IDC – Cloud Computing
Exercise 2
Due Date 16/06/2019

Spring Semester 2019
Mr. Dan Amiga / Mr. Oren Eini

Submission

- If you don't submit before the submission date you will be graded zero.
- Make sure you do not submit or publish your AWS access keys outside of your laptop.
- You will place a zip file in S3 with public access. And fill the following form with the URL and student details. https://forms.gle/s7tR4dMFJXQMxtki9
- The zip file will include the following two sub folders and a single summary file.
    - Script/Code folder (document your code when needed)
    - Screenshots of your AWS environment (with your user name on the top)
    - Summary & notes as described the question below

Expectations

- I understand that some of the topics here were not covered in detail in class. However, we did cover the fundamentals and important aspects in the exercise. You need to read and learn to complete this.
- There is a lot of online documentation to help you with your first steps; don't hesitate to search online and look in AWS documentation.
- This is not an exercise to conduct in couples, **only by yourself.** You are however allowed to advise with and get help from your peers.

Setup - Setting up your AWS account

1. Create an AWS account (normal, not a student one).
2. Use the coupon code in the course website to get AWS credits
3. Recommended: Setup billing alerts to make sure that you aren't using more resources than your credits. This is unlikely, but good practice nonetheless.

Parking lot management.

Our service needs to handle charging users for parking in various car lots worldwide.

Each parking lot has an IoT device that will generate a REST call to a specified endpoint with the following parameters whenever a car enter / exit the parking lot:

/notify?parkingLotId=<PARKING LOT ID>&status=<STATUS>&plate=<CAR LICENSE PLATE>

The PARKING LOT ID is a string designating the specified parking lot.

The STATUS can be either "exit" or "enter".

The CAR LICENSE PLATE is the string of the car license plate.

- Build a serverless function that would accept such REST calls and store the data into DynamoDB. You need to record the data for the duration of each parking session in each lot based on the input from the parking lot IoT devices.
- You also need to provide a way for the parking lot management to get a weekly report for how many hours each car parked in a specific lot.
- Users need to be able to pull monthly report of how many hours they parked overall (possibly over more than a single parking lot).
- Use Dynamo Streams to compute the charges for each user and trigger a request to http://charges.examples.com/charge?plate=<LICENSE PLATE> whenever the total amount owed per car is over 50$.

Pay attention to data modeling concerns in this exercise and include documentation explaining your reasoning for modeling the data the way you did. Take into account high scaling needs, such as deployments with MANY parking lots, cars.

In the attached documentation, discuss how you handle concurrency and consistency inside your solution.