Hosting LinearAlgebraTutor project

Introduction

Over the course of this class we gained a broad understanding of managing systems using AWS. For our project, our group wanted learn more about one of the AWS services that we did not get a chance to explore in the class.

For this, we chose to use the Amazon's Elastic Beanstalk service. The reason for this is in the past while working on other projects we have seen Elastic Beanstalk recommended to host Web Applications. Elastic Beanstalk is a service used to to deploy and scale web applications all in a single environment.

Project goals

Our goal with the project was to gain experience with Amazon's Elastic Beanstalk. In order to do this, we chose to use a pre-existing project that needed to be hosted and deploy that using Elastic Beanstalk. The project that we chose is called LinearAlgebraTutor. It is a python-flask web application designed to teach linear algebra in an interactive manner. In order to verify that we successfully deployed LinearAlgebraTutor using Elastic Beanstalk, we have the following requirements:

Deploy LinearAlgebraTutor project on Elastic Beanstalk
- Project Accessible over the web
- Elastic Beanstalk configured correctly
    - Capacity Provisioning
    - Load Balancing
    - Application Health Monitor
    - Alarms configured for application

Possible effects on user population

The application we chose to deploy over Elastic Beanstalk, LinearAlgebraTutor, aid in students learning of Linear Algebra. By deploying this project over the web, it allows users with basic technical proficiency to access the application, and hopefully .

Possible user issues would be if the information within the application is incorrect. It would be unfortunate to use the power of the internet to bring incorrect information to people.

Results

We were able to deploy LinearAlgebraTutor to Amazon's Elastic Beanstalk (EBS), We ran into some issues initially because of how the project was initially configured, but once we got past those, we were able to upload it.The main reason there were issues was the fact that the application files were not in the root directory. After changing this the application was able to be uploaded to Elastic Beanstalk easily.

After the project was refactored so the files were in the root directory, Elastic Beanstalk proved to be a simple way to deploy web applications. After uploading and deploying the project via submitting a zipped version of the project, the Elastic Beanstalk service takes care of the rest. It will automatically provision the necessary capacity, and health monitors.

Load balancers were able to be set up easily in the configuration. We tried to test the load balancer's via spamming the application, however the application has minimal calculations and thus we were unable to generate enough workload for the load balancers to create another instance.

We added two alarms, one to monitor CPU Utilization and one to monitor the status of the instance.

Initially we planned on implementing a database to work with the application. However, LinearAlgebraTutor is a web application that has already been built and uses cookies. To refactor the code in the application to use a database would involve writing a lot more code and goes beyond the scope of this project. We also liked that when using cookies, a login is not required, so it keeps the web application simple. In the future, a database may be implemented so that the user can access their information on different machines.

Security
Possible security issues include Cross-Site Scripting attacks and Cross-Site Request Forgery attacks. Flask is set up so that it escapes html text to prevent XSS attacks, and with the nature of the application a CSRF attack at best could yield a change in the user's local point database stored in cookies.

Another security issue could be the fact that module completion data is stored in cookies. A user could gain local access to a machine and interact with the cookies. We do not believe this is a huge security risk because a malicious agent should not have local access to a machine.

Documentation
All the steps required to deploy the LinearAlgebraTutor project over Amazon's Elastic Beanstalk is described below:

1. Download a zip of the LinearAlgebraTutor project:
   https://mega.nz/#!TS50xSZY!wPvEyHlj7VKb2X6vpUemJdxZevoW-PQbvDRMJihrIW4
1. Proceed to the elastic beanstalk hompage:
   https://us-west-2.console.aws.amazon.com/elasticbeanstalk/home?region=us-west-2#/applications
2. On the right hand "actions" menu, select create an environment.
3. On the environment tier screen, select a web application environment
4. Select Python as the preconfigured platform

5. Scroll to the bottom and select the "upload your code" radio button
6. Select the button labelled "upload", and then select the LinearAlgebraTutor zipped file from your local machine.
7. When all these steps have completed, select "Create Environment" and Elastic Beanstalk will do the rest.
8. Optional: After the project has been uploaded, if you would like to set up load balancing then do the following:
   a. Select Configuration -> Capacity
   b. Change environment type from single environment to load balanced