# CSE 546 Cloud Computing

**Laksh Gangwani** **ASU ID: 1225578211**

My main contribution in project 1 for CSE 546 was on the architecture design and scaling activities of the Auto Scaling Group. I made sure of the creation, management, and updation of AWS resources. I used CloudFormation to provision and manage my infrastructure instead of manually configuring infrastructure, since it is a tool which enables the automated creation and management of infrastructure on AWS. By doing this, infrastructure can be defined as code. It makes it easier for me to manage and replicate infrastructure across multiple AWS accounts.

For the project, I created two stacks, which are the network and infra using CloudFormation. In the Network stack I defined the configuration of my network components, which include Virtual Private Clouds, subnets, security groups, network ACL, InternetGateway and routing tables. The security group allows traffic from port 22 and port 80 (HTTP). In the Infra stack, I created the resources which are required for this project. This includes two queues, RequestQueue and Response Queue, two s3 buckets to store the result, Auto Scaling Group, Alarms and Policies. Both stacks can be easily managed and updated.

For the Auto Scaling implementation, I created two CloudWatch Alarms, an ASG, and two scaling policies. I preferred to use dynamic scaling instead of manual/step scaling, since the scaling decision is based on the metric I am using and it changes dynamically, i.e. based on the actual demand. I avoided simple or step scaling because in case of sudden spikes it won't be as effective, since it relies on fixed thresholds. I used the "ApproximateNumberOfMessagesVisible" metric to monitor the RequestQueue, attached an IAM role to the ASG, and decreased the Cooldown period of the ASG to accomplish scaling up and down quickly. I realized that with dynamic scaling, the system can adjust to the changing queue sizes, ensuring that the system is always properly provisioned and responsive to user demand.

To ensure consistency, I monitored and updated the two stacks, network and infra. I also monitored the "Activities" tab for the ASG and compared it against the RequestQueue metric "ApproximateNumberOfMessagesVisible" to ensure that everything was working correctly.

Moreover, I used CloudWatch metrics to monitor and visualize metrics of AWS resources and I created a CloudWatch dashboard to keep track of the metrics which are important for this project, for example, I plotted the graph on how the number of desired instances are changing with reference to the Approximate number of message visible in the Request Queue.

This project provided me with hands-on experience on a range of AWS services. Moreover, I learned a lot about different types of Scaling and about the management of the Infrastructure as a code using CloudFormation. I was able to scale up to all 20 instances (Maximum) and then scale down to 0 instances (Minimum) within 6 minutes, demonstrating my understanding of Auto Scaling.