Vishal Raavi

240-854-2384 — vishalraavi.work@gmail.com — linkedin.com/in/vishalraavi — github.com/vishal45-coder

Education

University of Maryland, College Park — **College Park, MD**
M.Eng, Cybersecurity (GPA: 3.86/4) — *Jan 2024 - Dec 2025(exp)*
Coursework: Security Tools, Hacking with C, Penetration Testing, Linux System Administration

GITAM Deemed University — **Bangalore, India**
B.Tech, Computer Science and Engineering (GPA: 3.2/4) / First Class with Distinction — *Jun 2019 - May 2023*
Coursework: Data Structures and Algorithms, FLAT, Databases, Web Development, Object Oriented Programming.

Skills

- **Programming Languages**: Python, Java, JavaScript, C, PHP, SQL, HTML, CSS.
- **Security Tools**: MetaSploit, Burp Suite, NMap, Wireshark, JMeter, SQLInjection, Shodan, Dig, Weevely, Sublist3r.
- **Cloud (AWS)**: EC2, S3, RDS (MySQL), VPC, Application Load Balancer (ALB), Auto Scaling, CloudFront, Route 53.
- **Cloud Security (AWS)**: WAF, IAM, KMS, AWS Certificate Manager (ACM), CloudWatch, CloudTrail.
- **Frameworks Libraries**: React, Node.JS, Flask, RESTful API, Pandas, Numpy, Tkinter.
- **Developer Tools**: Git, Github, Docker, Apache2, Azure (Basics).
- **Operating Systems**: Linux (Ubuntu, Kali), Windows, MacOS.

Work Experience

Center for Advanced Life Cycle Engineering (CALCE) — **College Park, MD**
Graduate Research Assistant / Full Stack Developer — *Aug 2024 - Present*
- Transitioned the desktop-based Reliability Tool to a web application using the **React** framework and the **React Flow** library, enhancing accessibility for **100+ users**.
- Learned **React from scratch** while building the application, delivering a **100% fully functional** solution with **0 reported bugs**.
- Engineered key features like upload layout, save layout, save data, and upload data, enhancing user experience and reducing task completion time by **45%**.
- Integrated the frontend with the **Flask** backend through **RESTful APIs** for the first time, facilitating data flow for 100+ users between the client and server.
- Ensured the application utilized stable packages and dependencies while constantly monitoring for **vulnerabilities**, resulting in **100% uptime** and **zero security incidents** during the first 3 months post-launch
- Resolved technical hurdles with research and AI tools like GPT and Claude, reducing development time by **20%**.

Center for Advanced Life Cycle Engineering (CALCE) — **College Park, MD**
Full Stack Developer — *May 2024 - Aug 2024*
- Built the Reliability Tool desktop version with **Python**'s **Tkinter**, supporting both Windows and macOS, reaching **100% cross-platform functionality**.
- Designed interactive elements like movable blocks and self-adjusting arrows, improving overall tool usability by **60%**.
- Developed core logic using **Python** to calculate system reliability and availability, benefiting **100% of users**.

Center for Advanced Life Cycle Engineering (CALCE) — **College Park, MD**
Web Developer — *Apr 2024 - May 2024*
- Crafted a responsive website with **HTML**, **CSS**, and **JavaScript**, featuring 5 key sections for improved user engagement.
- Integrated **Sheets DB via API** for seamless data storage, managing **50+ user submissions** related to feedback and issues.
- Optimized load times by **40%** and ensured **100% cross-browser compatibility**, enhancing user experience.

Projects

Scalable and Secure E-Commerce Platform on AWS
- Designed, developed, and deployed a robust e-commerce platform leveraging a suite of **AWS** cloud services for scalability, security, and performance.
- Architected for high availability using an **Application Load Balancer (ALB)**, **Auto Scaling Groups** with **EC2** instances, and a Multi-AZ **Amazon RDS (MySQL)** database.
- Implemented multi-layer security using **AWS WAF** to filter malicious traffic (e.g., **SQL injection**), **AWS Certificate Manager (ACM)** for **HTTPS** encryption, and **AWS KMS** for data-at-rest encryption.
- Optimized global content delivery and reduced latency by configuring **Amazon CloudFront (CDN)** to cache static assets stored in **Amazon S3**.
- Established comprehensive monitoring with **Amazon CloudWatch** for application health and performance metrics, and used **AWS CloudTrail** for API action auditing.