

# NAVEEN HEMANTH KOKKONDA

Hyderabad, India

☎ +91-6300296968 | ✉ naveenhemanth4@gmail.com | 🔗 linkedin.com/in/hemanthkokkonda | 🌐 github.com/hemanth090 | 🌐 hemanthk.dev

## Summary

Final-year Computer Science student with hands-on experience building AI applications using LLMs and Python. Skilled in developing RAG systems, working with APIs, and solving problems with data structures and algorithms. Interested in generative AI and its applications in automation.

## Education

<b>Sreyas Institute of Engineering and Technology</b> <i>B.Tech in Computer Science (Data Science) — CGPA: 7.46/10</i>	Oct 2022 – Jun 2026 <i>Hyderabad, India</i>
<b>Sri Chaitanya Jr College</b> <i>Intermediate (MPC) — 85.9%</i>	Jun 2020 – Mar 2022 <i>Khammam, India</i>

## Technical Skills

**Languages:** Python (Pandas, NumPy), JavaScript  
**Core CS:** Data Structures, Algorithms, Problem Solving  
**AI/ML:** LangChain, LLMs, RAG systems, Vector databases (FAISS), Sentence Transformers  
**Frameworks:** Flask, Streamlit, React, Node.js, Express.js  
**APIs & Tools:** Groq API, OpenAI API, REST APIs, PyPDF2  
**Development:** Git, VS Code, Jupyter Notebooks, Docker (Basic)  
**Databases:** MongoDB, SQL  
**Math:** Linear Algebra, Probability, Statistics

## Projects

<b>KnoRa AI - RAG Knowledge Assistant</b> 🤖   🔗 <i>Technologies: Streamlit, FAISS, Sentence Transformers, Groq LLMs, LangChain, Docker</i> <ul style="list-style-type: none"><li>– Built a RAG system that uses FAISS for vector search and transformer embeddings to answer questions from documents.</li><li>– Implemented text chunking to preserve context and used batch processing for generating embeddings efficiently.</li><li>– Created a Streamlit interface with analytics to track queries and response times, and containerized it with Docker.</li></ul>	<b>Aug 2025</b>
<b>ATS-V3 - AI-Powered Resume Analyzer</b> 🤖   🔗 <i>Technologies: Flask, Groq AI, PyPDF2, MongoDB, Bootstrap, Render</i> <ul style="list-style-type: none"><li>– Developed a Flask web app using Groq AI to analyze resumes against job descriptions and generate relevance scores.</li><li>– Added PDF parsing with PyPDF2, drag-and-drop file uploads, and MongoDB to store analysis history.</li><li>– Deployed on Render with Gunicorn and implemented light/dark theme toggle for better user experience.</li></ul>	<b>Mar 2025</b>
<b>LegalEase - AI Legal Document Simplifier</b> 🤖   🔗 <i>Technologies: React, Node.js, Express.js, TailwindCSS, Groq LLM, OCR, Render</i> <ul style="list-style-type: none"><li>– Built a full-stack app that simplifies legal documents into plain language and translates them into 77+ languages.</li><li>– Created a pipeline that extracts text from PDFs using OCR, simplifies it with Groq LLM, and handles translations.</li><li>– Designed the frontend with React and TailwindCSS, with features to copy or download the simplified text.</li></ul>	<b>Jul 2025</b>

## Achievements

- Ranked in **Top 5** in two inter-college hackathons for AI-based projects.
- Built AI tools achieving over **85% accuracy** in resume scoring and feedback generation.

## Certifications

- Oracle Cloud Infrastructure 2025 Certified Generative AI Professional (Aug 2025) 🔗
- Simplilearn – Front End Development (Jun 2025) 🔗
- MongoDB Python Developer Path (Jan 2024) 🔗