

VISHNU VIVEK VALETI

+1 (352) 226-7455 | vishnuvivek.valeti@gmail.com | [linkedin.com/in/valetivishnuvivek](https://www.linkedin.com/in/valetivishnuvivek) | github.com/valetivivek

EDUCATION

M.S. in Computer Science

University of Florida

Aug 2024 – May 2026

GPA: 3.83

- **Coursework:** Advanced Data Structures, Distributed Operating Systems, Internet Storage Systems, Software Engineering, Distributed Multimedia Systems, Bioinformatics, Natural Language Processing.

B.Tech in Information Technology

Vignan University

Aug 2020 – May 2024

GPA: 3.43

- **Coursework:** Data Structures and Algorithms, Java, Python, Web Applications, DBMS, Computer Organization and Architecture, IoT, Network and Cyber Security, Operating Systems.

TECHNICAL SKILLS

Programming Languages: Python, Go, C/C++, JavaScript, TypeScript, SQL, HTML, CSS.

Frameworks & Libraries: React.js, Node.js, Express.js, PyTorch, TensorFlow, Streamlit.

Databases: PostgreSQL, MySQL, MongoDB, Redis, Supabase.

Cloud & DevOps: AWS, Docker, CI/CD, GitHub Actions, Linux/Unix.

Testing & Tools: Unit Testing, Cypress, Jest, Git, REST APIs, Agile/Scrum, Go test.

EXPERIENCE

Machine Learning Engineer Intern – ReplyQuick LLC

Dec 2025 – Present

- Trained and fine-tuned YOLOv8-based computer vision models on dental imaging datasets for object detection tasks.
- Built Python-based data preprocessing, augmentation, and annotation pipelines for large-scale dental image training.
- Developed reproducible cloud-based training pipelines on AWS to support controlled experimentation and evaluation.
- Collaborated with software engineers to integrate ML inference outputs into production application workflows.

Graduate Student Assistant – University of Florida

Aug 2025 – Dec 2025

- Designed and developed a full-stack web application using React and backend APIs to automate accreditation workflows.
- Implemented backend business logic and decision-analysis rules to support consistent tiered accreditation outcomes.
- Built dynamic, data-driven UI components and reporting features that reduced manual review effort by 50 percent.
- Deployed and maintained the application on Vercel, ensuring reliability, performance, and long-term maintainability.

Software Development Program Apprentice – Vignan University

Jan 2024 – May 2024

- Built and maintained a full-stack academic portal using React, PHP, and MySQL to streamline academic workflows end-to-end.
- Automated record management, validation, and update logic to eliminate redundant manual processes across systems.
- Designed secure, role-based dashboards with authorization controls for academic and administrative users.
- Implemented backend integrations and optimized database queries to ensure transactional consistency and system reliability.

ACADEMIC PROJECTS

JobScoop

Stack: Go, PostgreSQL, React, Redis, Docker, Cypress, Jest

- Built a responsive React frontend with protected routes, authentication flows, and subscription management dashboards.
- Implemented job filtering, date-based search, and company-role trend visualizations using Chart.js analytics components.
- Integrated REST APIs with structured loading states, granular error handling, and retry logic to improve reliability.
- Developed reusable UI components and context-based state management patterns to support scalable frontend architecture.

Reddit-Style Discussion Platform

Stack: Go, PostgreSQL, REST APIs

- Built a concurrency-safe backend supporting users, subreddits, posts, nested comments, voting, and messaging features.
- Implemented synchronization primitives using mutexes, atomic counters, and sync.Map for thread-safe state management.
- Designed RESTful APIs for feeds, voting, messaging, and karma computation across end-to-end application workflows.
- Developed a Go-based CLI client to simulate users, replay workflows, and validate behavior under concurrent load.

Rule-Based Chatbot

Stack: React, TypeScript, Node.js, Express, WebSockets, Redis, Docker

- Built a full-stack rule-driven chatbot using React and Node.js with WebSocket-based real-time messaging support.
- Implemented hot-reloadable rule evaluation using JSON schemas, checksum validation, and in-memory context storage.
- Designed REST APIs and WebSocket protocols to support rule validation, simulation runs, and real-time chat workflows.
- Containerized frontend and backend services using Docker and Docker Compose for local multi-service environment setup.