

Muhammad Bilal Rafique

Full-Stack Developer · Next.js & TypeScript

bilalrafique271@gmail.com | bilaldev.tech | github.com/Muhammad-Bilal-Rafique | linkedin.com/in/muhammad-bilal-rafique

SUMMARY

Self-taught Full-Stack Developer and 4th-semester BSCS student at UVAS, specialising in Next.js and TypeScript. Experienced in building production-grade web applications with secure payment flows, role-based access control, and AI integrations. Available for Summer Internship 2026.

TECHNICAL SKILLS

Frontend: Next.js 15, React, TypeScript, JavaScript, Tailwind CSS, Zustand, Framer Motion, Shadcn UI, HTML/CSS

Backend & Databases: Node.js, Express.js, MongoDB, REST APIs, Server Actions

Tools & Infrastructure: Git & GitHub, Vercel, Stripe, Cloudinary, Auth.js, Resend

Currently Exploring: AI SDKs, LLM APIs, Prompt Engineering, Performance Optimisation

PROJECTS

Orbit V2 — Full-Stack E-Commerce Platform

Next.js · TypeScript · Tailwind CSS · Stripe · Zustand · Auth.js · MongoDB · Cloudinary · Resend · AI SDK | orbit-three-iota.vercel.app

- Architected a type-safe, production-grade e-commerce platform from scratch; complete rewrite of Nexus V1 to resolve scalability and maintainability issues.
- Implemented Stripe Checkout with webhook-driven order confirmation and automated transactional emails via Resend.
- Built a full RBAC system with Auth.js — hashed passwords, Google OAuth, email verification, and protected admin routes via middleware.
- Developed an admin dashboard with live inventory tracking, order status management, revenue charts, top-product analytics, and date-filtered CSV export.
- Integrated LLM APIs to auto-summarise customer reviews, reducing reading load and surfacing product insights.
- Shipped product variants (size/colour/storage), wishlist, real-time order tracking timeline, verified-purchase review system, and low-stock automated email alerts.

Nexus — MERN Stack E-Commerce (V1 Prototype)

React · Node.js · Express.js · MongoDB · JavaScript

- Built a full-stack e-commerce prototype from scratch in vanilla JavaScript, without component libraries, to deeply understand frontend fundamentals.
- Encountered real-world limitations — prop-drilling, manual state, lack of type safety — that directly motivated the architectural decisions made in Orbit V2.
- Served as the proof-of-concept foundation that justified migrating to Next.js, TypeScript, and centralised state management.

EDUCATION

Bachelor of Science in Computer Science (BSCS)

University of Veterinary and Animal Sciences (UVAS) | Expected Graduation: August 2028 | GPA: 3.2 / 4.0

Relevant Coursework: Data Structures & Algorithms, Web Development, Object-Oriented Programming

Open to Summer Internship 2026 · Portfolio: bilaldev.tech · References available on request