

Aryan Singh Bisht

9695136939 | aryansinghbisht19@gmail.com | [LinkedIn](#) | [Github](#)

SUMMARY

Results-driven backend developer with proven expertise in Node.js, Express, and AWS, as well as decentralized application development on Solana blockchain using React and Solana Web3.js. Successfully built scalable solutions like a GenAI system at Samsung PRISM, achieving a 48.92% boost in DSL query generation accuracy, and developed user-friendly Solana Token Launchpad and Token Generator apps enabling seamless creation and minting of SPL tokens with secure wallet integration. Experienced in full-stack development with SQL databases and Java-based enterprise solutions. Passionate about delivering high-performance tools that reduce operational costs and accelerate innovation for forward-thinking organizations.

PROFESSIONAL SKILLS AND EXPERTISE

Languages & Technologies: Java, SQL, JavaScript, TypeScript, Rust (basic for Solana), Solana Web3.js

Frameworks & Libraries: Next.js, Node.js, React.js, Solana Wallet Adapter, PostgreSQL

Database and Misc: MySQL, MongoDB, Git, REST API, POSTGRES

Architectures and Paradigms: Monolithic, Microservice, Event Driven Architecture, Object Oriented Programming, Decentralized Applications (DApps)

Miscellaneous: AI, ML, Blockchain Development

Certificates:

- *AWS Certified Solutions Architect – Associate* - [Certificate Link](#)
- *AWS Cloud Practitioner* - [Certificate Link](#)
- *Microsoft: Azure AI Fundamentals* - [Certificate Link](#)
- *Microsoft Certified: Azure AI Engineer Associate* - [Certificate Link](#)
- *DSA using Java* - [Certificate Link](#)

EDUCATION

Vellore Institute of Technology (Vellore)
Bachelor of Technology in Information Technology

Sept. 2022 – Dec 2026

City Montessori School (X)
Indian School Certificate

92.7%
June. 2018 – April 2019

City Montessori School (XII)
Indian School Certificate

91.7%
June. 2020 – April 2021

WORK EXPERIENCE

Samsung PRISM Research Intern
Project Repository — [Certificate Link](#)

VIT Vellore, Sep 2024 – May 2025

- Increased PromQL query generation **validity rate to 83.33%** by implementing a modular RAG framework with hybrid dense-sparse retrieval and syntax correction, resulting in a **48.92% improvement over baseline ChatGPT accuracy**.
- Decreased query generation **ambiguity and failure rates for complex user inputs** by introducing agentic RAG flows leveraging dynamic tool orchestration in LangChain, improving the system's ability to handle ambiguous or multi-step natural language descriptions.

- Enhanced system **scalability and relevance** by curating and integrating datasets (Prometheus documentation, Kaggle NL–PromQL pairs, and custom metrics), ensuring coverage from simple to complex intents, and systematically addressing data context challenges during pre-processing.

PROJECTS

- Zentry Clone** | *React.js, Tailwind CSS, Vite, GSAP, Framer Motion* | [Live Demo](#) [Project Link](#)
- Built a modern and interactive portfolio website called **Zentry Clone** using **React.js** and **Tailwind CSS**, inspired by an Awwwards-winning design.
 - Implemented smooth page transitions and scroll-based animations using **GSAP** and **Framer Motion** to enhance user experience.
 - Optimized performance and development workflow with **Vite**, and followed best practices to ensure clean, responsive, and scalable code.
- Smart Agriculture** | *Node Js, Javascript, Python, AWS, MongoDB, Nodejs*, [Project Link](#)
- Developed a **Real Estate Developer** lead generation platform with a user-first approach, utilizing **AWS services** for seamless functionality.
 - Implemented **AWS Lambda** with **API Gateway** to process user messages and store images securely in **Amazon S3**, integrating **AWS Rekognition** for intelligent image analysis.
 - Built backend services with **Node.js** and enhanced client-side interactions using **JavaScript** for a smooth and responsive user experience.
- E-commerce Website** | *React, Node.js, Express, MongoDB, EJS, SQL, Stripe, Razorpay* [Project Link](#)
- Developed a mobile-first property booking platform with **dynamic search functionality** based on location, price, and amenities, using **MongoDB** for efficient data management.
 - Built the backend with **Node.js** and **Express**, integrating user authentication, property management, and transactional data handling with a **SQL database** for bookings and payments.
 - Enhanced performance and user experience through **caching**, **pagination**, and **real-time updates** using **EJS**, while following *clean code architecture* and *Google best practices for Node.js*.
 - Designed and deployed a full-stack e-commerce website using **React**, **Node.js**, and **MongoDB**, featuring secure payments via **Stripe** and **Razorpay**, and robust order management workflows.
- Solana Token Generator** | *React, Solana Web3.js, SPL Token, Wallet Adapter* | [Live Demo](#) [Project Link](#)
- Built a decentralized application to create and mint custom SPL tokens on the Solana blockchain with a user-friendly interface.
 - Integrated wallet connection using **Solana Wallet Adapter** supporting Phantom and Solflare wallets for secure transaction signing.
 - Implemented token minting logic including mint account creation, associated token account setup, and token minting with real-time transaction status and error handling.
 - Provided cost breakdown and wallet balance checks to ensure users have sufficient SOL to cover token creation and transaction fees on Devnet.
 - Designed with React for a smooth UX and detailed feedback on transaction progress and results.
- Solana Token Launchpad** | *React, Solana Web3.js, SPL Token, Wallet Adapter* | [Live Demo](#) [Project Link](#)
- Built a decentralized application to create and mint custom SPL tokens on the Solana blockchain with a user-friendly interface.
 - Integrated wallet connection using **Solana Wallet Adapter** supporting Phantom and Solflare wallets for secure transaction signing.
 - Implemented token minting logic including mint account creation, associated token account setup, and token minting with real-time transaction status and error handling.
 - Provided cost breakdown and wallet balance checks to ensure users have sufficient SOL to cover token creation and transaction fees on Devnet.
 - Designed with React for a smooth UX and detailed feedback on transaction progress and results.