Poojitha Challa

📞 +1-716-908-5413 | ✉ poojachkev883@gmail.com | 🔗 LinkedIn | 🐙 Github

SUMMARY

Software Engineer specializing in backend development with a focus on **payment systems, transaction processing, and financial platforms.** Proficient in Java, Go, Ruby, and AWS, with expertise in **distributed systems, API design, microservices, and secure payment flows.** Experienced in optimizing financial transactions, improving engineering processes, and enhancing system performance. Passionate about **scalable cloud-based commerce solutions** that ensure reliability and efficiency.

EDUCATION

University At Buffalo Buffalo, NY
Master's in Computer Science, AI/ML *August 2023–December 2024*
- **Courses:** Operating Systems, Computer Security, Analysis of Algorithms, Machine Learning, Deep Learning, Data Models & Query Languages, Data Mining, Data Intensive Computing, Computer Vision.

Sreenidhi Institute Of Science And Technology Hyderabad, India
Bachelor of Technology in Electrical And Electronics Engineering *August 2016–April 2020*

EXPERIENCE

RESEARCH ASSISTANT Jan 2024 - Dec 2024
SUNY- School of Applied Sciences *Buffalo, NY*
- Organized and classified meme datasets into categories such as COVID-19, politics, and social issues using GPT-4 and developed prompts to identify hateful versus non-hateful content, improving dataset accuracy by 20%
- Refined data processing workflows with RAG and Llama Index, enhancing model training efficiency by 15%.Collaborated with professors to build a dataset, advancing research into AI-driven meme analysis.

Software Developer Jan 2024 - Aug 2024
REDPRINT *Buffalo, NY*
- Developed a **cloud-based SaaS platform for gym management**, utilizing AWS (EC2, RDS, S3) and MySQL to optimize real-time data processing for high-traffic environments.
- Designed and implemented **REST APIs** for user authentication and data retrieval, increasing API efficiency by 20%.
- Integrated Stripe API for **secure payment processing**, reducing database read-write cycles by 13% and improving system performance by 11%.
- **Tech Stack**: TypeScript, HTML, React.js, MySQL, Node.js, Firebase, AWS, REST APIs.

Software Engineer Jan 2020 - Jul 2023
Capgemini *Hyderabad, India*
- Enhanced **partner bank downtime detection** algorithm , speeding up degradation detection by 3x. Worked with a cross -functional team of 5 to reroute transfers, improving success rates and processing times.
- Designed and implemented a payout system based on partner bank availability, leading a team of 3 engineers and 2 QA specialists, reducing transaction processing times by 30% and boosting partner satisfaction by 25%.
- Led the phased introduction of new features with **canary and phased deployment** , which reduced customer impact by 30% and boosted system stability during bank downtimes by 18%.
- Took ownership of the **FTS Service Database Migration** to a new RDS instance , implementing DB archival and partitioning to support traffic of 500TPS , up from 50TPS , eliminating database bottlenecks.
- **TechStack**: Java, Golang, Python, Kubernetes, Redis, Memcache, AWS, Kafka, Docker, REST APIs.

TECHNICAL SKILLS

Languages: Java, Python, C,C++,C#, Golang,Go,Ruby
Cloud Platforms:AWS (EC2, RDS, S3, Lambda, SageMaker, SES), GCP, Firebase
Frameworks & Tools: React, CSS, Docker, Kubernetes, Stripe API
Databases: SQL Server, MySQL, PostgreSQL, MongoDB, Firestore
Development Practices: Microservices Architecture, Containerization, Agile Development, CI/CD
Machine Learning & AI : TensorFlow, Keras, Scikit-learn, Deep Learning, NumPy, Pandas, PyTorch

PROJECTS

SaaS Web Application: Client Management-RedprintFitness | *MySQL, Cloud Firestore, React, AWS*
- Developed a SaaS Gym Management System using MySQL, Cloud Firestore, React, and AWS (EC2, RDS, S3), ensuring scalability and reliability for real-time analytics and data storage and Optimized AWS auto-scaling, reducing infrastructure costs by 15%.

Pintos OS Enhancement | *C, Linux, Process Scheduler, Memory Management*
- Enhanced Pintos OS by implementing advanced thread synchronization, dynamic priority scheduling, priority donation, and a multi-level feedback queue scheduler. Reduced idle CPU time by 95%, improved thread-switching efficiency by 70%, and increased system fairness and responsiveness.