SITUATION: The situation is I have a web application and want to integrate it to Auto scaling group and elastic load balancer (ELB) and to utilize it in order to adjust the instances depends upon the incoming traffic so we have to make it high availability and also create health checks for scaling automatically and take a AMI for instance back up.  
  
TASK: The task is to create an instance which runs web app and attach an auto scaling group.  
  
ACTION:  
  
STEP1. Create an EC2 -Instance and configure all the dependencies and write a simple user data script that displays the output when we run the instance. Also create an AMI for the back up purpose.  
  
STEP 2. Create an ELB in the AWS Management Console here I have taken a classic load balancer, configure listeners like (HTTP or HTTPS) and health checks for the ELB by giving the AZ’s where the instance is launched.  
  
STEP 3.Before creating an auto scaling group create a launch configuration which is integrated in ASG and give the desired number of instances and configure scaling policies also Define health checks to monitor the instances' status Configure the Auto Scaling group to use the ELB created.  
  
STEP 4.We can choose any metrics like CPU utilization ,Memory space and give the scaling policies based on application metrics b setting a threshold value.  
  
STEP 5. Next is to Configure SNS (Simple Notification Service) for sending notifications in case of scaling events or issues by giving our own mail ID [shaziashaik196@gmail.com](mailto:shaziashaik196@gmail.com) or any and can also create Cloud Watch metrics.  
  
RESULT: The output we get is a perfectly designed instance with auto scaling group and load balncer when if we spin the auto scaling group or instance the instances are automatically created depends upon our desired value given and which results ensuring the high availability and cost-effectiveness by dynamically adjusting the number of instances based on traffic. The use of AMIs allows for quick instance provisioning, and health checks ensure the availability of instances. Notifications and advanced features contribute to a well-monitored and optimized system.