Prakash Natarajan

Toronto, ON M4Y3C1 | +1 716-361-8720 | mail@prakashn.com | LinkedIn | GitHub | Portfolio

Software Engineer

Microservices Developer ▪ Highly Scalable Design and Development ▪ Cloud Enthusiast

- **Innovative software engineer** offering experience in the full software development lifecycle – from concept through delivery of next-generation applications and customizable solutions.
- **Experience in many different technologies,** like C#, C++, Go Lang, Java, JavaScript, Python.
- **Architected many applications** – for professional as well as personal projects.

Technical Tools

C#, C++, Go Lang, Java, Python, JavaScript, HTML5, CSS3, ASP.NET Core, .NET framework, NodeJS, ExpressJS, ReactJS, AngularJS, Spring Framework, Azure, AWS, Firebase (GCP), Ubuntu, Windows, Ansible, Docker, Kubernetes, XML, TCP/IP, REST, Visual Studio, Eclipse, SQL, MS SQL Server, MySQL, IIS and more

Professional Experience

Software Engineer, 3/16 to 7/19 – FACTSET RESEARCH SYSTEMS, Norwalk, CT, USA

Architected monitoring infrastructure, Enhanced microservice architecture of server, Implemented changes to toolkits in C++, C# and Java codebase, Automated deployments of our server. ***Achievement Highlights:***

- Designed a full stack application for detecting downtime in any of our services and added notification system for alerting on-call person. Configurations are rendered from Redis, backend monitoring logic is in C++, middleware web server is in Go and frontend in React JS. This helped to monitor our services all the time without manual intervention. During issues, this helped us to easily edit servers in load balancer. Frontend code added capabilities to segregate servers by both application and region.
- Automated windows server deployments using Ansible. This reduced the deployment time by 80%.
- Enhanced permissions microservice to use Zookeeper for detecting duplicate user subscriptions. This used to have a http server listening on all our boxes. This reduced the load on each server by almost 15%.
- Created options chain microservice, which enabled our clients to get options chain data using our toolkits.
- Added enhancements to both server and toolkits based on client requirements.

Software Development Engineer Intern, 6/15 to 8/15 – AMAZON WEB SERVICES LAMBDA, Seattle, WA, USA

Designed a framework for easily integrating event sources into AWS Lambda. ***Achievement Highlights:***

- Developed a framework in AngularJS 1.7 for rendering user interface dynamically from configuration file. Data rendering in backend is managed by interface implementation in Java Spring. This reduced the time for adding new servers from 2 weeks to just 2 days.

Programmer Analyst, 6/11 to 3/14 – COGNIZANT TECHNOLOGY SOLUTIONS, Chennai, TN, INDIA

Enhanced few insurance-based applications in .NET framework 3.5. ***Achievement Highlights:***

- Improved the dashboard for displaying time critical defects in ASP.NET, collaborated with quality assurance team for running and maintaining automated scripted and integrated the results with dashboard. Created a mailing windows service in C#.Net to email screenshot of defects to the team.

Education

UNIVERSITY AT BUFFALO, STATE UNIVERSITY OF NEW YORK – Buffalo, NY, USA 8/14 to 2/16

Master of Science in Computer Science and Engineering (MS-CSE)

ANNA UNIVERSITY – Chennai, TN, INDIA – 6/07 to 5/11

Bachelor of Engineering in Electronics and Communication (BE-ECE)

Personal Projects

Codeforces Ranking Notification System, GitHub Link

Designed a system to subscribe and receive notification to ranking changes of Codeforces.com . ***Highlights:***

- Architected entire backend which is microservices of event driven queue based model. Used dockers to deploy each microservice. API's are created using ASP.NET core webapi and services for manipulating queue are done using ASP.NET Console App.
- STACK: C#, ASP.NET Core, RabbitMQ, Docker, MS SQL Server, visual studio code.

Serverless Number to String Converter, GitHub Link

Created a infinitely scalable serverless function to get string representation of a number. ***Highlights:***

- Eg. 123 gives one hundred twenty three. Supports numbers up to 10 to power 26.
- STACK: C#, Azure functions, Visual Studio Code.
- Application link

Reminder App

Architected server for storing reminders and alerting users based on spaced repetition. ***Highlights:***

- Server is written in Python; data stored in AWS Dynamodb; deployed as AWS lambda functions; accessed using HTTP Trigger (AWS API Gateway); authenticated using AWS Cognito; uses Google's protobuffer as data interchange format. Currently in alpha stage yet to be release to public.
- STACK: Python, AWS Lambda, AWS API Gateway, Google's Protobuffer, AWS Cognito.