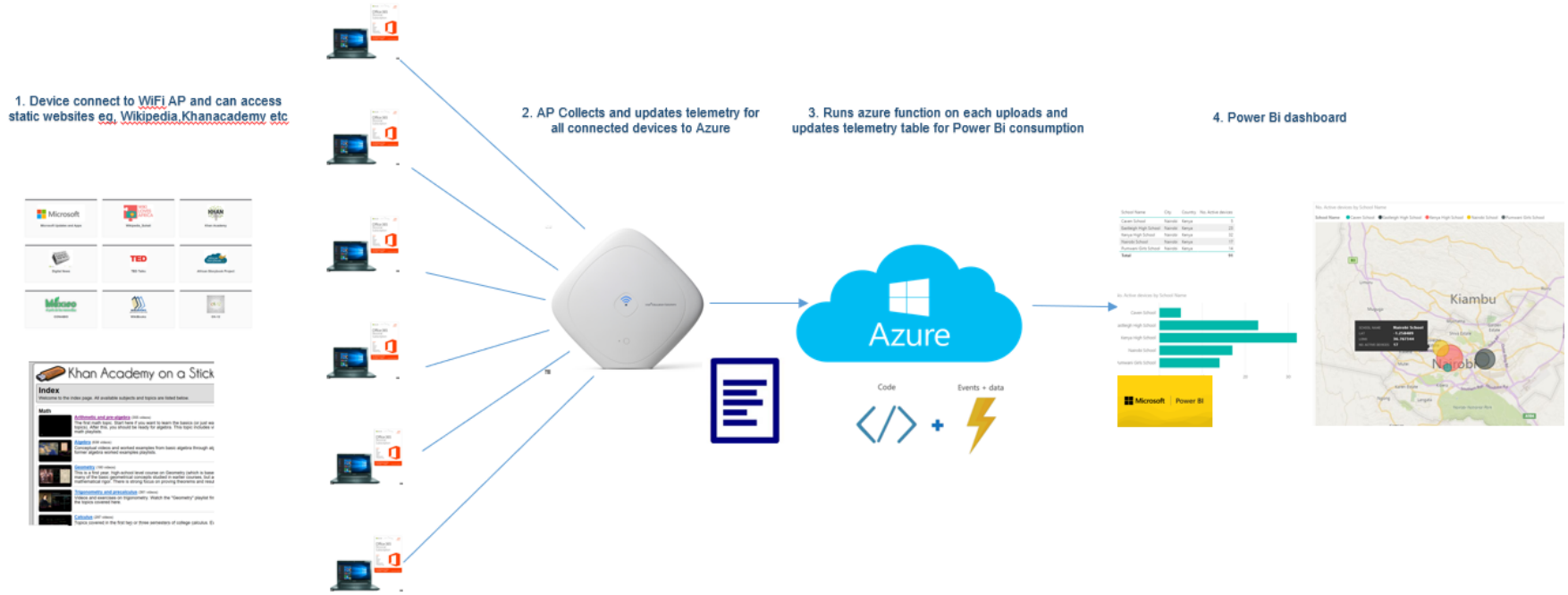
Objective::

We wish to provide tracking and usage information for all the devices that get connected to access point,

How we achieved this:



We setup a script on access point that collect information on all connected devices to an access point, This information is uploaded by each access point to Azure cloud storage, Every time a new file is uploaded the azure function triggers and updates tracking table, PowerBi portal uses updated table to display information in graphical way.

DETAILS:

Prerequisite

- Azure command line capability installed on access point
  - https://docs.microsoft.com/en-us/cli/azure/install-azure-cli?view=azure-cli-latest

Steps

1. Generate a log file that creates all the devices connected to access point periodically in this case we are calling it every minute using cron command below

   ## crontab -l

   */1 * * * * /root/iot-edge/MyIoTfileupload/Filegenration/mytest/getdevices.sh

   ---Details here ---

   www.adminschoice.com/crontab-quick-reference

   ## Content of getdevices.sh

```
cd /root/iot-edge/MyIoTfileupload/Filegenration/mytest
if cmp -s ./finallog.txt ./backuplog.txt
then
        echo "No new device detected no action needed"
else
        echo "new Device Connection detected Calling script to upload to cloud"
        ./mystorage_upload.sh
fi
# Coping current file to backup as we will generate a new file here
cp ./finallog.txt ./backuplog.txt

# Getting all device connected to AP and storing in log.txt file
arp-scan --interface=wlan0 --localnet |grep "192" >./log.txt

# Comparing the current device connected to last connected devices and creating fianllog.txt with all unique entries
sort ./log.txt  ./backuplog.txt | uniq >./finallog.txt

cd /
```

2. Above step will generate a finallog.txt that need to uploaded and has information on all the devices connected in past
3. Upload finallog.txt to azure blob whenever a new device is detected or this can also be done on periodic basis we are doing it as soon as we get a new device connected as per code in getdevices.sh

   ## Content of mystorage_upload.sh

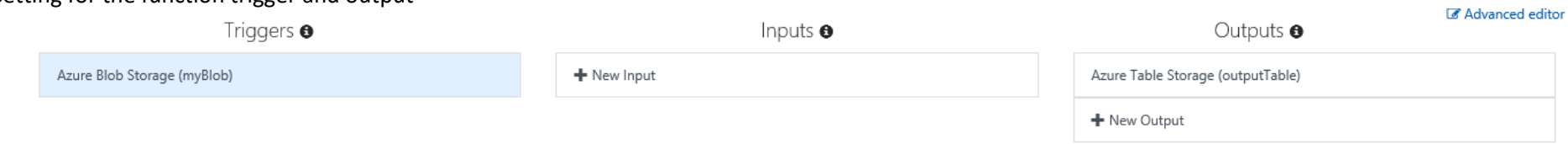
```
#!/bin/bash
# A simple Azure Storage example to upload a file
export AZURE_STORAGE_ACCOUNT=<replace with your storage account name>
export AZURE_STORAGE_ACCESS_KEY= <replace with your access key>
export container_name= <Replace with your blob container name>
export file_to_upload=~/iot-edge/MyIoTfileupload/Filegenration/mytest/finallog.txt
azure storage blob upload --file $file_to_upload --container $container_name -q
azure storage blob list --container $container_name
```

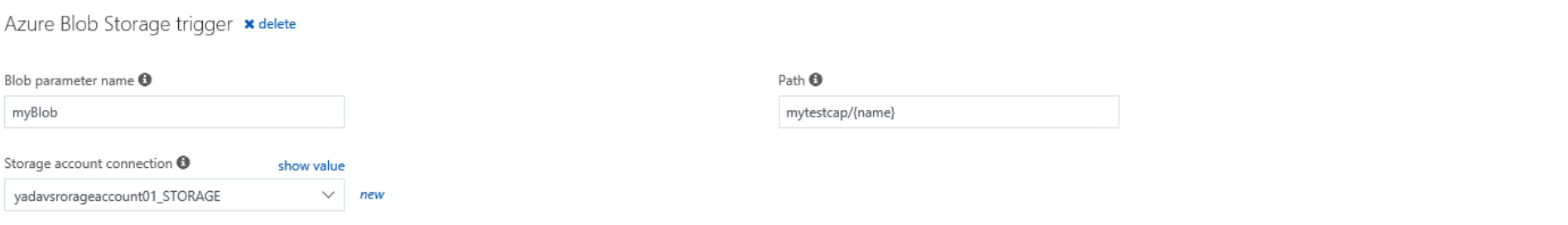
4. Trigger a function to read the fianllog.txt on cloud and generate a table as below

Azure function that reads telemetry finallog.txt and update a table name DeviceDataCap with relevant information

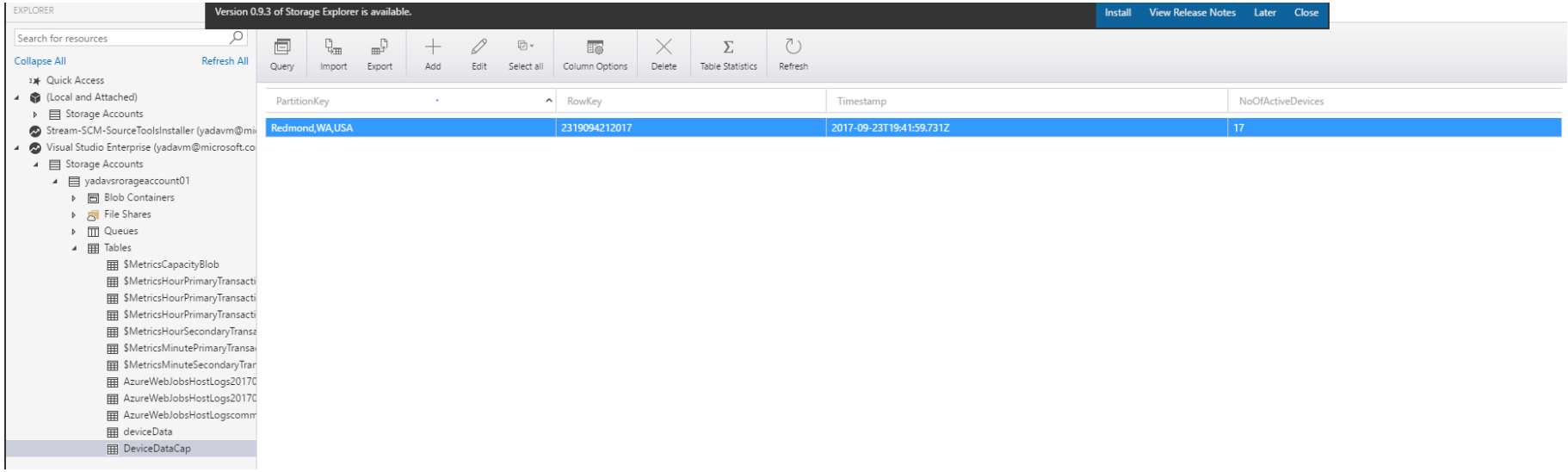
```
public static void Run(Stream myBlob, string name, TraceWriter log, ICollector<AccessPoint> output
//public static void Run(Stream myBlob, string name, TraceWriter log)
{
    log.Info($"C# Blob trigger function Processed blob\n Name:{name} \n Size: {myBlob.Length} Byte
       int count =0;
       //DateTime today = DateTime.UtcNow.Date;
        DateTime _now = DateTime.Now;
        string _dd = _now.ToString("dd"); //
        string _mm = _now.ToString("MM");
        string _yy = _now.ToString("yyyy");
        string _hh = _now.Hour.ToString();
        string _min = _now.Minute.ToString();
        string _ss = _now.Second.ToString();

        string _uniqueId= _dd+ _hh+ _mm+_min+_ss + _yy;
           log.Info(_uniqueId);
        using (StreamReader reader = new StreamReader(myBlob))
        {
            while(!reader.EndOfStream)
            {

                log.Info(reader.ReadLine());
                count++;
            }
            log.Info("No fo connected devices are :: "+ count);
        }
        outputTable.Add(
            new AccessPoint(){
                //PartitionKey = ""+ Math.Floor(int(DateTime.Today)/24*60*60*1000),
                PartitionKey = "Redmond,WA,USA" ,
                RowKey = _uniqueId,
                NoOfActiveDevices = count}
        );
        log.Info("!!!!!!UPdated tracking table !!!!!!!");
}

public class AccessPoint
{
    public string PartitionKey { get; set; }
    public string RowKey { get; set; }
    public int NoOfActiveDevices {get; set;}
}
```

Setting for the function trigger and output

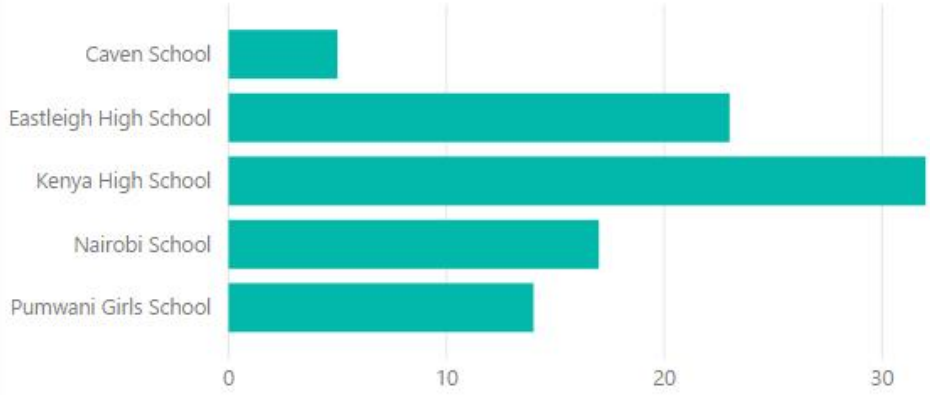
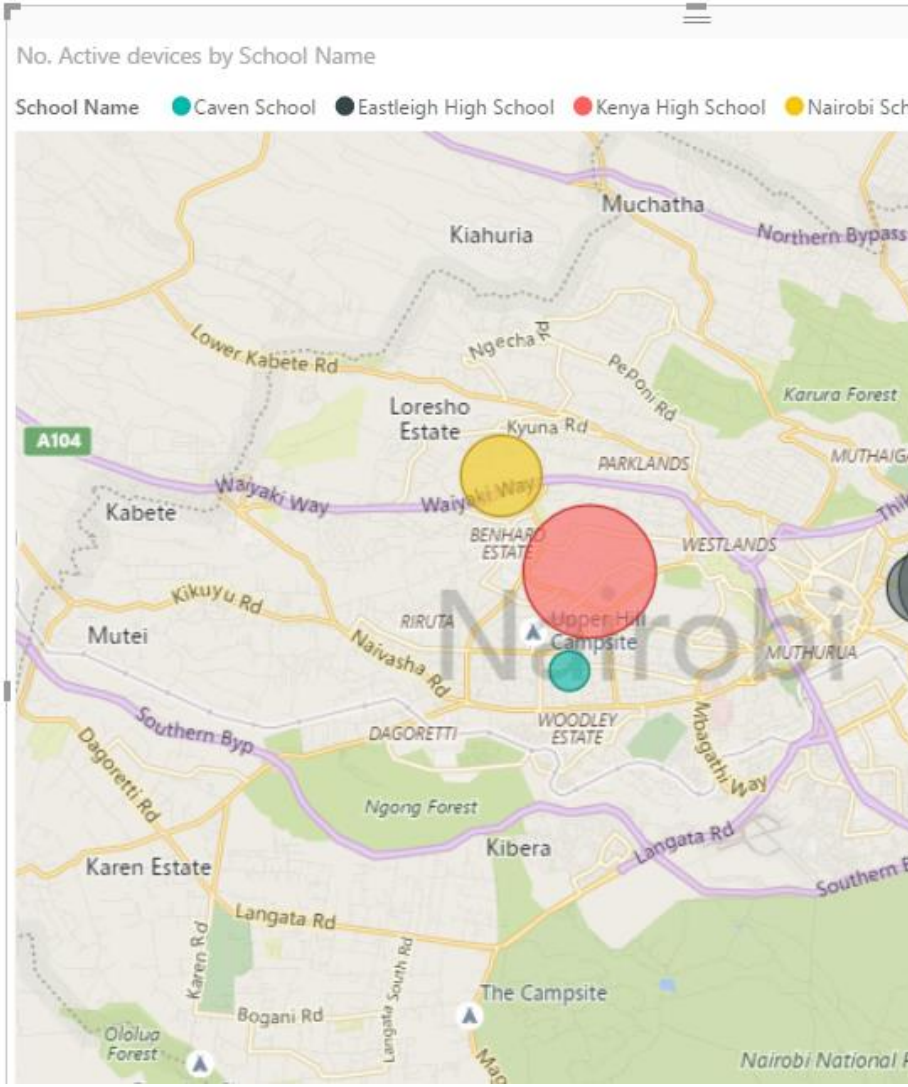




After function we get a table named DeviceDataCap



The above table can be used further by Power Bi portal to present the data graphical on dash board as

School Name	City	Country	No. Active devices
Caven School	Nairobi	Kenya	5
Eastleigh High School	Nairobi	Kenya	23
Kenya High School	Nairobi	Kenya	32
Nairobi School	Nairobi	Kenya	17
Pumwani Girls School	Nairobi	Kenya	14
Total			**91**

No. Active devices by School Name