

# SWETHA PATURU

☎ (857)313-2977 ✉ [paturu.s@northeastern.edu](mailto:paturu.s@northeastern.edu) [www.linkedin.com/in/swetha-paturu](https://www.linkedin.com/in/swetha-paturu) [github.com/swethapaturu9](https://github.com/swethapaturu9)

## Education

### Northeastern University, Boston MA

Expected May 2024

*Master of Science in Computer Software Engineering*

Relevant Coursework: Networks & Cloud Computing, Database Design, Agile Methodologies

### National Institute of Technology Warangal, India

May 2021

*Bachelor of Technology in Electronics & Communication Engineering*

## Technical Skills

<b>Certifications</b>	Certified Kubernetes Application Developer (CKAD)
<b>Languages &amp; Frameworks</b>	Java, Python, JavaScript, Spring MVC, Spring Boot, React.js
<b>Databases</b>	MySQL, MongoDB
<b>DevOps</b>	Jenkins, GitHub Actions, Docker, Kubernetes, Helm, Terraform, Pulumi, Packer

## Work Experience

### Larsen & Toubro Infotech Ltd.

July 2021-July 2022

*Software Engineer - Cloud Native*

Hyderabad, India

#### ***Skill Testing Platform Development***

- Developed a grading microservice as part of transforming a large-scale employee skill testing system into Spring Boot microservices, using RabbitMQ to serve 15,000 users
- Added unit and integration tests using JUnit and Spring Test achieving a total test coverage of 90%
- Implemented CI/CD by containerizing the microservice, integrating it into the existing Jenkins pipeline, and orchestrating deployment to Kubernetes
- Collaborated with multiple teams, in a Scrum-based setting, leveraging Jira for project tracking

#### ***IBM Multicloud Management - DevOps***

- Played a pivotal role in a four-person team exploring Multi-Cloud and Hybrid Cloud technologies, using IBM CloudPak for Multicloud Management to develop a Proof of Concept
- Managed the deployment of Kubernetes resources on both OpenShift and Google Kubernetes Engine with Helm charts, leveraging the capabilities of the IBM Multicloud Management
- Established an automated CI/CD pipeline using Jenkins to streamline code deployments within IBM Multi-Cloud
- Led the presentation of the Multi-Cloud Management tool's functionalities to the Sales team, using a scalable, multi-cloud deployment strategy using Google Kubernetes Engine and OpenShift
- Maintained thorough project documentation for transparent communication and comprehensive record-keeping

## Projects

### **Cloud Infrastructure & Application Development** | *Java, AWS, GitHub Actions, Pulumi*

- Developed a Assignment Management Spring Boot application featuring RESTful User and Assignment controllers, integrated with a securely configured AWS RDS MySQL database
- Simplified deployment processes by creating an Amazon Machine Image (AMI) using Packer, automated via GitHub Actions, and improved system monitoring with CloudWatch agent integration
- Deployed the application on Amazon EC2 and introduced scalability with an Auto Scaling group, configured to dynamically adjust resources based on CPU utilization thresholds monitored by CloudWatch alarms
- Configured an AWS Application Load Balancer with SSL termination, ensuring secure and efficient request routing
- Acquired and managed a custom domain in AWS Route 53, integrating it with the load balancer for domain resolution
- Set up cloud infrastructure resources, such as Virtual Private Cloud, subnets, security groups, IAM roles, and EC2 instances, utilizing Pulumi with Python

### **File Upload and Cloud Storage Integration** | *AWS Lambda, Python, Google Cloud Storage*

- Developed a feature in an application to automatically transfer user-uploaded files to Google Cloud Storage buckets
- Integrated AWS SNS and Python-based AWS Lambda for file processing and real-time email notifications to users

### **Cloud Resume Challenge** | *React.js, Terraform, AWS S3, DynamoDB*

- Created a portfolio website with React.js, deployed on AWS S3 secured with AWS CloudFront
- Built a dynamic view counter using AWS Lambda and DynamoDB, employing Terraform for efficient infrastructure management and deployment
- Configured GitHub Actions for the CI/CD of both the website and Terraform code, with Terraform backend on S3