ANWAR KADER

aikader@uchicago.edu | (704)-705-0522 | GitHub | LinkedIn

EDUCATION

The University of Chicago, Chicago, IL
Bachelor of Sciences in Computer Science, Expected: June 2028; GPA: 3.84/4.0

- Coursework: Discrete Mathematics, Systems Programming I, Mathematical Methods for Economic Analysis

TECHNICAL SKILLS & CERTIFICATIONS

Languages: Python, JavaScript, HTML, CSS, C#
Frameworks and Libraries: Flask, React, Next.js, TensorFlow, Bootstrap
Databases & Tools: MongoDB, Supabase, Git, Unity
Cloud Technologies: AWS (Lambda, API Gateway, EC2, VPC, ELB, Route 53, CloudWatch, EventBridge)
Certificates: GIAC Foundational Cybersecurity Technologies Certificate (SANS Institute), Machine Learning Specialization (Stanford Coursera/Deeplearning.AI), Professional Certificate for IBM Python Data Science (edX), Full-Stack Web Development with React (HKUST Coursera)

EXPERIENCE

The University of Chicago, Chicago, Illinois
ACM Board Member, February 2025 – Present

- Co-lead the Software Engineering Committee, organizing weekly meetings for the **over 70 members** to teach data structures, algorithms, and technical interview preparation, enhancing members' career readiness and coding skills.

SUPERgroup Research Collective, Chicago, Illinois
Research Assistant, April 2025 – Present

- Integrated the Anthropic **Claude LLM API** and later engineered and refined prompt templates to structure over 200 entries of previously unformatted data, supporting an ongoing PhD research project on privacy and security.
- Developed a semi-automated pipeline in Python to assist in labeling and analyzing **over 1,000 data entries** from a dataset of 3,000+, significantly accelerating research workflow while maintaining data integrity.

Research Computing Center's Data Visualization Laboratory, Chicago, Illinois
Visualization lab Assistant, April 2025 – Present

- Extracted data from large NetCDF4 climate datasets using Python (XArray, Pandas), converting 10+ atmospheric variables into optimized formats for visualization
- Optimized system performance by **reducing application load time by over 95%**, from 20 minutes to 40 seconds, through object pooling, efficient memory management, and minimizing costly instantiations of data points
- Accepted to present application to over 15 computational scientists, including the director of the research computing center at the University of Chicago

PROJECTS

Media Viewing Website [https://github.com/WoothAmwar/Complete-Personal-Website]

- Utilized the Google YouTube API to import user data from YouTube, created a custom homepage that caches the over 100 channels and 300 videos delivered to each user, and implemented a channel tag and watch later system.
- Built with the React library for the frontend, MongoDB and Python for backend, Vercel to host the application, and AWS Lambda and AWS API Gateway to create a personal **REST API**.

Hurricane Helene 3D Visualization [https://github.com/WoothAmwar/hurricane_vcc_unity_project]

- Designed an interactive 3D application to view hurricane datasets containing 200,000+ points across 6 days using Unity and C#, replicating NASA-inspired visualization methods in the **user interface design**
- Engineered automated data pipelines with Python (Pandas, Matplotlib, XArray) to process raw data files for visualization, **reducing dataset size by 99%** (5GB to 50MB)

MCP Server for Flight API [https://huggingface.co/spaces/WoothAmwar/mcp-sentiment/tree/main]

- Hosting **custom MCP Server** connected to the Amadeus Flights API to retrieve flight details between locations.
- Used Python and Gradio, hosted on HuggingFace, and compatible with my **custom MCP Client** and **Claude Desktop**

Schoolwide Game Tracking Platform [https://github.com/GoldEnter21/NCSSM-Spoons-Online]

- Developed a website to organize our school's student-led assassination game with **over 250 registered participants**, giving them the ability to track player eliminations, view user information, and view a live leaderboard.
- Utilized the React library for the frontend, MongoDB to store information, and Vercel to host the application.