Sujatha Pushparaj

1215 E Vista Del Cerro Dr, Apt 1023, Tempe, AZ 85281

spushpar@asu.edu linkedin.com/in/sujatha-pushparaj-93931712a/ +1-928-671-8544

EDUCATION

Arizona State University - Tempe, AZ *August. 2022 – May. 2024*

Master of Science - Computer Engineering (Computer Systems) GPA: 3.93/4

courses: Foundations of Algorithms, Software Security, Advanced OS, DB Systems Implementation, Software Design

Sri Krishna College of Engineering and Technology - Coimbatore, India *June. 2014 – April. 2018*

Bachelor of Engineering in Electronics and Communication Engineering GPA: 9.63/10

courses: Data Structures and OOP, Digital Image Processing, Computer Networks and Embedded systems

TECHNICAL SKILLS

Programming: **Backend:** Node.js, Express.js, Typescript, JAVA, AWS, Elasticsearch
Frontend: Javascript, React JS, Redux JS, HTML, CSS, SASS, D3.js, Plotly, Styled Components
Database: SQLite, PostgreSQL, Apollo GraphQL, RDS, Dynamo DB
Testing Frameworks: Unit testing - JEST, E2E Testing - Selenium, Cypress
Others: Rest APIs, Serverless design, CI/CD pipeline, cloud infrastructure

Software Tools : Git, NPM, Yarn, Webpack, Postman, Github & Atlassian tools (JIRA, Bitbucket)

WORK EXPERIENCE

Amazon, *Boston, MA*

SDE Co-op, Amazon Robotics *July. 2023 – December. 2023*

- Established a unit testing framework for the front-end code and seamlessly integrated it into the deployment pipeline.
- Managed the product's entire software development lifecycle for UI features, encompassing the creation of an in-app resource page and a unified preview component for various file types within the application.

Arizona State University, *Tempe, AZ*

Student Mobile Developer, University Technology Office *August. 2022 – April. 2024*

- Designed and developed react based front-end components in ASU rewards web and mobile app used by 50k+ students/staffs and deployed the changes to production using Terraform and AWS serverless architecture.
- Reduced user reported defects by 50% by performing E2E testing on mobile application and fixing the identified bugs.

Cisco Systems, *Bangalore, India*

Software Engineer, CISCO DNA Center *July. 2021 – July. 2022*

- Transformed 90% of the code base handling inventory management from JQuery to React based scalable components.
- Developed a plugin to manage beacon LED on network switches that helps network administrators to locate devices. Also, designed and developed graphQL layer to optimize the API calls for the feature.

Soliton Technologies, *Coimbatore, India*

Project Engineer *June. 2018 – July. 2021*

Plexus — Soliton's IC evaluation GUI framework

- Identified and eliminated memory leaks and redundant re-renders of UI components which significantly enhanced the app's performance by reducing the memory usage by 50% and 3X faster loading of UI components.
- Redefined data fetching flow by incorporating custom sampling mechanism in backend SQL query that increased the maximum storage limit from 1GB to 25GB while retaining the performance.

SISU Lab — Cinematic Robot Controller's GUI

- Performed requirements analysis, design, development and test management for phase 1 of the project.
- Automated installer creation steps which improved the productivity of the team by cutting down the time to create installer from 2 hours to 15 minutes and reduced the manual efforts and errors.
- Researched best practices in software engineering and successfully incorporated data-driven models in D3 based UI, which resulted in faster loading of complex UI and scalable codebase.

ACADEMIC PROJECTS

Android Malware Static Analyzer

- Created a python application to classify Android APKs into malware and benign groups and achieved 90% accuracy.
- Improved the classification performance by 5% using adaptive boosting technique.

Minibase Variant using Map datastructure

- Enhanced compliance and reliability by executing a shift from tuples to maps, integrating rigorous prototyping into the extensive overhaul of critical database functions.
- Introduced novel index types to optimize sorting and join operations, resulting in notable performance enhancements.

RISC-V XV6 OS Using C and Assembly

- Implemented lazy memory allocation technique to improve the efficiency of memory management unit in xv6 OS.
- Designed and developed a module to enable self threading which is not supported in native xv6 code.