Siddhant Thakare

+1 (951)-544-8606 siddhant.thakare@outlook.com
siddhantthakare.github.io in/siddhant-thakare iambyt3z

Education

University of California, Riverside **Sep 2023 – Dec 2024**
M.S. in Computer Science *GPA: 3.93/4*
Coursework: Advanced Algorithms, High-Performance Computing, Big-Data Management

BITS Pilani Hyderabad Campus **Aug 2018 – Jun 2022**
Bachelor of Engineering
Coursework: Data Structures, Algorithms, Object-Oriented Design, Database

Technical Skills

Languages: Typescript, Javascript, HTML, CSS, Go, C/C++, Java, Python, SQL
Frontend Technologies: ReactJS, NextJS, VueJS, AngularJS, Vite, Tailwind, Redux, Axios, Jest (Automated tests)
Backend Technologies: Node, REST APIs, Express, GraphQL, gRPC, Spring/Springboot, Flask, Jest (Unit Test)
Databases and Frameworks: MongoDB, PostgreSQL, MySQL, DynamoDB, Redis
DevTools: Git/Github, Docker, Terraform & AWS CDK (Infrastructure as code), Jenkins (CI/CD)
Cloud (AWS): Serverless Functions, Virtual Servers, Document Storages, CDNs, Containers, AWS CloudFormation

Experience

Kfintech **Jul 2022 – Aug 2023**
Full Stack Developer *Hyderabad, India*

- Contributed to the development of the organization's fintech product, which handles over **200K daily transactions** totaling **60M USD**, by implementing foundational systems & an agile workflow, and leading the **10-member** frontend team.
- Generated over **40K USD** revenue and **20%** profit over **2 quarters** by solving **4 clients'** inefficiencies in on-boarding, through designing the Distributor Onboarding multi-tenant SaaS solution that streamlined and simplified their workflows.
- Delivered the Distributor Onboarding solution by leading the **7-member** development team, which utilized **Typescript, React** for frontend, and **Python, Postgres, Serverless & Microservice architecture** for backend, and **AWS** for cloud.
- Improved the web application latency by **35%**, and boosted the development team's efficiency by **50%**, by setting up the best practices, and optimizing the tech stack throughout the **Software Development Life Cycle (SDLC)**.

Amazon **Feb 2022 – Jun 2022**
SDE Intern *Bangalore, India*

- Increased the Amazon Vendors team's bug-solving efficiency by **75%**, by enabling easier analysis of documents stored in **S3 buckets** for bug resolution, through building an **S3 Document Viewer**, and integrating in the team's **agile workflow**.
- Reduced the document fetch and display time to under 700ms by delivering a fast and scalable S3 Document Viewer, built with **React.js, Typescript, Java, Maven**, and hosted using **AWS Lambda & AWS CloudFront**.
- Lowered the deployment time of S3 Document Viewer by **400%**, through implementing **CI/CD Pipelines** for auto deployment and setting up the infrastructure configurations using **AWS CDK & AWS CloudFormation**.
- Boosted productivity of the team's on-calls by **200%**, by eliminating the need to go to AWS console for data verification, through implementing **more than 10** Data metric dashboards in **AWS CloudWatch** and integrating them in team's wikis.

Projects

High-Performance Stock Price Simulator | *C++, Multi-threading, Low-latency Optimization, Python, Pandas, Numpy*

- Built a **High-Performance** Stock Price Simulator capable of simulating real-time market scenarios and stock pricing of **100s** of companies simultaneously, to be used for training of a stock price predictor **Reinforcement Learning** model.
- Achieved a **reduction of 35%** in the runtime of the simulation & **increase in throughput** of the simulator, by implementing **multi-threading techniques** and designing multiple **data structures, classes, and algorithms** for fast data processing.
- Improved insights of stock price data to optimize training strategies of the Reinforcement Learning model, by utilizing **Python, Pandas & Numpy** to process, analyze, and visualize the stock price data generated from the simulator.

Whiteboard Application with Real-Time Sharing | *React, Canvas API, Firebase, WebSockets, E2E Encryption, AES*

- Enhanced engagement of remote teams by **30-40%** by removing the need for screen-sharing tools to share drawings during meetings and brainstorming sessions, through developing a collaborative whiteboard application with real-time sharing.
- Implemented the scalable and high-performance collaborative whiteboard application using **React, Typescript, Canvas APIs, Firebase, and Web sockets** technology stack.
- Ensured the security requirements of the real-time shared drawings by allowing only authorized users to access the drawings, by engineering end-to-end encryption using the **AES-GCM** method.