**ACIT 3640 – Cloud Computing with AWS**

**(Lab 6)**

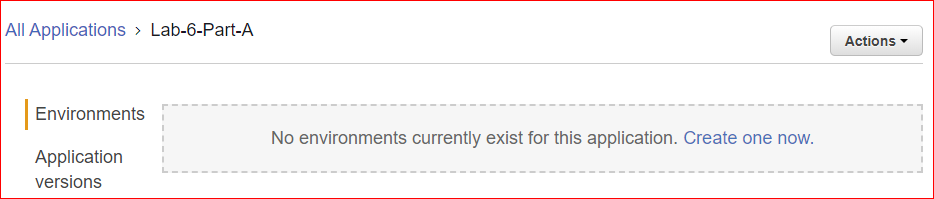
**Launching an Application in AWS Elastic Beanstalk**

“AWS **Elastic Beanstalk** makes it even easier for developers to quickly deploy and manage applications in the AWS cloud. Developers simply upload their application, and **Elastic Beanstalk** automatically handles the deployment details of capacity provisioning, load balancing, auto-scaling, and application health monitoring.” Source: <http://aws.amazon.com/elasticbeanstalk/faqs/>

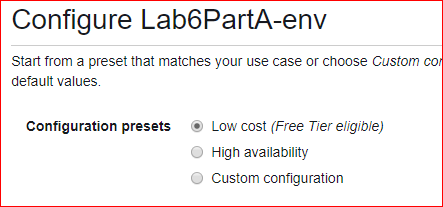
**Part A**

In this part, we will deploy a default website using AWS Elastic Beanstalk.

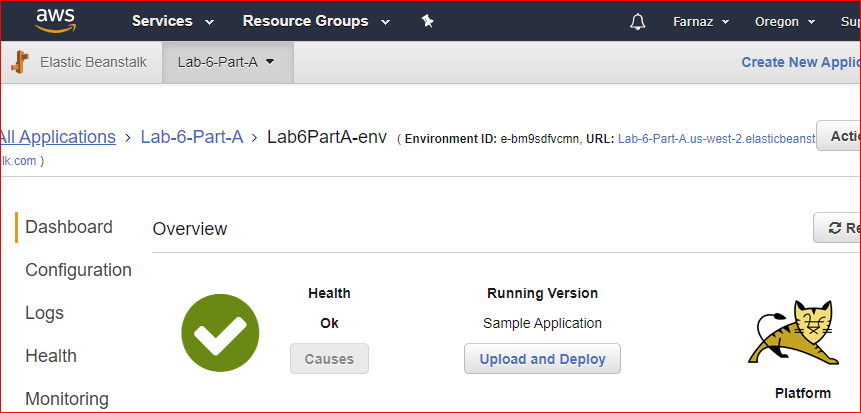
1. Login into AWS make sure to choose Oregon as your region for the entire lab.
2. From the Services menu, select Elastic Beanstalk.
3. In the top right corner of the screen, click on the “Create New Application” button.
4. Enter a name Application Name like “Lab-6-Part-A” and click Create.
5. On the EB applications screen, under Lab-6-Part-A, click “create one now.”



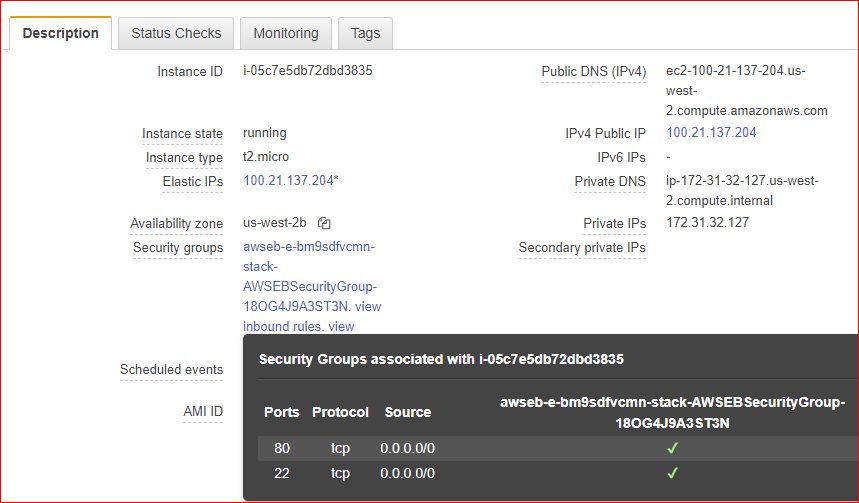
1. On the “Choose an environment tier” screen, click “Web server environment.”
2. Enter a Domain name, like Lab-6-Part-A, and click Check Availability.
3. On the “Create new environment,” go to the Platform field. Select “Tomcat” from the Preconfigured choices.
4. Make sure the “Sample application” radio button is selected.
5. Click the gray “Configure more options” button at the bottom of the screen.
6. Make sure to select low cost (free tire eligible) for configuration presets.



1. Click Modify under Security window.
2. Add your “EC2 key pair” and Save.
3. Click Modify under the Notifications window and enter your Email address.
4. Click the blue “Create environment” button at the bottom of the screen.
5. Monitor the activities in the black screen area while your EB environment is being built.
   1. What activities are occurring?
   2. Messages should be displayed indicating that an instance is being created, a security group, and Elastic IP Address (EIP), etc.
6. While the Beanstalk environment is being created, open another browser window. Log into AWS and go to your EC2 Dashboard. An instance is being created.
7. Continue to monitor your Beanstalk environment (it may take up to 4 minutes).
8. After a few minutes, you should eventually see the message “Successfully launched environment.” The screen will change and display a green Health button.

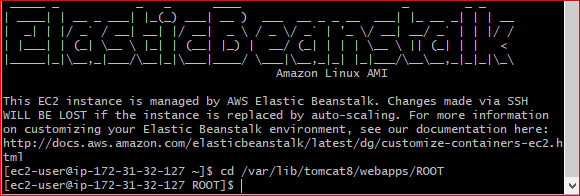


1. When your environment is ready, click on the blue hyper-link in the top middle of the screen to access your website. This may take some time, so be patient. You should see a web page that says, “Congratulations!” Now you are going to change this webpage to display your name.
2. Go to your EC2 instances dashboard.
3. Click on the Security Group associated with the instance in description tab. Do you have inbound SSH access (TCP port 22) to this instance? (You should.)



1. SSH into this instance. The username is “ec2-user.”
2. Navigate to this directory: /var/lib/tomcat8/webapps/ROOT

Tomcat is a webserver. This directory contains the website’s homepage called “index.jsp.”



1. Use a Linux text editor, like nano or vi, open the index.jsp file. You may have to use “sudo” to open and change the file.

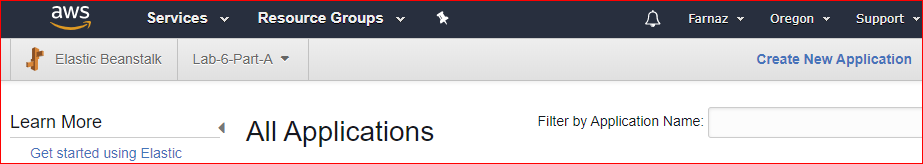


1. Find the word “Congratulations!” and change it to your name and A00 number.
2. Close your SSH connection.
3. Refresh your browser that contains the Congratulations web page. You should see your name displayed.
4. Keep your browser open and show your work as part A to your instructor when you finished all parts of the lab.

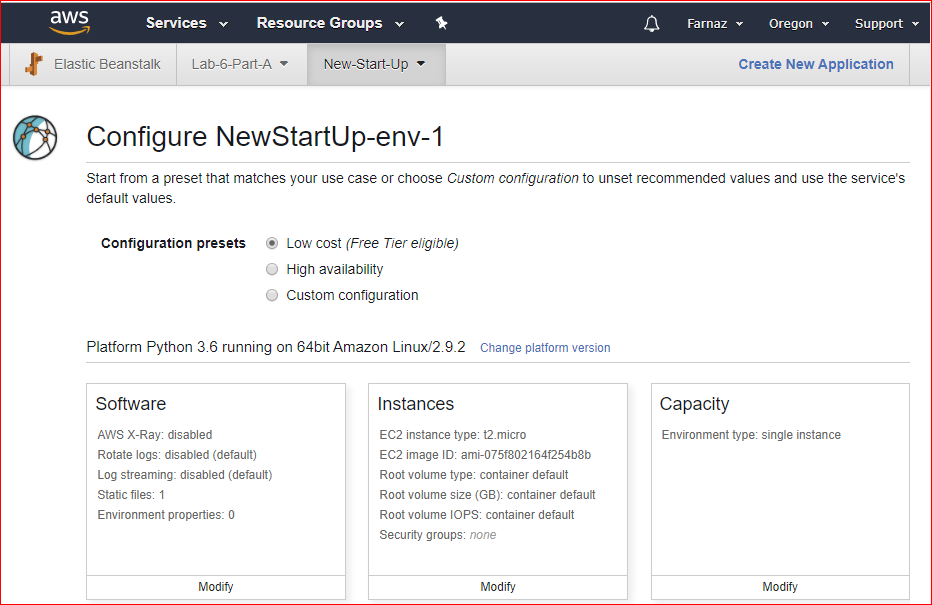
**Part B**

In this part, we will launch a startup sample App in elastic beanstalk. This application replicates a website sign-up page and customer data is stored in an Amazon DynamoDB database (key store). The idea is that this website goes viral and many people are trying to create profiles at the same time. This example uses an application stack that is stored on GitHub.

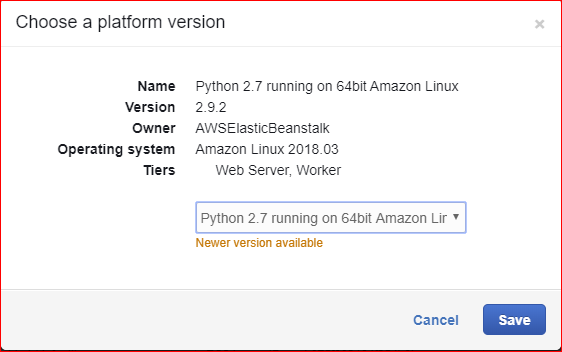
1. The lab instructions that follow are adapted from these two videos.
   1. Part 1: Run the App: <https://www.youtube.com/watch?v=rsg4YI4mljg>
   2. Part 2: Use the App: <https://www.youtube.com/watch?v=IuwfVX52PV8>
2. Download the following sample application: <https://s3-us-west-2.amazonaws.com/us-west-2-aws-training/intro-to-beanstalk/static/eb-py-flask-signup-v1.1.0.zip>
   1. The name of the file is “eb-py-flask-signup-v1.1.0.zip.”
   2. No need to unzip the file at this time.
   3. This zip file contains an entire application that will be launched automatically by EB. We will examine the contents later.
3. Login into AWS make sure you are in Oregon region.
4. From the Services menu, select Elastic Beanstalk.
5. In the top right corner of the screen, click on the “Create New Application” button.



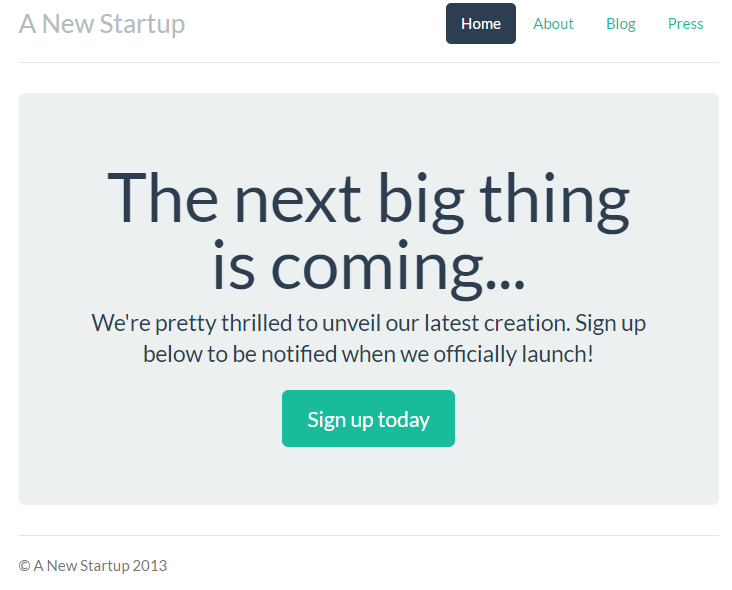
1. On the “Application Information” screen, enter an Application name, like “New Startup” and click Create.
2. Click on “Create one now”. On the “Select environment tier” screen, select “Web server environment” and click “Select”.
3. Choose a domain name and check availability.
4. Go to the “Predefined configuration” field. Select “Python” from the **Preconfigured** choices. By default, EB uses the latest version of Python (3.6). Our application has been developed using Python version 2.7. In the next steps, we will change the python version.
5. Select the “Upload your own” radio button under Application code.
6. Click “upload” to upload the zip file (eb-py-flask-signup-v1.1.0.zip) you downloaded earlier.
7. Click on “Configure more options”.
8. Important: click on “Change platform configuration”.



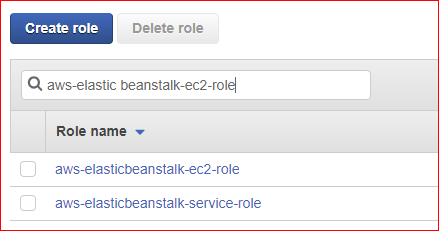
1. Change the Python version to 2.7 and save it.



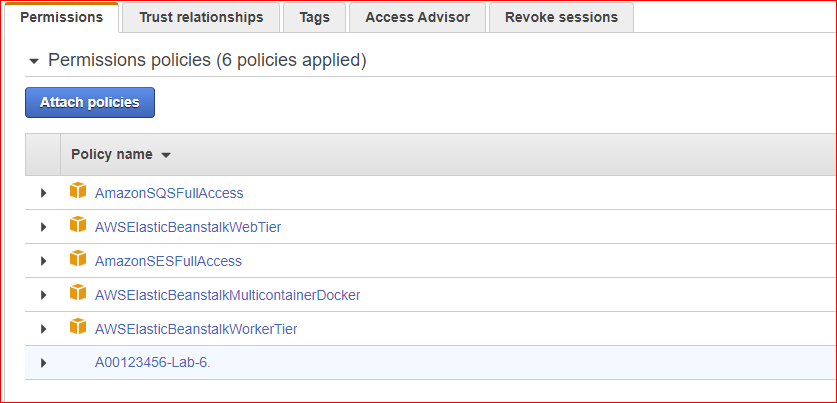
1. On “Instance” screen, make sure the Instance Type is “t2.mico.”
2. On the “security” screen, click on modify and choose your kay pair.
3. On “Notifications” screen, enter your email address. (Mybcit or Gmail, don’t use Hotmail)
4. On “Monitoring” screen, change the Health reporting from “Enhanced” to “Basic.”
5. Click on “create environment”.
6. Monitor the creation of you EB environment.
   1. Was an Elastic -IP address created? Verify by going to the EC2 Dashboard.
   2. Was an instance launched? Verify.
7. When the Health check is Green, click on URL for your website. It is in blue, at the top middle of your screen, and is labeled “Environment ID.” You can click on that link after your environment has been provisioned.  ***Do not*** click thesign up button until we have created a security policy. You should see the following screen:



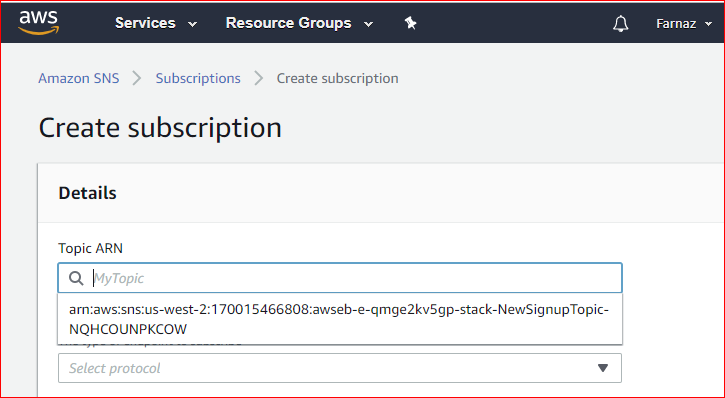
1. Now, we need to create a policy that will give anonymous users the ability to write to the DynamoDB. In addition, the policy should permit our application to send confirmation email to customers and notification email to you using SES.
2. From the Services menu in the AWS Console, select IAM (Identity & Access Management) under the Security, Identify & Compliance area.
3. On the left side, select Policies
4. Click the “Create Policy” button.
5. Download a file called “iam\_policy.json” from D2L.
6. In the JSON tab, copy and paste the code from “iam\_policy.json” file.
7. On the Create Policy page, in the “Name” field, enter a policy name that starts with your A00 number, like “A00123456-Lab-6.”
8. Click the “Create Policy” button.
9. Now, click “Roles” on the left side of the dashboard.
10. Under the list of role names, click on “asw-elasticbeanstalk-ec2-role.”



1. On the Permissions tab, click the “Attach Policy” button.
2. Select the policy you created in the previous steps, also select AmazonSESFullAccess and attached them.
3. You should have all the following polices attached. If you do not have any of them, attach it.

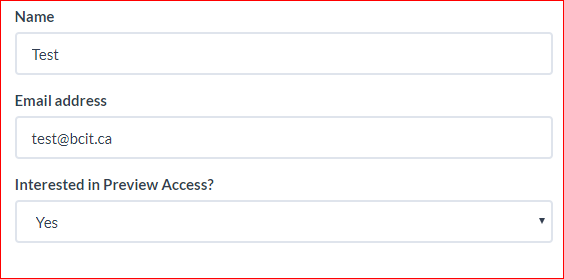


1. Now you are going to sign up for SNS messages. When the user submits data to the website, you will receive an email.
   1. From the AWS Console, click on Simple Notification Service (SNS) under the Application Integration.
   2. Click Subscriptions on the left.
   3. Click “Create Subscription.”
   4. On the next screen, select the ARN (Amazon Resource Name) with the word “NewSignupTopic” as part of the text (This will be a very long piece of text).

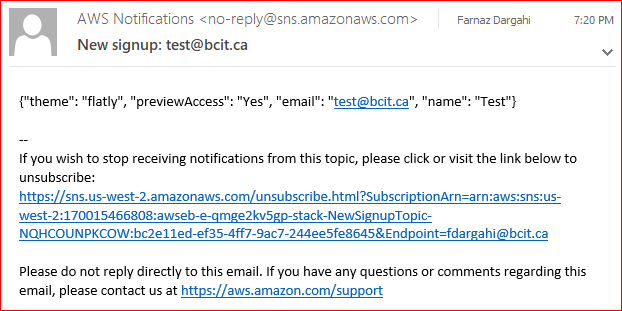


* 1. On the Protocol field, select Email.
  2. In the Endpoint field, enter your email address and Create Subscription.
  3. After your subscription is created, you must confirm it. Open the email and click the “Confirm Subscription” link.

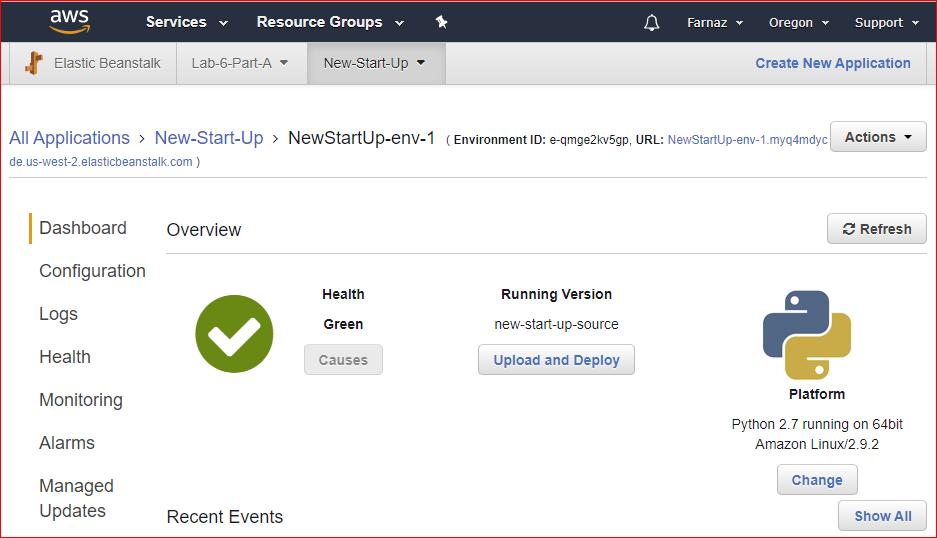
1. Refresh the browser tab containing the home page of your website.
2. Click the “Sign up today” button and enter the following test data:



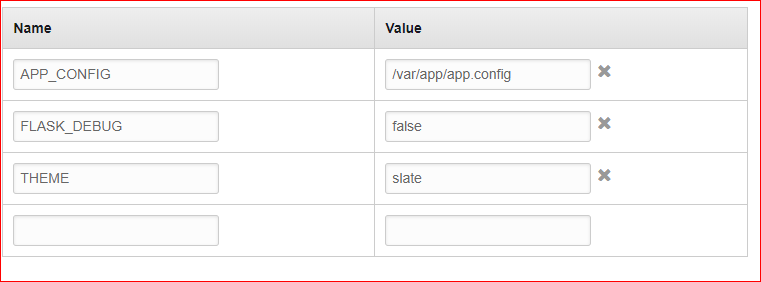
1. Check your email inbox. You should receive an email with the below content.



1. From the AWS Services menu, select DynamoDB under Database.
2. On the left side, select Tables. You should see the table that was created for this application.
3. Click on the hyperlink name of the table.
4. On the right side of your screen, click on the Items tab.
5. You should see the data that you entered from the website.
6. Go back to the web page and sign up another user. You must use a different email address! Verify the data was entered in DynamoDB.
7. Notice the value in the “theme” column. The value is “flatly.”
8. Now, we want to change the theme of the website’s display
9. Go back to the Elastic Beanstalk application page.
10. On the left side, click on “Configuration.”



1. Click on Modify button associated with the Software Configuration.
2. At the bottom of the Environment Properties area, change the theme from “flatly” to “slate.” Click Apply.



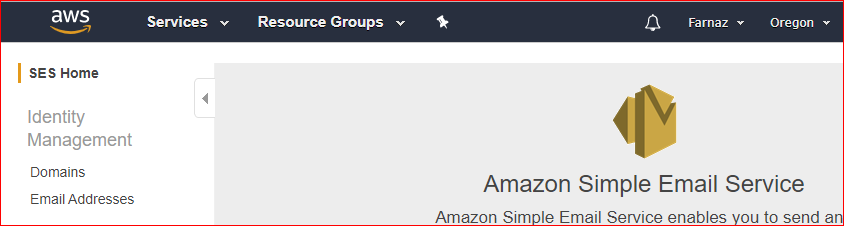
1. After the update is green again, refresh your website in your browser.
2. Keep your browser open and show your work as part B to your instructor when you finished all parts.
3. Watch the following instructional videos for the Startup Sample App on YouTube to understand the App & Architecture: <https://www.youtube.com/watch?v=DrRr-JgdgzE>

**Part C**

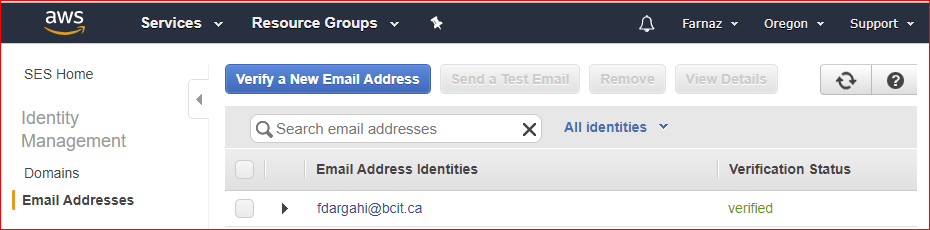
1. In this part we will create the worker tier as described in this video: <https://www.youtube.com/watch?v=jSVY-SVcCAM>

Following instructions are adopted from the above video.

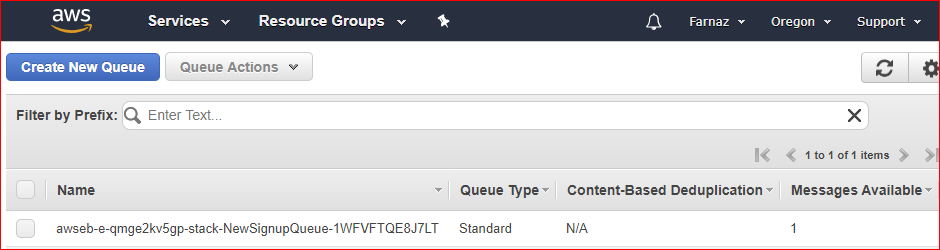
1. Worker tier handles background-processing tasks generated by your application.
2. In our application, background task is to send conformation email to the customers when they are signing up. For that, the sender email address, which is your email address, should be verified.
3. From AWS service menu, select simple email service (SES). On the left side choose email address.



1. Click on verify a new email address. Enter your email address. (Mybcit or Gmail, don’t use Hotmail)
2. Login to your email and confirm the email.
3. Now you should see your email address as a verified email address.



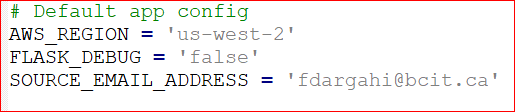
1. When a Customer signs up in our application, her information will be stored in database table and on a queue so we can send her confirmation email later on.
2. In AWS service page, select simple queue service. You will see a queue. Look at messages available column.



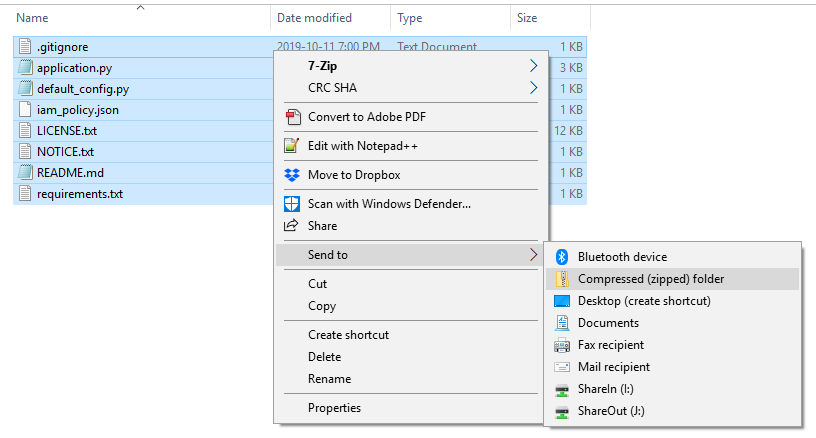
1. For me there is one available message and that shows the number of customers who have signed up. This message is waiting to be processed for sending a confirmation email to the customer.
2. Click on green download button to download the code for worker tier from: <https://github.com/awslabs/eb-py-flask-signup-worker>



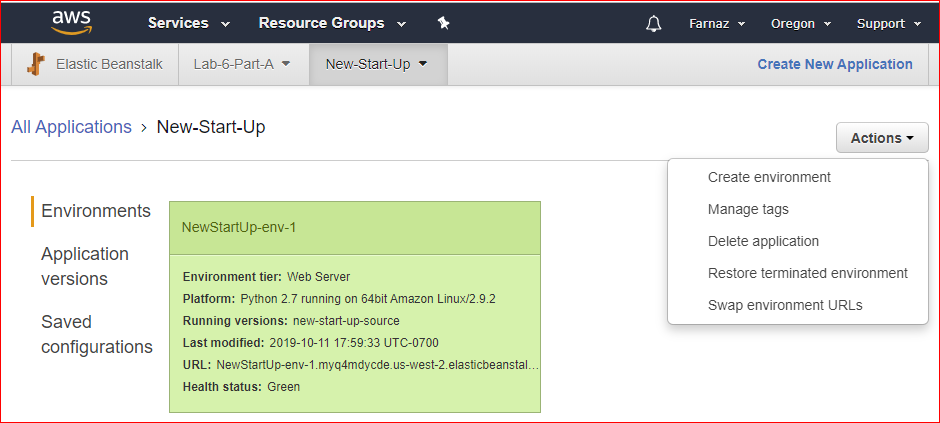
1. Unzip the file to change the default\_config.py file. Open default\_config.py with an editor. Change SOURCE\_EMAIL\_ADDRESS to the email that you just verified above (your email address) and save your changes.



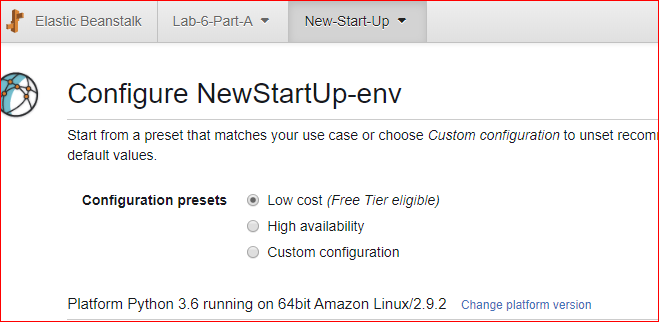
1. Select all files including modified default config fie and zip them up. (Zip files not folder). Name the zip files as worker.zip we need this file later to upload to EB. Save it in a place that you can easily find it.



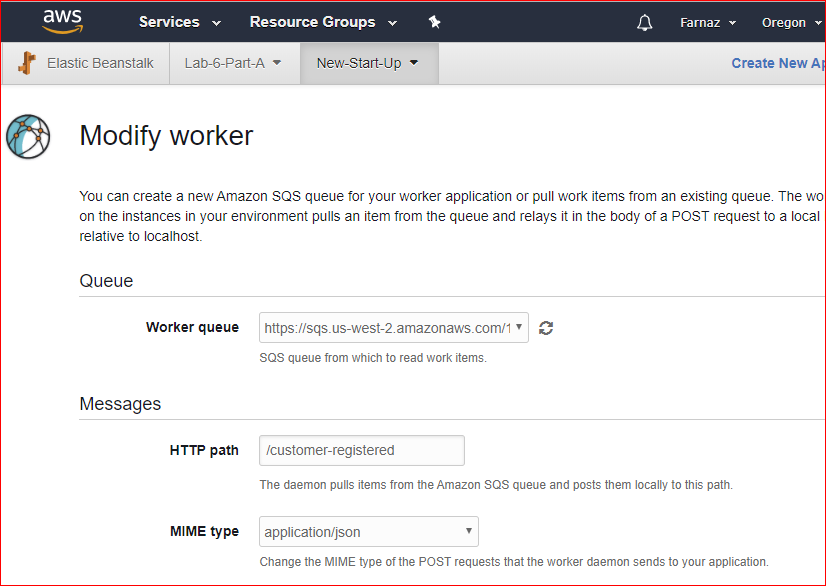
1. Go back to elastic beanstalk service. Up to now, you should have two applications (Lab-6-Part-A and New-Start-Up).We want to create an environment to deploy worker tire.
2. In your New Startup application, click on action and select create environment.



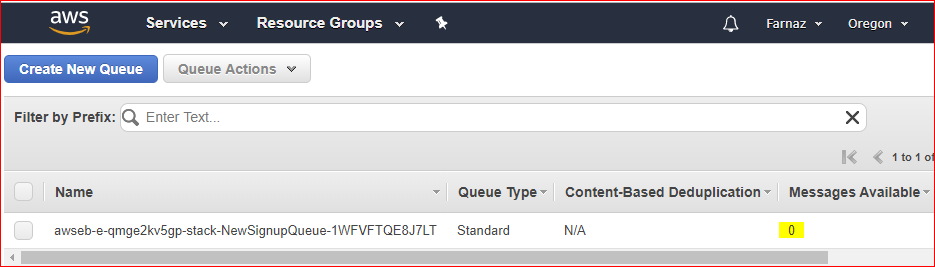
1. For environment tier select worker environment.
2. For preconfigured platform, select python
3. Upload your worker.zip folder created in above steps by clicking on upload button.
4. Click on configure more options.
5. Change platform version to python 2.7.



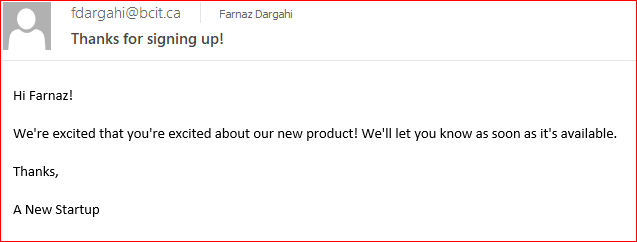
1. Click modify under worker window. For worker queue, select your application queue. Starts with sqs-us-west-2…
2. For the HTTP path enter /customer-registered. This path has been defined in application.py file in worker.zip.



1. Save your changes.
2. Click modify in security window, select your key pair for EC2 key pair, and save.
3. Wait around 4 minutes for environment to be green.
4. Do you remember that I had one available message in the queue? That message should have been processed now.



1. Go back to your application home page and signup yourself (the email you have verified).
2. You should receive an email like below:



1. To prevent fraud and abuse, AWS applies certain restrictions to SES:

You can only send mail **to** verified email addresses.

You can only send mail **from** verified email addresses.

In this link: <https://docs.aws.amazon.com/ses/latest/DeveloperGuide/request-production-access.html>

You can learn more about how to request and remove these restrictions from your account.

1. Show your work for part C to your instructor. Open your email address and show confirmation email with “Thanks for singing up!” title.

**Instructor’s initials: \_\_\_\_\_\_\_**

**Clean up**

After the instructor has checked, you work,

* Terminate all the Elastic Beanstalk environments using the Actions button on the Elastic Beanstalk dashboard page. This should terminate instances.
* Wait a few minutes and then check your EC2 instances. Is it still running? If it is, terminate the AMI used with the Elastic Beanstalk.
* Verify that the DynamoDB table has been deleted.
* In the IAM, delete the custom policy you created.
* Delete your SNS topic.

**Part D (Optional)**

1. Go to the website Getting Started Using Elastic Beanstalk:

<http://docs.aws.amazon.com/elasticbeanstalk/latest/dg/GettingStarted.html#GettingStarted.Walkthrough.EditConfig>

1. Go to “Step 3: Deploy a New Version of Your Application.”
2. From the list of 13 sample applications, select a technology you are familiar with. Download the zip file, unzip it, and examine the contents. Try to get an understanding of the various files and what they do.
3. Note: You do *not* have to deploy the application. Just see if you can understand the code. (If you choose “Java with Tomcat,” then launch it as a Tomcat application, not a Java application.)
4. Create a short report and explain what you have understand about code. Submit your work to Lab6 dropbox on D2L.