

SHREELAXMI HEGDE

shreelaxmihegde9@gmail.com | [GitHub](#) | [LinkedIn](#)

TECHNICAL SKILLS

- ❖ **Languages:** JavaScript, TypeScript, Python
- ❖ **Frontend:** React.js, Next.js, Tailwind CSS, MUI
- ❖ **Backend:** Node.js, Express.js, FastAPI, REST APIs, JWT
- ❖ **Database:** MongoDB, PostgreSQL, Prisma ORM, Mongoose
- ❖ **Devops & Tools:** Docker, Docker Compose, GitHub Actions (CI/CD), AWS (EC2, S3)
- ❖ **Testing:** Jest, Vitest
- ❖ **AI:** Agno, Gemini API, Pydantic

PROJECTS

- ❖ **ConstraintEngine** – AI Software Architecture Platform (May2026 - July2026)
[Gemini 2.5 Flash, Agno, FastAPI, Node.js, PostgreSQL] [Live Demo](#) | [GitHub](#)

Built an AI Agent system that extracts software requirements into structured architectural constraints and enables iterative architecture discussions.

- Implemented structured LLM outputs using Pydantic schemas and validation to ensure consistent agent responses.
- Built contextual conversation management with architecture versioning and state tracking across discussions.
- Engineered product-aware rate limiting and multi-stage input validation combining regex filtering, semantic validation and LLM classification to reduce unnecessary AI requests.
- Implemented JWT authentication, project versioning and PostgreSQL persistence for conversation history and architectural state.
- Designed multi-service architecture integrating Node.js backend with FastAPI-based AI services through RESTful APIs.

- ❖ **Accommate** – E-Commerce Platform (Aug2025 - March2026)
[MERN, Docker, GitHub Actions, AWS EC2] [Live Demo](#) | [GitHub](#) | [DockerHub](#)

Built an end-to-end production-style e-commerce application using MERN stack with a modular full-stack architecture.

- Designed REST APIs and MongoDB data models following MVC architecture to improve maintainability and separation of concerns.
- Implemented secure session-based authentication and authorization using Passport.js, along with request validation using Joi to protect backend endpoints.
- Engineered centralized error-handling pipeline across both frontend & backend, providing consistent API responses and simplified debugging.
- Integrated secure image upload workflows using Multer and Cloudinary with server-side validation for invalid file handling.
- Containerized the application using multi-stage Docker builds and Docker Compose for consistent development and production environments, and published images to Docker Hub.
- Built automated CI/CD pipeline with GitHub Actions to run Jest unit tests, Vitest component tests, build Docker images, and deploy the application to AWS EC2.

❖ Siya-cli – Version Control CLI tool

(Jan2026 – Feb2026)

[Node.js, yargs.js, AWS S3]

[GitHub](#)

- Developed a Git-inspired Version control system supporting repository initialization, staging, commits, push, pull and revert operations.
- Implemented **argument parsing** and structured **command handling** execution pipeline.
- Integrated AWS S3 for remote repository storage and synchronization.

OPEN SOURCE CONTRIBUTIONS

- **stdlib.js**: Contributed to lint fixes and documentation improvements following function migrations. ([PR Link](#))
- **AWS SDK for JavaScript V3**: Submitted documentation improvements accepted into the project backlog. ([Issue Link](#))
- **Prisma ORM web**: Reported reproducible web UI issue with detailed investigation. ([Issue Link](#))
- **Hacktoberfest 2025**: Contributed to web UI functionality issues across 2 open source projects. ([PR Link1](#)) ([PR Link2](#))