SWE 645: Assignment 1

Application overview:

Your first assignment is to create your own class homepage and a Student Survey Form using HTML. You will use Amazon Web Services (AWS) cloud to host your Website and other submissions. More specifically you will use AWS Simple Storage Service (AWS S3) to create your homepage and AWS Elastic Compute Cloud (AWS EC2). The homepage should have a hyperlink to the survey page. The survey page allows prospective students to provide feedback about their campus visits. The following are more specific requirements of this application.

Requirements:

- Develop your class homepage using AWS S3 with a link to the student survey page described below. This homepage will be used to provide links to your future homework submissions.
- Develop a student survey form to allow a user to enter the survey data. In particular, the student survey form contains the following:
    a. Text boxes for first name, last name, street address, city, state, zip, telephone number, e-mail, and date of survey, which are required fields.
    b. Checkboxes that allow prospective students to indicate what they liked most about the campus. The checkboxes should include: students, location, campus, atmosphere, dorm rooms, and sports.
    c. Radio buttons that allow the prospective students to indicate how they became interested in the university. Options should include: friends, television, Internet, and other.
    d. A dropdown list of options for the user to select the likelihood of him/her recommending this school to other prospective students. The three options of the dropdown list are: Very Likely, Likely, Unlikely.
    e. An additional text box called Raffle. The user will be asked to enter at least ten comma separated numbers ranging from 1 through100 in the Raffle field. This information will be used to announce whether the student wins a free movie ticket.
    f. A text area for additional comments, and
    g. Submit and cancel buttons.

- Provision a free tier eligible Amazon EC2 instance using an Amazon Machine Image (AMI) that may already have Tomcat (or JBoss7) configured, depending on which server you prefer to work with. For example, once you are on EC2 Console, click Launch Instance link. For the AMI, search for key word "**Tomcat**" in **AWS Marketplace** and select the machine image labeled something like "Tomcat powered by Bitnami" – the one labelled as free tier eligible. In addition to Tomcat, this instance also hosts MySQL, which could be used for successive assignments, or use AWS Relational Database Service (AWS RDS) to provision RDS, but this is for later assignments. You should provision linux or window-based machine depending on your preference and comfort level.

- To get ready for the future assignment, create a dynamic web project using your favorite IDE such as, Eclipse or IntelliJ, on your local machine (laptop/desktop) and have the Student Survey form as part of this dynamic web project. Make sure you can deploy and access your application on your local Tomcat instance running on your local machine. Once done with your testing, create and deploy your war file on Tomcat (or jBoss) on EC2 instance, add the link of your application to your website, and provide URL of your homepage as well as of the application deployed on EC2 in readme file as part of your submission.
- Above are the core functional requirements. Since most of you have prior CSS, JavaScript background, feel free to use them and be creative.

Reference/Examples:

Here are a couple of references that could be used to create a static website on AWS S3 as well as an example how to provision and access EC2 machine in AWS cloud.

Setting Up a Static Website on AWS S3

http://docs.aws.amazon.com/AmazonS3/latest/dev/HostingWebsiteOnS3Setup.html

Amazon EC2 - Virtual Server Hosting

Please read **Appendix A below for an example** to provision a Linux based EC2 machine and access it from your Mac/Linux laptop.

You can read get more details on EC2 using the following link: http://aws.amazon.com/ec2/

Submission

The submission for this assignment should be through the blackboard website. I expect a zipped package containing the source files, war file, and any additional packages, scripts, or files that you used. I also require a readme file which contains installation and set up instructions so that the TA and myself can deploy and run the assignment. I also expect AWS URL of your homepage as part of readme file.

Submit **all source, and war file, files necessary to run the application** and the installation and execution instructions. Please put all of the files in a **zip** file. If you submit an assignment late (i.e., not submitted on due date) the late penalty will apply.

NOTE: A late assignment carries a 10% late penalty for each week it is late. Assignments are NOT accepted after being 2 weeks late.

Make sure your name is on every programming artifact so we know who it belongs to. For every source file, please include comments at the top of the program describing what the program does. This only needs to be 1 or 2 sentences.

Be sure to **test access and functionality** to your submission before the due date.

Grading:

The following areas will be used in the basic grading of these projects:

- Does system meet the functional requirements: 85 points
- Does the assignment run without errors: 13 points
- Comments: 2 points

Instant Point Deductions:

I reserve the right to deduct points instantly for the following reasons:

- The source, or binary, files are not included in the package.
- The readme file is not included in the package.
- The program doesn't run due to errors in the code.
- I spend more than 5 minutes trying to debug the assignment.
- I can't figure out how to use the assignment, and instructions are left out.

Appendix A:

Below is an example with detailed steps to provision and use Linux based EC2 instance and how to access it from your local Mac/Linux.

(**Please note** that if you are accessing Linux EC2 machine from your Window based laptop/desktop, you will need to use putty. You can find more details here: http://www.putty.org/ **Also**, if you have Window based EC2 machine and accessing it from your Window based laptop/desktop, you will need to use **RDP**)

1. Go to https://aws.amazon.com and open an account. You can use it for free for a year. But you have to enter credit card information though they don't charge you.

2. Go to EC2 Virtual Servers in the cloud.

3. Click on "Launch Instance"

4. Click on "AWS Marketplace", it is present on the left side of the page.

5. In the search tab, type "Tomcat Bitnami". You will get a list of images. Select the one appropriate for you. Remember to choose the

free tier eligible. It will contain ready to use versions of Apache, Tomcat and MySQL etc, which will be used in our assignments.

6. After selecting the particular image scroll down to the end of page and click "Review and Launch".

7. Then click "Launch". Once you click launch a window will pop up and ask for "Select an existing key pair or create a new key pair". Choose "Create a new key pair". Enter the name of your choice for the key pair. Then download and save the key pair without fail. Remember this is the only time when you can save the key pair to your computer. You will need this while connecting. Finally click on "Launch Instance"

8. Then on the next appeared page scroll down and click "View Instances". It will take some time to initialize and to start running.

Note: Once the instance is launched you can go to "Actions", click on "Instance State" and then choose to Stop, Start, Reboot and Terminate options for your instance.

10. To connect to your instance. Select the instance and then click on "Connect"

11. Choose "I would like to connect with A standalone SSH client" (Your Linux computer most likely includes an SSH client by default. You can check for an SSH client by typing ssh at the command line. If your computer doesn't recognize the command, the OpenSSH project provides a free implementation of the full suite of SSH tools. For more information, see http://www.openssh.org.)

11. Open the SSH client.

12. Locate your private key pair (key pair you downloaded and saved). Change directories to the location of the private key pair file that you created when you launched the instance.

13. Use the chmod command to make sure your private key file isn't publicly viewable. For example, if the name of your private key file is my-key-pair.pem, you would use the following command

```
chmod 400 my-key-pair.pem
```

14. Use the following ssh command to connect to the instance. You'll specify the private key (.pem) file and user_name@public_dns_name. For Amazon Linux, the user name is ec2-user. For RHEL5, the user name is either root or ec2-user. For Ubuntu, the user name is ubuntu. For Fedora, the user name is either fedora or ec2-user. For SUSE Linux, the user name is either root or ec2-user. Otherwise, if ec2-user and root don't work, check with your AMI provider.

```
ssh -i my-key-pair.pem ec2-user@ec2-198-51-100-
1.compute1.amazonaws.com
```

Note: You will get the example of your ssh command in the "Connect to your Instance" window when you clicked "Connect". You can copy & paste the same command. As it will have your credentials (key pair name, user name and IP address).

15. You'll see a response like the following.

```
The authenticity of host 'ec2-198-51-100-1.compute-1.amazonaws.com
(10.254.142.33)' can't be established. RSA key fingerprint is
1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f.
Are you sure you want to continue connecting (yes/no)?
```

16. Enter yes

You'll see a response like the following.

```
Warning: Permanently added 'ec2-198-51-100-1.compute-1.amazonaws.com'
(RSA) to the list of known hosts.
```

17. For transferring Files to Linux Instances from Linux Using SCP, use following command. Most Linux, Unix, and Apple computers include an SCP client by default. If yours doesn't, the OpenSSH project provides a free implementation of the full suite of SSH tools, including an SCP client. For more information, go to http://www.openssh.org.

```
    scp -i /path/my-key-pair.pem ~/path of your war file ec2user@ec2-
198-51-100-1.compute-1.amazonaws.com:~
```

This will copy your war file in the ec2-user home directory. To deploy your applications on EC2 you are required to keep your war file in apache-tomcat's webapps folder. Locate this folder and keep the file there.

19. You'll see a response like the following.

```
The authenticity of host 'ec2-198-51-100-1.compute-1.amazonaws.com
(10.254.142.33)' can't be established.
RSA key fingerprint is
1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f.
Are you sure you want to continue connecting (yes/no)?
```

20. Enter yes

21. You'll see a response like the following.

```
Warning: Permanently added 'ec2-198-51-100-1.compute-1.amazonaws.com'
(RSA) to the list of known hosts.
Sending file modes: C0644 20 SampleFile.txt
Sink: C0644 20 SampleFile.txt
SampleFile.txt  100%   20     0.0KB/s   00:00
```

Note : If you receive a "bash: scp: command not found" error, you must first install scp on your Linux instance. For some operating systems, this is located in the openssh-clients package. For Amazon Linux variants, such as the Amazon ECS-optimized AMI, use the following command to install scp.

```
[ec2-user ~]$ sudo yum install -y openssh-clients
```

22. In the browser copy paste the Public DNS path of your instance. You will get the home page of Tomcat Bitnami.

23. Now just add the name of your homework folder to the path and you can see your homework homepage.

Example: http://ec2-52-89-94-175.us-west-2.compute.amazonaws.com/HW1/

24. Stop the instance once you are done and logout from the AWS Amazon account.