to cluster real LLVM bug reports across frontend, backend, and optimizer subsystems, surfacing recurring failure patterns

Motion Arena – Distributed Real-Time System *Node.js, Socket.IO, WebSockets, TLS, Linux*

[GitHub]

- Designed a distributed system handling concurrent multi-client sessions over WebSockets with TLS-secured connections and per-process session state management
- Diagnosed and resolved a critical production failure (mobile sensor permissions block) through root-cause isolation; shipped live at Hack Amrita with zero critical failures

POMpil – AI Agent with Local LLM Inference *Python, FastAPI, Ollama (Qwen3 30B), ChromaDB, RAG*

- Built a fully local AI agent: LLM inference layer via Ollama (Qwen3 30B), a RAG retrieval pipeline (ChromaDB + Sentence-Transformers), and Playwright-based browser automation – with zero cloud dependency

LinkedIn Connect/Accept Automation *Python, Playwright*

[GitHub]

- Built a browser automation tool that accepts pending LinkedIn invitations and sends new connection requests using persistent saved sessions and randomized human-like interaction timing

Experience

Microsoft Learn Student Ambassadors (MLSA) – Amrita Vishwa Vidyapeetham **Aug 2023 – Aug 2025**
Head Executive, Media & DataCamp Donates Program Lead *Amaravati, AP*

- Initiated and coordinated delivery of technical campaigns across an 8-member team, achieving 100% on-time delivery
- Led the DataCamp Donates programme for 50+ students, collaborating across teams on day-to-day execution and resolving participant issues in real time