Vyshnav Kandamath

301-250-6977 vyshnavkandamath@gmail.com linkedin.com/in/vyshnavk1 github.com/vyshnavkandamath

Education

University of Maryland, College Park **Graduation: May 2025**

Bachelor of Science in Computer Science, Law and Society Minor ***(3.5 GPA)*** *College Park, Maryland*

Honors & Awards: President's Scholarship, University Honors, University Dean's List

Relevant Coursework: Data Structures, Object Oriented Programming I & II, Introduction to Computer Systems, Organization of Programming Languages, Algorithms, Discrete Structures, Introduction to Data Science, Programming Handheld Systems (Android), Web Application Development with JavaScript

Organizations: Maryland Minza Mixer and Registration Chair, Phunktions Hip Hop Dance Company, Alpha Kappa Psi Professional Business Fraternity

Technical Skills

Programming Languages: Java, Python, TypeScript, JavaScript, HTML5/CSS3, C, Ruby, Kotlin, Robot Framework

Frameworks: Node.js, Flask, Express, Git, GitHub, GitLab, Google Firebase, VMware, Android Studio, Unix/Linux, Visual Studio Code, Eclipse

Tools/Libraries/APIs: REST, Postman, Bootstrap, AJAX, SQLAlchemy, MongoDB

Work Experience

Hughes Network Systems **May 2024 – August 2024**

Software Engineering Intern *Gaithersburg, Maryland*

- Automated enterprise service delivery orders using **Robot Framework** and **Selenium**, significantly reducing manual entry time and increasing efficiency.
- Leveraged **Llama3, GPT-4 Large Language Models (LLMs)** and utilized **Python** and **LangChain** to develop an AI model that accurately determines and selects appropriate work order templates, utilizing **Retrieval-Augmented Generation (RAG)** model to optimize template selection.
- Deployed **ChromaDB** to manage and query data, streamlining efficient access to internal knowledge and increasing template retrieval efficiency.

Hughes Network Systems **May 2023 – August 2023**

Software Engineering Intern *Germantown, Maryland*

- Implemented API endpoints to enable create, read, update, and delete **(CRUD)** functions, decreasing manual data entry time for the team.
- Optimized configuration of **200+** satellite data sets by developing a **Node.js** and **TypeScript**-based parsing and auditing tool.
- Documented tool, adhering to **OpenAPI** specification for optimized internal reference.
- Conducted extensive **Postman** testing and debugging to ensure compatibility and reliability on Windows and Linux platforms.

Mindgrasp.ai **May 2022 – August 2022**

Software Engineering Intern *College Park, Maryland*

- Built free trial program that routes new users to payment pages using **Stripe API**.
- Programmed backend using **Flask** to authenticate users, distinguishing premium and free-trial users.
- Leveraged **AJAX** and **JavaScript** to make frontend API calls, displaying the appropriate webpages.
- Managed external database of **75,000+ users** using **Google Firebase** & **REST APIs**.
- Orchestrated seamless collaboration with an **agile-based** team of 5, leveraging their expertise to accelerate product development and achieve **10%** increase in monthly active users.

Projects

DC Metro Tracker Application | *Kotlin, Android Studio, Google Maps Platform* **December 2023**

- Developed a comprehensive DC Metro navigation application utilizing **Kotlin** and **Android Studio**, enhancing route planning and journey visualization.
- Seamlessly integrated **Google Maps Platform** to provide real-time tracking for **25+** metro routes, enhancing navigational accuracy and user trust.
- Designed and implemented an intuitive interface that successfully reduced route selection time, enabling users to quickly choose from **20+** stations as their departure and destination points.

VibeShift - Music Recommendation App | *Python, Flask, JavaScript, REST API* **July 2022**

- Built a music web application using Flask that takes in weather and temperature data from **WeatherAPI** and generates playlist recommendations through **Spotify API**.
- Developed **Python** backend API that authenticates Spotify users and receives playlist data. Used **JavaScript** and **AJAX** to receive playlist information from the backend.
- Created user interface using **HTML**, **CSS**, and **Bootstrap** to display correct playlist information.