Final Project
Social Media Monitoring via Logic Apps

Problem Statement:
Social media applications have become part of our everyday lives. This is even more true for the younger generations. With smart phones becoming the next thing in the hands of a child shortly after they put down their baby bottles, parents are frequently left blind to the messages being sent by their children. Parents need a way to monitor the usage of these messaging applications to ensure appropriate and safe usage. Although there are commercial applications that can perform these tasks, they typically require software to be installed on the device being monitored. The goal of this project is to capture messages from these applications and load them into a central location where the information can be viewed by a parent.

> "When past generations of parents let their children socialize, the setting was in-person and it was often easy to know what kids were doing. In fact, everyone kids knew either lived on their block or went to their school. Today, the internet and social media have evolved into the main social gathering spot for teens and kids. Forget the mall; Snapchat and Instagram provide the same social gratification as any real-life interaction. The problem? Who knows what's being sent, said…or saved."
>
> ~TeenSafe.com
>
> https://www.teensafe.com/blog/parents-monitor-childrens-social-media

Overview of the Technology:
This project will use Azure Logic Apps to extract messages from social media applications. Logic App triggers will be defined to execute at regular intervals to retrieve messages from Instagram, Facebook, and Twitter via API connectors defined within the Logic App workflow. Message attributes will be inserted into tables in an Azure SQL database. A second Azure Logic app will be scheduled daily that will call an Azure Function App to aggregate the message data and send a notification email stating the data is ready. A Python script will then connect to the Azure SQL database and render a bar chart showing the daily message distribution.

High Level Steps:
1) Create Azure SQL database and the message and summary tables.
2) Create Logic App with three API connectors: Instagram, Facebook, Twitter.
3) Create Function App to connect to Azure SQL database and aggregate the message data.
4) Define second Logic App to call Function App and send notification of completion via email.
5) Utilize Python client script to connect to Azure SQL database and generate trend graph.

Data Source: Data is loaded directly from Instagram, Facebook, and Twitter via the Logic App.

Software Used:
Python 2.7.14 (https://www.python.org/downloads/)
MatPlotLib 2.1.2 (https://matplotlib.org/)
Visual Studio Code 1.18 (https://code.visualstudio.com/)
Microsoft SQL Server Management Studio 2017 (https://docs.microsoft.com/en-us/sql/ssms/sql-server-management-studio-ssms)

Lessons Learned: Logic Apps is a very flexible architecture capable of delivering complex workflows with minimal coding required.

YouTube Links:
2 Min: https://youtu.be/Nq21domeXjc
15 Min: https://youtu.be/Hf0IBo7nDko

GIT Repository:
https://github.com/scmprouty/AzureDeepFinal