

DIGVIJAY TANAJI WAGHAMODE

Full Stack Developer · DevSecOps Engineer · AI Integration

Belgaum, Karnataka, India | +91 9353348198 | digvijaywaghamode2004@gmail.com

github.com/digvijay208 | linkedin.com/in/digvijay-waghamode-978260275 | portfoliodigvijayw.netlify.app

PROFESSIONAL SUMMARY

Computer Science Engineering student (Expected July 2027) with hands-on experience building production-grade full-stack web applications, DevSecOps automation pipelines, and AI-powered systems. Designed and shipped SurfaceIQ — a live web security scanner deployed on Vercel — and InfraGuard, a GitHub Actions CI/CD pipeline that auto-blocks vulnerable infrastructure deployments. Proficient in TypeScript, Node.js, MongoDB, PostgreSQL, Docker, Kubernetes, GitHub Actions, and Playwright. Seeking a Full Stack Developer or DevSecOps Intern/Trainee role in Maharashtra or remote.

TECHNICAL SKILLS

Languages: JavaScript, TypeScript, Python, C, C++, Java, SQL, Bash

Frontend: React.js, Next.js, HTML5, CSS3, Tailwind CSS, Vue.js, Figma

Backend: Node.js, Express.js, REST APIs, JWT Authentication

Databases: MongoDB, PostgreSQL, MySQL

DevSecOps & Cloud: GitHub Actions, CI/CD Pipelines, Docker, Kubernetes, Terraform (IaC scanning), Linux, Jenkins, AWS (basics)

Security & Testing: Vulnerability scanning, OWASP principles, Playwright (E2E), Postman, Vitest

Platforms & Tools: Vercel, Git, VS Code, Netlify, Vercel Blob, Next.js App Router

Concepts: DevSecOps, Infrastructure Security, Agile/Scrum, OOP, Data Structures & Algorithms

PROJECTS

SurfaceIQ — Authorized Web Security Scanner | [github/SurfaceIQ](https://github.com/SurfaceIQ) *TypeScript · Next.js · Playwright ·*

PostgreSQL · Vercel

Live: surfaceiq.vercel.app

- Built a **full-stack authorized website security scanner** using Next.js (App Router), Playwright for live browser automation, and PostgreSQL for user-isolated scan data storage.
- Implemented three scanning modes — public crawl, credential-based login, and session cookie import — with all credentials AES-encrypted before persistence using a server-side secret.
- Architected a monorepo with separate Next.js web app and background worker service; auto-selects local file storage (dev) vs. hosted Postgres + Vercel Blob (production) for zero-config deployment.
- Deployed on Vercel with production-ready storage adapters; generates structured HTML reports with findings, evidence screenshots, and remediation recommendations.
- Wrote a Vitest test suite covering scan rules and core logic; structured codebase with packages/core for shared types, rules, and storage interfaces.

InfraGuard — DevSecOps Vulnerability Scanning Pipeline | <https://github.com/digvijay208> *GitHub Actions*

· YAML · Bash · Docker · Kubernetes · Terraform

- Designed and implemented an **automated DevSecOps pipeline** that scans infrastructure-as-code files (Dockerfile, .yaml, .tf) for security vulnerabilities before any deployment.
- Built a custom risk scoring engine that calculates a severity score per git push; deployments scoring above 60 or containing any critical vulnerability are automatically blocked.
- Integrated GitHub Actions as the CI/CD trigger so every commit automatically invokes the InfraGuard CLI scanner without manual intervention — zero-touch security gate.
- Pipeline outputs a structured JSON vulnerability report on every run providing a full audit trail for both blocked and approved deployments.

FusionAI — Multi-Model AI Collaboration System | [github/multi_agent_ai](#) TypeScript · JavaScript · CSS · GPT API · Gemma API · DALL-E

- Built a system where **multiple AI models** (GPT-oss and Gemma) analyze queries independently and synthesize their insights to deliver richer, cross-validated answers.
- Integrated DALL-E image generation API enabling AI-generated visual outputs within multi-turn collaborative sessions; persisted conversation history across sessions.
- Implemented full conversation history management to maintain context across multi-turn interactions with three independent model APIs simultaneously.

AI Prompt Generator — Full Stack Web Application | [github/ai-prompt-generator](#) TypeScript · Node.js · Express.js · MongoDB · HTML/CSS

- Built a **full-stack TypeScript web application** that generates optimized prompts for AI tools including ChatGPT, Claude, and other LLMs — with persistent storage and responsive UI.
- Implemented backend REST API with Node.js/Express and stored prompt data in MongoDB; designed iterative prompt refinement workflow based on user testing sessions.
- Deployed frontend on Netlify; backend API follows REST conventions with proper route handling, input validation, and JSON response structure.

EDUCATION

Bachelor of Engineering — Computer Science & Engineering — Hirasugar Institute of Technology, Nidasoshi *Expected: July 2027*

Relevant Coursework: Data Structures & Algorithms, Operating Systems, Database Systems, Computer Networks, Software Engineering

SELF-DIRECTED ENGINEERING EXPERIENCE

Full Stack & DevSecOps Engineering Projects — Independent / Academic *2023 – Present*

- Shipped 4 production-grade projects spanning web security scanning, DevSecOps automation, multi-agent AI systems, and full-stack TypeScript applications.
- Applied DevSecOps principles by building InfraGuard — an automated CI/CD security pipeline using GitHub Actions for infrastructure file scanning with zero manual intervention.
- Deployed SurfaceIQ on Vercel with a Next.js + PostgreSQL + Playwright stack; implemented production/local environment switching, Vercel Blob artifact storage, and a monorepo architecture.
- Practiced multi-model AI integration, REST API design, JWT authentication patterns, and end-to-end TypeScript development across multiple shipped projects.
- Debugged and optimized applications across JavaScript, TypeScript, Python, and Bash; applied Agile-style iteration with feature branches and pull requests on GitHub.

CERTIFICATIONS & TRAINING

- UNIX & Linux OS Fundamentals — Certification Completed
- UI/UX Design Certification
- Claude Code in Action — AI-assisted development workflows

LINKS & PORTFOLIO

Portfolio: [portfoliodigvijayw.netlify.app](#) | **SurfaceIQ Live Demo:** [surfaceiq.vercel.app](#)