AWS ELASTIC BEANSTALK PROJCET DESCRIPTION

Student Name	Adnan Branković
Project URL	http://cloudapp-env.eba-9kaweitz.us-east-2.elasticbeanstalk.com/
Scenario	I imagined my application as site for movies reviews (comments, opinions, grades etc.) which is visited by different visitors who want to do a research on various movies from different types of geners before watching or purchasing them. This application is deployed in US East 2 region and is used by visitors from Ohio, while web application is developed by using Python version 3.6 and Django version 2.1.1. The website is mainly used during non working hours and weekends when most people have free time and want to relax with a good movie after a tough working day and working week. During the work hours, web site has average 500 visitors every hour while between 18:00h and 21:00h, number of visitors increase to 1500. Between 21:00h and 00:00h, is the busiest period and number reaches to 2500. Therefore, during working hours there should be only 1 instance to serve people after working hours. During non working hours, to ensure the best user experience and limit cost, it is scheduled to serve 3 instances and expand to even 4 if necessary.

ENVIRONMENT CONFIGURATION DETAILS

	Parameter	Value	Reasoning
1	AWS X-Ray	Disabled	This application is not complex enough to warrant the use of AWS X-Ray for debugging.
2	S3 log storage	Disabled	This application is using CludWatch, so there is no need for S3 log storage.
3	Monitoring interval	5 minutes	Application monitoring metrics should be reported to CloudWatch every 5 minutes.
4	Root volume	Container	Since this application does not depend on any specific storage hardware or the amount of storage space, elasticbeanstalk will create a default root volume.
5	Environment type	Load balanced	Application's capacity should increase dynamically by adding new instances, so the load balanced environment is required.
6	Instances	Min 1 Max 4	If each instance can handle up to 750 concurrent users, app would need at least 1 application instance to support minimum traffic (500 users), and 4 instances to support peak traffic (2500 users).
7	Instance type	t2.micro	The website is not too much resource-intensive, a "micro" instance type should be enough to handle its operations.
8	Availability Zones	Any, us-east-2	The application is intended for US users, US-East, the application was deployed in the closest Availability Zone.
9	Scaling cooldown	60 seconds	60 seconds should be enough for the initialization actions to complete, before starting new scaling activities.
10	Metric	NetworkOut	Average network throughput for transmitted traffic.
11	Period between metric evaluation	5 minutes	It is used for monitoring.

12	Breach duration	5 minutes	If it exceeds 5 minutes either there are many requests or the instance is unstable and it is time to scale up, and vice versa.
13	Upper threshold	6000000 Bytes	The Auto Scaling in Elastic Beanstalk environment uses two Amazon CloudWatch alarms to trigger scaling operations. The default triggers scale when the average outbound network traffic from each instance is higher than 6 MB or lower than 2 MB over a period of five minutes.
14	Lower threshold	2000000 Bytes	
15	Scale up increment	1	Add EC2 Instances.
16	Scale down instances	-1	Remove EC2 instances.
17	Time-based scaling	None	We will not be utilizing time-based scaling as app is not advanced and does not need it.
18	Health reporting	Enhanced	Enhanced health reporting provides free real-time application and operating system monitoring of the instances and other resources in our environment.
19	Retention and Lifecycle	7 days, keep logs after terminating environment	Health logs should be kept for 7 days in order to have an insight into the environment's health on a weekly basis. Also, we want to keep logs of terminated environments for the purposes of analytics and anomaly detection.
20	Max execution time (seconds)	60s	The maximum time a script is allowed to run before the environment terminates it. This helps prevent poorly written scripts from tying up the server.