**USE CASE 1** : Create Notification Services with Spring boot and AWS services (SQS , SES , EC2, S3)

* Create SQS Service ( use either of FIFO or standard )
* Create a HTML notification template with placeholders and keep it in S3 bucket.
* Create Spring boot application for sending messages to the SQS queue and read asynchronously messages from SQS queue (producer -consumer pattern).deploy this service on EC2
* Create a SES service (Please note we need to have a domain name to configure this service).
* Once the SES is configured the messages which are read from the SQS queue will get an appropriate HTML notification template from S3 bucket ,populate the placeholders with the data from the SQS message and send mail notification.

**USE CASE 2**:  Create AWS API Gateway to call AWS Lambda function which sends Email using AWS SES service

Stage 1 : Create API-Gateway (resources & methods)

* Open Amazon API Gateway console. Create new API by entering a name.
* On the API tree view, just click the Actions button in the menu to “Create Resource” and name the resource.
* With the selected resource, create a new method and select the HTTP verb with the method.

Stage 2 : Create a Lambda proxy integration. Because it's a proxy integration, you can change the Lambda function implementation at any time without needing to redeploy your API.

   Reference : <https://docs.aws.amazon.com/apigateway/latest/developerguide/api-gateway-create-api-as-simple-proxy-for-lambda.html#api-gateway-proxy-integration-create-lambda-backend>

Stage 3 : Following steps to be send mail from AWS Lambda

* AWS Identity and Access Management (IAM) permissions for Lambda to execute the API call.
* A [verified Amazon SES identity (domain or email address)](https://docs.aws.amazon.com/ses/latest/DeveloperGuide/verify-addresses-and-domains.html).
* A Lambda function with logic for sending email via Amazon SES.

The above steps will provide solution to create a serverless notification service