

ITMO 544/444 Mini Project 1

Due date October 22nd 11:59 PM

Worth 50 points

MP1

Contains 3 Parts

1. Installation and deployment of resources

Create a script that will automate resource deployed on EC2

- a. Script A - installenv.sh will run and launch ec2-run-instances quantity of 2 instances launching an instance of Ubuntu server and passing the install.sh to it via -f
- b. Script B - install.sh will pull all system pre-reqs and required libraries and install AWS SDK library via composer as well as wget your project down and deploy it to the correct directory copying your custom-config.php to the correct location.
- c. Script C - will deploy the load balancer and register your two instances with the load balancer. Also create a SimpleDB domain, an SQS queue, and SNS Topic – it can be the same script or another PHP script run from the command line 1 time

2. Uploading & Processing

In index.php

- a. will have a basic form that asks for email, phone, and a picture to upload and post this data to process.php
- b. Send a subscription notice to the user so that they may receive SMS later at the end of the project (this could be done in part 1C as well)

In process.php

- c. add uploaded photo to S3 bucket - set metadata tags for a md5 hash and an epoch timestamp
- d. return S3 URI for uploaded object
- e. create Item in SimpleDB that contains:

- rawurl - (from above)
 - email
 - bucketname
 - filename
 - phone
 - id - use php method uniqid() <http://us2.php.net/manual/en/function.uniqid.php>
 - finishedurl - initial value is blank
- f. Use SQS to place a queue with the id as the sqs body
 - g. use the system sendmail to send an email (really easy) thanking them for submitting the altered image

3. Rendering & Cleanup

In resize.php

- a. read the queue message body into a variable.
- b. Select the Item from your SimpleDb that matches the ID
- c. parse the response object -
- d. use S3 getObject - store in the /tmp directory
- e. Pass the downloaded object to the php gd library and add a water mark - image provided
- f. Upload the newly rendered image back to the S3 bucket the original came from
- g. Update the SimpleDB object giving the URI of the S3 object to the 'finishedurl' Attribute Value Pair in Simple DB

In cleanup.php

- h. Send SMS via SNS to let the user know the URL for downloading the finished picture (first time they will have to register to receive sms)
- i. Mark the ACL for this object as Public::Read in S3
- j. Mark the Object expiration time for the S3 bucket as 10 minutes
- k. Display all the images in the S3 bucket on the screen. (Before and After)
- l. Consume/Destroy Queue