

# BIBEK KUMAR THAGUNNA

+91-7983183689

bibek@velayon.com

bibek.velayon.com

github.com/Bibek-Kumar-Thagunna

## Professional Summary

AI-Assisted Systems Engineer and Founder specializing in rapid prototyping and production deployment. Built 5 full-scale SaaS platforms (web, mobile, desktop) in 1 year using AI-accelerated workflows—all deployed and functional, available for testing at velayon.com. No active users or customers yet, but systems prove technical capability. Expert in system architecture, real-time data processing, and shipping MVPs in weeks. Seeking remote roles at AI-forward startups where velocity and pragmatism outweigh manual coding purity.

## Experience

### Founder & Lead Systems Engineer

Velayon Dynamics

Jan 2024 – Present

Remote (Kathmandu, Nepal)

- Architected and deployed 5 production-grade systems using AI-assisted development (Cursor, Replit Agent, Google Antigravity), reducing time-to-market by 10x compared to traditional development
- Built **Attendify** (SaaS): QR + GPS attendance system with real-time analytics, anti-proxy detection, and Flutter mobile app. Fully deployed at attendify.velayon.com—no active users yet, but available for live testing
- Engineered **HMS Core**: Multi-platform hotel management system (Next.js web, Flutter mobile/desktop) with Nepal tax compliance, billing, room management, and kitchen workflows. Complete system deployed across 5 platforms—ready for pilot customers
- Developed **Face Recognition Dashboard**: Real-time CCTV-integrated system processing 30fps video streams with <100ms latency using OpenCV and ML pipelines. Proof-of-concept deployed and functional
- Created **Expense Tracker**: Automated financial tracking via SMS parsing with 95% accuracy using Hive (offline-first) and ML-based transaction categorization. Working demo available on GitHub

## Core Competencies

**System Architecture:** Multi-tier design, API architecture, database modeling, real-time systems, microservices

**AI-Assisted Engineering:** Cursor IDE, Replit Agent, Google Antigravity, GitHub Copilot (10x velocity workflows)

**Technologies:** Next.js, React, Flutter, Node.js, FastAPI, Supabase, Firebase, Prisma, PostgreSQL, Docker

**Platforms:** Web (Vercel, Netlify), Mobile (iOS/Android), Desktop (Windows/macOS/Linux), Cloud (GCP, AWS)

**Specialized Skills:** Real-time data processing, Face recognition (OpenCV), SMS/NLP parsing, Location-based systems

## Key Projects

### Attendify | Flutter, Firebase, Node.js, QR/GPS Integration

2025 – 2026

- Smart attendance tracking with anti-proxy system: QR codes + GPS fencing to prevent location spoofing
- Real-time analytics dashboard with session management, late detection, and teacher/admin role separation
- Deployed and functional at [attendify.velayon.com](https://attendify.velayon.com)—no active users yet, but available for live testing and demos

### Hotel Management System (HMS Core) | Next.js, Flutter, Prisma, Multi-platform

2025 – 2026

- Full-featured hotel operations platform: billing, room management, restaurant/kitchen workflows, Nepal tax compliance
- Cross-platform deployment: Web, iOS, Android, Windows, macOS, Linux (single codebase for mobile/desktop)
- Scalable for 5-star hotels and local lodges with modular architecture supporting custom feature sets

### Face Recognition Dashboard | Python, OpenCV, FastAPI, Real-time Processing

2025 – 2026

- CCTV-integrated attendance system processing 30fps streams with multi-face detection and confidence scoring
- Built logging system with FPS monitoring and real-time dashboard updates for security and attendance use cases

## Education

### Thapar Institute of Engineering and Technology

2023 – 2027 (Expected)

Bachelor of Engineering in Computer Science

Patiala, India

## Additional Information

**Availability:** Open for remote full-time roles and contract work. Flexible schedule with 4-6 hour overlap with US time zones (EST/PST)

**Rate Expectations:** \$15-25/hour (contract), \$32k/year (full-time) – negotiable based on project scope

**Engineering Philosophy:** Ship first, optimize later. Prioritize working software over perfect code. Leverage AI to build 10x faster without sacrificing production quality