Gopal Krishna JS

+91 8050283766 | gopalkrishnajs.mca25@rvce.edu.in | linkedin.com/in/gopalkrishnajs | github.com/Kr1szz
Bengaluru, Karnataka

Summary

MCA student with strong foundations in **Computer Networks, Data Structures, and Software Engineering**. Proficient in **Python, Java, and C**, with hands-on experience in designing scalable architectures and network simulations. Certified in AI and data analytics, seeking to leverage technical expertise in backend development and systems engineering.

Technical Skills

Languages: Java, Python, C/C++, SQL (MySQL), JavaScript, HTML/CSS, R
Frameworks: React, Node.js, FastAPI, Flask
Developer Tools: Git, Docker, Linux, Google Cloud Platform, VS Code, IntelliJ
Libraries: pandas, NumPy, Matplotlib

Education

R V College Of Engineering — Bengaluru
Masters in Computer Applications — *Nov. 2025 – 2027*
Relevant Coursework: Data Structures & Algorithms, Computer Networks, Operating Systems, DBMS.

Sri Bhuvanendra College — Karkala
Bachelors in Computer Application — *Sept. 2022 – June 2025*

Projects

SD-WAN Hospital Network Simulator | *Python, Flask, React, Vite* — Jan 2026 – Present

- Engineered a full-stack network simulator utilizing **REST APIs** for real-time state synchronization between a Flask backend and React frontend.
- Implemented **Shannon Entropy algorithms** to detect traffic anomalies, successfully identifying DDoS patterns with high accuracy under simulated loads of **1,000+ packets/sec**.
- Developed an interactive topology visualizer using React Flow, reducing rendering latency for dynamic network state updates to **sub-100ms**.
- Modeled complex network constraints (bandwidth throttling, packet loss) to analyze QoS requirements for critical hospital departments like ICU and Radiology.

Java RPG Engine (Dungeon Crawler) | *Java, Maven, OOP* — Dec 2025 – Jan 2026

- Architected a scalable 2D game engine applying **SOLID principles** and **Design Patterns** (Singleton, Factory, and Observer) to decouple rendering from game logic.
- Optimized the rendering loop to maintain steady frame rates while handling multiple entity updates and collision detection checks per frame.
- Implemented a modular entity component system allowing for easy extension of new enemy types and weapons without refactoring core code.
- Integrated persistent state management to serialize and save player progress (inventory, XP) locally.

Certifications

- **Academic Process Mining Fundamentals** — Celonis (2024)
- **Introduction to LLMs and Prompt Design** — Google Vertex AI (2024)
- **Microsoft Fabric and Data Analytics** — Microsoft (2024)