Assignment #04

Due Date

Section	Due Date	Grading Deadline
Saturday & Online	09:00pm on 06/28/2019	09:00pm on 07/12/2019

Getting Help

> **Info**
>
> Ask all your questions on Piazza. Assign **assignment4** tag to your posts.

Assignment Weightage

Individual Assignment Weightage on Course Grade of this assignment is **10%**.

Objectives

Infrastructure as Code

Order of CloudFormation Stack Execution

network -> application

The Application Stack

1. Create a new CloudFormation template `csye6225-cf-application.json` or `csye6225-cf-application.yaml` that can be used to setup application resources. This template is separate from the `csye6225-cf-networking.json` or `csye6225-cf-networking.yaml` template created in previous assignment.
2. Create shell script `csye6225-aws-cf-create-application-stack.sh` to create and configure required application resources using AWS CloudFormation. Script should wait until the stack is terminated and print message on script if it was success or failure. Script should take `STACK_NAME` as parameter and the CloudFormation should not contain specific hardcoded values. Your shell script may assume that the CloudFormation template is in the same directory as your script.
3. Create shell script `csye6225-aws-cf-terminate-application-stack.sh` to delete the application CloudFormation stack. Script should take `STACK_NAME` as parameter. Script should wait until the stack is terminated and print message on script if it was success or failure. Your shell script may assume that the CloudFormation template is in the same directory as your script.

Security Group for Web Servers

Create a EC2 security group for your EC2 instances that will host web application. Add ingress rule to allow TCP traffic on port `22`, `80` and `443` from anywhere in the world. This security group will be referred to as webapp security group.

Security Group for DB Servers

Create a EC2 security group for your RDS instances. Add ingress rule to allow TCP traffic on port `3306` for MySQL or whatever port the database you are using only from the webapp security group.

EC2 Instance

Your CloudFormation template for application stack, `csye6225-cf-application.json` or `csye6225-cf-application.yaml` should be updated to create EC2 instance with following specifications:

> **Note**
>
> For any parameter not provided in the table below, you may go with default values. For this assignment the EC2 instance can belong to the VPC you have created in networking stack.

Parameter	Value
Amazon Machine Image (AMI)	Your custom AMI id
Instance Type	t2.micro
Protect against accidental termination	No
Root Volume Size	20
Root Volume Type	General Purpose SSD (GP2)

Webapp security group should be attached to this EC2 instance. *Make sure the EBS volumes are terminated when EC2 instances are terminated.*

DynamoDB Table

Create a DynamoDB table with provided configuration. You may use default value/settings for any property not mentioned below.

Property	Value
Table Name	csye6225
Primary Key	id
Primary Key data type	string

S3 Bucket

> **Manual Step**
>
> Execute this task manually. This does not have to be part of your CloudFormation template.

Create a S3 bucket with bucket name `YOUR-DOMAIN-NAME.TLD.csye6225.com` where `YOUR-DOMAIN-NAME.TLD` is your domain name. Make sure bucket is private. You can leave everything else set to default.

RDS Instance

> **Warning**
>
> Setting `Public accessibility` to `true` will expose your instance to the internet.

Your RDS instance should be created with following configuration. You may use default value/settings for any property not mentioned below.

Property	Value
Database Engine	MySQL/Postgres/Aurora
DB Instance Class	Any
Multi-AZ deployment	No
DB instance identifier	csye6225-su19
Master username	csye6225master
Master password	csye6225password
Subnet group	Subnet for RDS instances
Public accessibility	Yes
Database name	csye6225

Web Application User Stories

API specifications [https://app.swaggerhub.com/apis-docs/csye6225/csye6225-summer2019/2.0.0]

1. As a user, I want to add, update or delete images (cover page) for books. I should be authenticated & authorized to be able to perform these operations.
    a. As a user, I want to add a images for the book. Only `.png` , `.jpg` ,, and `.jpeg` formats are supported.
    b. As a user, I want to update book image. Updating images should replace existing image.
    c. As a user, I want to delete image for the book.
2. Only one image can be uploaded per book.

3. **As a user, I expect image to be stored in Amazon S3 bucket when the application is running on cloud (when running in EC2 instance).**
4. As a user, I expect image to be stored locally in some directory when running application is running locally on developer's machine.
5. Metadata about book image must be stored in RDBMS such as MySQL or Postgres.
    a. When the application is running locally, it must use a local database instance.
    b. **When the application is running on cloud, it must use the AWS RDS service.**
6. **As a user, when I make a `GET` request, the URL returned for images in the private S3 bucket should be pre-signed with expiration of `2` minutes.**

Documentation

- AWS::DynamoDB::Table [http://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-resource-dynamodb-table.html]
- AWS::RDS::DBInstance [http://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-properties-rds-database-instance.html]
- AWS::RDS::DBSecurityGroup [http://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-properties-rds-security-group.html]
- AWS::RDS::DBSecurityGroupIngress [http://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-resource-rds-security-group-ingress.html]
- AWS::EC2::SecurityGroup [http://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-properties-ec2-security-group.html]

- AWS::EC2::SecurityGroupEgress [http://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-resource-ec2-security-group-egress.html]
- AWS::EC2::SecurityGroupIngress [http://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-properties-ec2-security-group-ingress.html]
- Getting Started with Amazon RDS [https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/CHAP_GettingStarted.html]
- Creating Pre-Signed URLs for Amazon S3 Buckets [https://docs.aws.amazon.com/sdk-for-go/v1/developer-guide/s3-example-presigned-urls.html]

Submission

Danger

Assignment will be considered late if commits are made to `master` and `feature` branch after due date.

1. All work for this assignment must be done on **assignment4** feature branch and merged to master when you are dev complete.
2. All team member's feature and master branches must be in-sync.

Grading Guidelines

Warning

Following guidelines are for information only. They are subject to change at the discretion of the instructor and TA.

TBD