Harsh Karia

hnkaria@ucdavis.edu | 669-336-2170 | [Personal Portfolio](#) | [LinkedIn](#) | [GitHub](#)

Education

University of California, Davis, B.S. in Computer Science — Sept 2022 – Dec 2025
- **Relevant Coursework:** Machine Learning, Deep Learning, Computer Networks(TCP/IP), Computer Vision, Algorithm Analysis, Data Structures, Operating Systems, Cybersecurity, UNIX Development

Skills

Technical: Python, C++, JavaScript, TypeScript, Machine Learning, Deep Learning, Backend Development, Large Language Models(LLMs), Systems, Linux, Network Protocols(TCP/IP), AWS, Cybersecurity, Penetration Testing

Experience

AI and Security Engineering Intern, Ethereum Foundation — June 2025 - Sept 2025
- Currently focused on code **security**, blockchain **infrastructure** and augmenting manual code reviews
- Creating custom **AST(abstract syntax tree) + RAG** system to represent syntactic structure of code and using LLM **fuzzy matching** with consensus layer specs to identify vulnerabilities in **Python and Rust**

LLM Researcher, Proper Data Group (Zubair Research Lab) — Jan 2025 – Present
- Conducting research into **memory of LLMs and AI agents** to identify potential lapses in data deletion, involuntary personalization, **security vulnerabilities**, and commercial memory controls

Software Engineering Intern, Corgi AI (YC) — Sep 2024 - Oct 2024
- Introduced automations for **financial data parsing** and email marketing using OpenAI's **LLM** models and **built backend APIs** for **financial data modeling** with **Python and React**, reducing employee requirements by **100x**
- Developed insurance policy pricing by building in-house risk management algorithms and **created databases** for property managers using **PostgreSQL and Python** for **backend**

Software Engineering Intern, [American Wild Horse Campaign](#) — June 2023 – Sept 2023
- Spearheaded end-to-end development of a **full-stack mobile application** (IOS and Android) using **React Native** and **JavaScript**, facilitating **data collection** and end to end **application security**
- Designed and deployed frontend user interfaces using **React** and integrated **object detection models** with **95% accuracy** to automate wild horse identification in images using Azure Cloud and **TypeScript backend**

Software Engineering Intern, SchedGo — Nov 2022 – Mar 2023
- Revamped meeting type display on SchedGo's web application to allow students to optimize schedules based on class times using **TypeScript** and **React.js (JavaScript)**
- Enhanced app functionality for over **2500 monthly users** across **4 universities** by refining design, **implementing database sharding** and improving website **speed by 20%**

Projects

AVIA (Stanford University Hackathon Winner) — [GitHub](#)
- Implemented **full-stack** AI command center that consolidates **real-time biometric data for 11 vital metrics** using Terra API and PPG recordings, **predicting blackouts** and pilot failures
- Created live strategy optimizations using **Python backend** with **React, JavaScript, and Tailwind CSS** frontend

SharkProof (UC Berkeley Hackathon) — [GitHub](#)
- Integrated Hume AI, Whisper, Google Gemini, and Groq **AI models** in the **backend** to **analyze 48 user emotions** and 36 facial expressions, **generating real-time** interview feedback
- Created real-time **web sockets** to route live interview data between the Hume AI and **React frontend**, improving **data flow efficiency by 25%** and **reducing feedback latency** for users

WordPlay (Hack MIT) — [GitHub](#)
- Developed interactive edtech game using **LLMs** and Godot, **integrating Google Gemini** to **enhance contextual understanding** and generate relevant AI game objects, improving content **accuracy by 40%**
- Designed a **Python based backend** level creation tool for educators to customize game levels and manage **AI-game interactions** with game physics, **reducing gameplay issues by 30%** and placing **top 12** at HackMIT

CourtCheck — [GitHub](#)
- Engineered **machine learning based tennis analysis** tool using **Python, PyTorch, OpenCV**, and **Detectron2** to achieve **95% accuracy** in court and ball detection, **reducing human error** in line calls by **90%**
- Utilized **NVIDIA T4 GPUs** to train Detectron2 ML models with **ResNet-50 backbone** and **Python backend**, processing **10,000 images** from tennis video film for keypoint detection with **React dashboard** for analytics