

HIMANSHU BHINDORIYA

FULL-STACK DEVELOPER | REACT, NEXT.JS, NODE.JS

Bhopal, Madhya Pradesh | +91-9713787369 | work.himanshu.hb@gmail.com
frontifybyhb.xyz | github.com/FrontifybyHB | npmjs.com/~frontifybyhb
linkedin.com/in/himanshu-bhindoriya-077812290

SUMMARY

Full-stack developer with production internship experience building React interfaces and REST APIs using Next.js, Node.js, Express.js, and MongoDB. Developed a reusable backend scaffold with JWT authentication, Redis caching, and BullMQ email jobs. Built AI-integrated web applications and published an open-source npm package.

WORK EXPERIENCE

Full-Stack Web Developer Intern

Feb 2026 – Apr 2026

Sheryians Coding School

hire.sheryians.com

- Built 10+ reusable React components and REST APIs for a quiz platform serving 1,000+ learners.
- Developed and shipped production features across Next.js, Node.js, Express.js, and MongoDB.
- Reviewed pull requests, resolved bugs, and collaborated with developers to deliver production updates.

PROJECTS

Backend Starter | GitHub | Live Demo

Dec 2025 – Present

Node.js, Express.js, MongoDB, Redis, BullMQ, Jest

- Built a reusable backend scaffold with layered controllers, services, repositories, JWT authentication, role-based access control, and centralized error handling.
- Implemented Redis caching with an in-memory fallback and BullMQ email jobs with retries and exponential backoff.
- Added Joi request validation, rate limiting, Winston logging, and a Jest/Supertest testing setup.

WebDev News | GitHub | Live Demo

Mar 2026

Next.js, TypeScript, TanStack Query, Tailwind CSS, Groq API

- Built a developer-news dashboard with category filters, article modals, loading skeletons, and load-more controls.
- Integrated Groq through server-side Next.js code, keeping API credentials on the server and validating generated news data.
- Used TanStack Query to cache API responses, reuse fetched pages, and support refresh and error handling.

NASA Knowledge Engine | GitHub | Live Demo

Sep 2025

React.js, Node.js, Express.js, OpenAI API, RAG

- Built a question-answering application using retrieval-augmented generation (RAG) over 600+ pages of NASA research documentation.
- Used vector embeddings to retrieve relevant document content and the OpenAI API to generate context-aware answers.

TECHNICAL SKILLS

Languages: JavaScript (ES6+), TypeScript, HTML5, CSS3

Frontend: React.js, Next.js, Tailwind CSS, Redux Toolkit, Zustand, TanStack Query

Backend: Node.js, Express.js, REST APIs, JWT, RBAC, OAuth 2.0, Socket.IO, Joi

Databases & Queues: MongoDB, Mongoose, PostgreSQL, Redis, BullMQ

AI Integration: OpenAI API, Groq API, LangChain, Retrieval-Augmented Generation (RAG)

Testing & Tools: Jest, Supertest, ESLint, Git, GitHub Actions, Postman, Vercel, Render

ACHIEVEMENTS

Top 5 of 105 Teams | Samadhan 2.0 National Hackathon, SISTEC Group

2025

Published npm Package | secure-test-guard

Feb 2026

1st Prize | Inter-College Website Development Competition

2025

EDUCATION

SAM Co-Ed College, Bhopal

2024 – 2027

Bachelor of Computer Applications (BCA), expected 2027