VAMSI KRISHNA PALLENI

(631)-681-3751 | vamsi45.vp@gmail.com | U.S.A | Linkedin | GitHub

EDUCATION

Stony Brook University **Aug. 2021 – Dec 2022**
Master's in Computer Science **Stony Brook, NY**

Course Work: Distributed Systems(Scott Stoller), Analysis of Algorithm(Michael A.Bender), Data Science

GITAM University **Aug. 2015 – Mar. 2019**
Bachelor's in Computer Science and Engineering; CGPA: 8.81/10 **Visakhapatnam , India**

Course Work: Data Structures, Analysis of Algorithms, Operating Systems, Database Systems

SKILLS

Programming Languages : Java, Python, C, C++, JavaScript
FrameWorks and Cloud Technologies : Spring, Hibernate, React, Jenkins, Docker, Fault Tolerance, CPI, AWS
Databases : PostgreSQL, MySQL, MongoDB, HANADB

EXPERIENCE

Electronic Arts May. 2022 – Aug.2022
Software Engineer Intern Redwood City, CA

- Developed JWK caching system by building shared framework with spring, caching system maintained on both scheduled& on-demand refresh which reduce chance of JWK cache not initialized and Integrated all the current clients to the framework at production level, resulted in 10% System performance improvement.
- Framework is extended to all EA services for those who support JWK which decreased 20% of the potential ability to load files during application starts and made easy configuration for any future mapping.
- Published a Design document by collaborating with C&I team to verify and validate the requirements and solutions for the Project.

Stony Brook University Jan, 2022- May 2022
Graduate Student Researcher Stony Brook,NY

- Developed pipeline for data collection from API's like Twitter, Newsapi with more data accuracy & completeness and integrated with cloud storage providers such as Amazon S3 and GCP Storage.
- Building Disaster alert and notification services system with interactive maps and data access by collaborating with USGS.

Accenture June. 2019 – Aug. 2021
Application Development Analyst Hyderabad, India

- Worked on Designing & Building Fault Tolerance in High Volume Distributed Systems at Client API gateway which fans out to several underlying subsystems with peak of over 10K dependency requests per second.
- The Fault Tolerance Design Controlled API downtime caused by single API Dependency failure and increased system availability and decreased latency and reduced 2+hours downtime/month.
- Migrated billing System from a Pharma company DC to AWS with zero down time. Moved Millions of rows of data over AWS.
- Incorporated various systems like S4hana, Fiori and Custom applications(Stock index, Sprinklr) through File,HTTP, ODATA, SFTP Protocols by Authentication and formatting to CPI.

Augmented Byte Mar. 2018 – Jul. 2018
Software Engineer Intern Hyderabad, India

- Designed and Developed automatic outage detection software using UiPath. This feature allowed team to reduce troubleshooting costs, minimized downtime by less than 2 minutes and rapidly resolved customer service issues with 90% success rate.
- Created a web application using Java, Spring Framework and RESTful APIs to store the status of financial sponsor budget in PostgreSQL database for use by 50+ NPO advisors.
- Implemented scheduler to retry failed requests, reducing manual labor by 20%, Increasing fault tolerance of the application by 35%.

PROJECTS

DiemBFT-v4 Implementation Under Prof. Scott Stoller(Stony Brook Univ.)

- Implemented Facebook's DiemBFT consensus Algorithm, as described in the paper "DiemBFT V4 – State Machine Replication in the Diem Blockchain," to create a fault-tolerant distributed system in conjunction with Blockchain. Python | DistAlgo

Distributed Blockchain-Based Authentication and Authorization Protocol for Smart Grid

- Implemented protocol that combines a novel blockchain technique with the decentralized authentication and immutable ledger characteristics of blockchain architectures ideal for power systems To realize both identity verification and resource authorization for smart grid systems.

Analysis of Video Streaming under HTTP1.1, HTTP2 and HTTP3 (Networks, HTTP, Data Analysis)

- Built and simulated different versions of HTTP servers for video streaming with varying network conditions and latency. Analyzed network data and found HTTP1.1 performing better than server push and HTPP3 for video streaming.