KRISHNAM BIYANI

Kolkata, IND | P: +91 8696612974 | krishnambiyani5@gmail.com | LinkedIn | GitHub | Portfolio

EDUCATION

VELLORE INSTITUTE OF TECHNOLOGY Vellore, Tamil Nadu
Bachelors in Technology, Major in Computer Science
Cumulative GPA: 8.83
Relevant Coursework: Data Structures and Algorithms, Web Development

TECHNICAL SKILLS

- **Languages:** C++, JavaScript
- **Backend:** Node.js, Express.js, MongoDB, Socket.IO, JWT
- **Frontend:** React.js, Zustand, Tailwind CSS, Bootstrap, HTML5, CSS3, Axios
- **Platforms & Infrastructure:** Docker, Docker Compose, Redis, RabbitMQ, NGINX, Git
- **Services & APIs:** Vercel, Render, Cloudinary, FFmpeg, MediaRecorder API

WORK EXPERIENCE

Influcon Digitals Kolkata, Remote
Frontend developer Intern May 2025 - June 2025

- Owned the frontend development for the 8-section official website, translating complex Figma mockups into pixel-perfect, mobile-first components using React and Tailwind CSS.
- Implemented a premium, "sticky stacking" scroll experience with a full-viewport video header, using React Hooks to dynamically calculate section heights and implement a custom smooth-scroll navigation..

TimeSlotter Vellore, Remote
Full-stack developer Intern May 2024 - June 2024

- Designed and developed RESTful APIs with Node.js and Express to manage user profiles, bookings, and time slot data.
- Implemented a secure authentication and authorization system using JSON Web Tokens (JWT) to protect user-specific routes and data.
- Collaborated with senior engineers to analyze and refactor legacy MongoDB schemas, optimizing data models to improve query performance and ensure platform-wide data integrity.

PROJECTS

SOCIAL MEDIA BACKEND - A Microservices Architecture Project link
Tech Stack: Node.js, Express.js, MongoDB, RabbitMQ, Redis, Docker, Docker Compose, JWT, Cloudinary

- Architected a distributed backend with 4 distinct microservices (Identity, Post, Media, Search) fronted by a central API Gateway that enforces JWT authentication and request proxying..
- Engineered an event-driven system using RabbitMQ with a 'topic' exchange to asynchronously propagate data; for example, a post.deleted event triggers media cleanup in the Media service and data removal in the Search service.
- Implemented multiple caching and performance patterns with Redis, including response caching for paginated API calls and global rate-limiting at the API Gateway.

STREAMHUB – Browser-based Livestream to YouTube & HLS Project link
Tech Stack: React, Node.js, Express, Socket.IO, FFmpeg, Docker, NGINX RTMP, MediaRecorder

- Captured browser audio/video using the MediaRecorder API, chunking the video/webm stream into 100ms fragments and transmitting them to a Node.js backend via Socket.IO (binarystream event).
- Engineered a real-time transcoding pipeline by piping the binary Socket.IO chunks directly to the stdin of a spawned FFmpeg child process, converting the stream to flv format on the fly.
- Pushed the transcoded flv stream to dual destinations: YouTube's live ingest (RTMP) and a Dockerized NGINX-RTMP server configured to automatically segment the stream into an HLS playlist for web playback.

DSA DUEL – Real-time Competitive Coding Platform Project link
Tech Stack: MERN, Socket.IO, Judge0 API, Zustand, Tailwind CSS, JWT, Render

- Engineered a real-time duel system using Socket.IO; the first of two users to join a room triggers a broadcast of a random DSA problem and a server-authoritative start time to synchronize a 10-minute countdown timer for both clients.
- Integrated the Judge0 API for code execution; on submission, the backend validates the code against all defined test cases and, upon a 100% pass, emits a Socket.IO event to broadcast the winner.
- Managed all frontend state by creating 3 distinct Zustand stores to modularize concerns for authentication, real-time room data, and code execution state.