Sudip Sarkar

☎ +91-62906-84081 ✉ official.sudipsarkar2003@gmail.com 🌐 sudip-sarkar.vercel.app
github.com/SudipSarkar1193 linkedin.com/in/sudip-sarkar

Education

Academy of Technology **2022 – Present**
Bachelor of Technology in Computer Science and Engineering *CGPA: 8.36/10*

Chandannagar R.D.P. Institute **2021**
Higher Secondary Education (Class XII) *Percentage: 76.50%*

Chandannagar R.D.P. Institute **2019**
Secondary Education (Class X) *Percentage: 85.28%*

Technical Skills

Languages: Java, C, C++, Golang, JavaScript, Python, SQL, HTML/CSS
Frameworks & Libraries: React.js, Node.js, Express.js, LangChain
Databases: MongoDB, PostgreSQL, Firebase
Operating Systems: Linux

Projects

Movie Ticket Booking System | *Java, Maven, OOP* **Github**

- Developed a console-based movie ticket booking system like BookMyShow using Java, enabling users to select theatres, shows, and seats with a layered architecture (models, services, controllers).
- Implemented core models (Movie, Theatre, Show, Booking) and enums (City, SeatCategory) to represent real-world entities, with services for managing theatres, shows, and bookings.
- Designed a BookingController to orchestrate user interactions, integrating TheatreService, ShowService, and BookingService for seamless theatre selection and ticket booking.
- Utilized Maven for dependency management and streamlined compilation, ensuring a robust build process with automated dependency resolution and execution.

Github
Parallel Merge Sort Implementation | *Java, Multi-Threading, OOP*

- Developed a console-based Java application to compare parallel and sequential Merge Sort algorithms, processing a large-sized random array with performance measurement.
- Designed a modular architecture with separate packages (sorters, utils, app), implementing a Sorter interface, ParallelMergeSort, SequentialMergeSort, and utility classes for array generation and merging.
- Utilized ExecutorService for multi-threading, parallelizing left and right subarray sorting with a threshold-based approach to optimize performance for large datasets.
- Implemented a user-driven demonstration mode to compare execution times of parallel and sequential algorithms, showcasing thread overhead effects on smaller arrays.

Live Demo
AI-powered Quiz application | *Golang, Python, Javascript, React, Postgresql*

- Developed a full-stack interactive quiz platform allowing users to create and share quizzes based on custom topics. Included features such as user authentication, profile customization, and real-time participation.
- Implemented user authentication using Google and email OTP verification for secure login.
- Leveraged AI technology to generate personalized quizzes based on user input.
- Enabled quiz participation, result tracking, and history saving to a database. Provided users with the ability to share quizzes via unique links.

Live Demo
Social Media application | *MERN Stack*

- Built a full-stack social media application enabling user registration, authentication, and email verification through secure verification links.
- Implemented core features including creating text and image posts, commenting on posts, and saving posts for later.
- Implemented an in-app notification system to inform users about post interactions and updates in real-time.
- Designed an intuitive user interface for smooth user experience and content interaction.
- Integrated backend APIs for handling user data and post interactions efficiently.