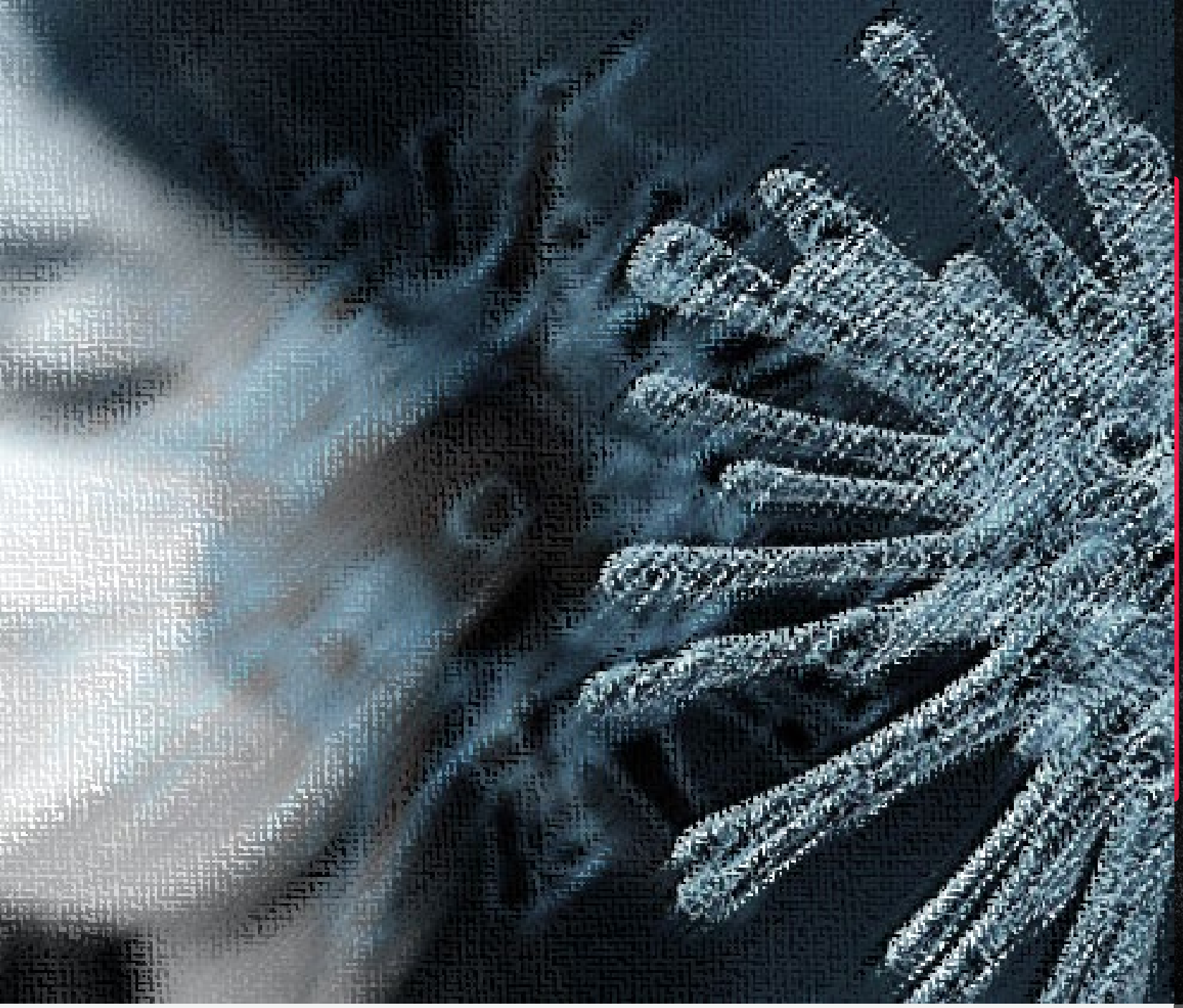


Combating COVID-19 Outbreaks

WQD7005 Data Mining
Master of Data Science | University of Malaya

Part C: Processing of Data Assignment

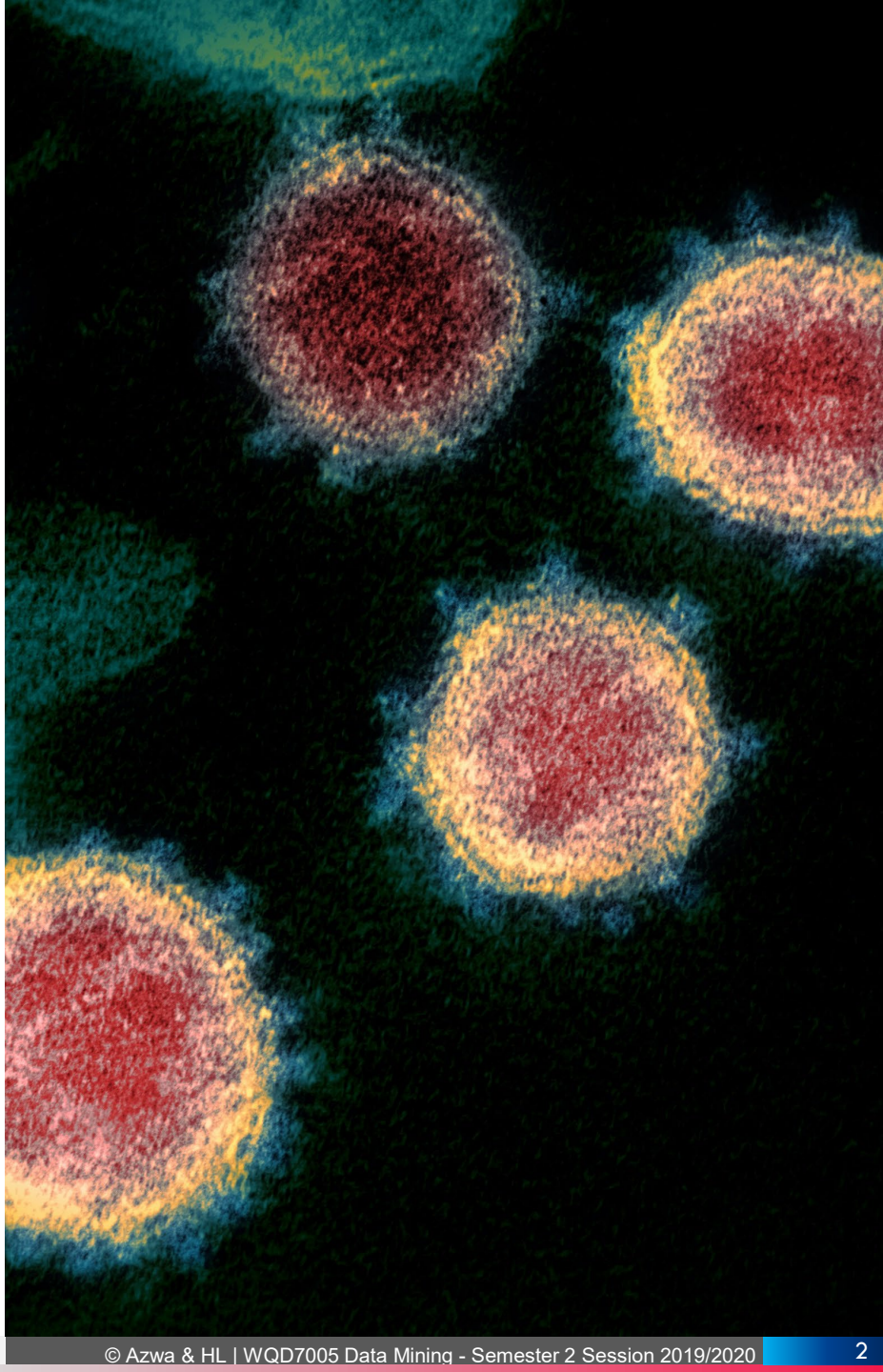
Group Members:
Azwa bin Kamaruddin (WQD170089)
Kok Hon Loong (WQD170086)

UNIVERSITY OF MALAYA

Assignment Milestones

WQD7005 Data Mining | Semester 2 Session 2019/2020

- Part A: (Group)
  - Web Crawling of Real-time Data
- Part B: (Group)
  - Management of Data using Hadoop Data Warehouse or Data Lake
- **Part C: (Group)**
  - Accessing and Processing of Data from Hadoop Data Warehouse or Data Lake using Python
- Part D: (Individual)
  - Interpretation and Communication of Data Insights
- Part E: (Group)
  - Deployment of the Data Mining Results on Web (Flask) and Mobile Application (Kivy)



Assignment Background

From the previous assignment milestones, our group have accomplished the following tasks:

I. We have demonstrated our method in acquiring the near real-time data using the web crawling approach using Python programming and BeautifulSoup package.

II. Illustrating the steps in storing the data using cloud storage (a.k.a. data lake) and also to the Hadoop data warehouse leveraging on the DataProc configuration and implementation using Google Cloud Platform (GCP), which the data was then being accessed using Hive.

For this assignment milestone, our group will be focusing on the following:

- Using **Python** that is coded on **Jupyter Notebook** web application in accessing the varieties of the stored data types (i.e. CSV, JSON) from the cloud storage.

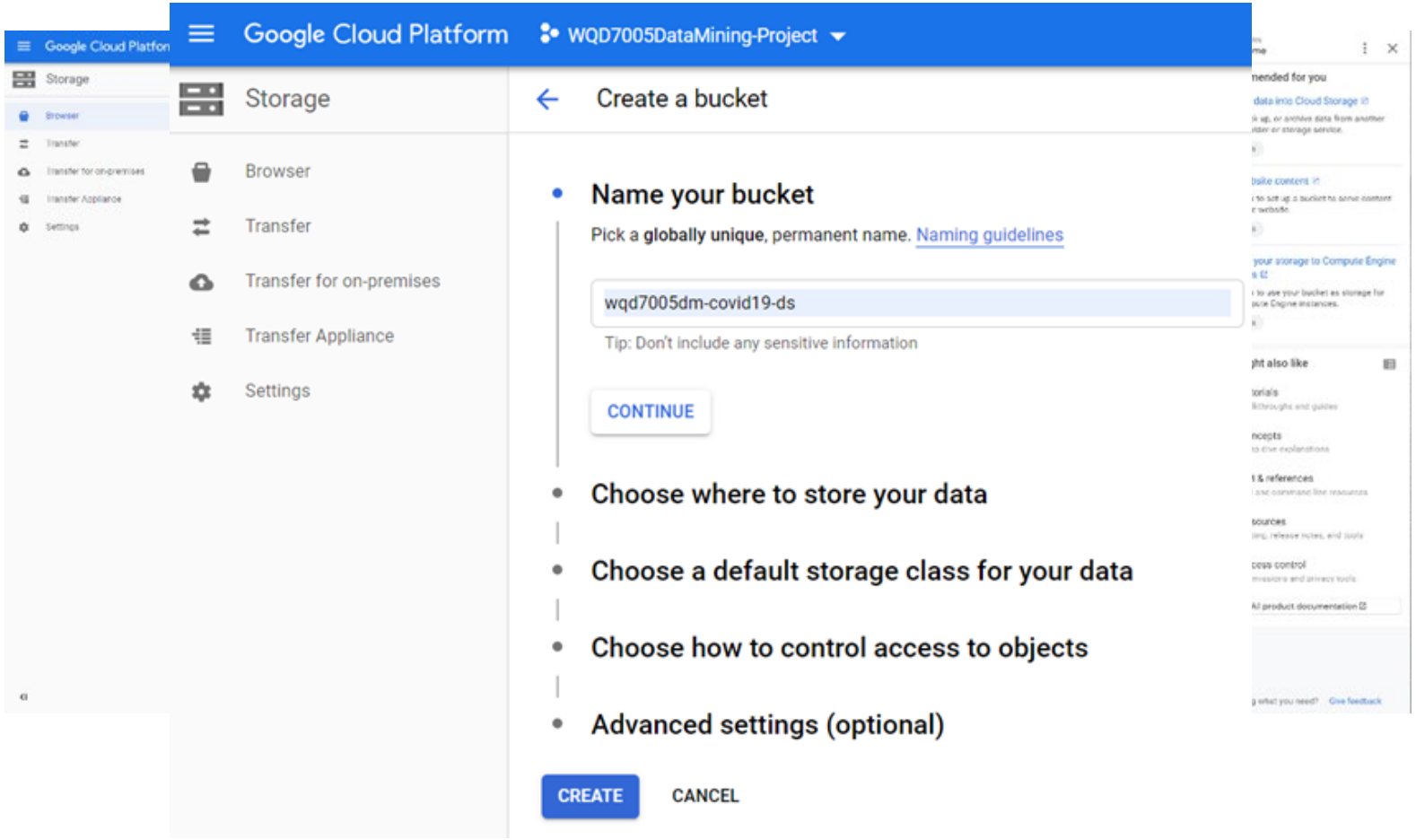
Our group Python code is also uploaded in our group assignment GitHub at the link below:

- https://github.com/scholarazwa/wqd7005-assignment

UNIVERSITY OF MALAYA

Accessing and Processing the Datasets

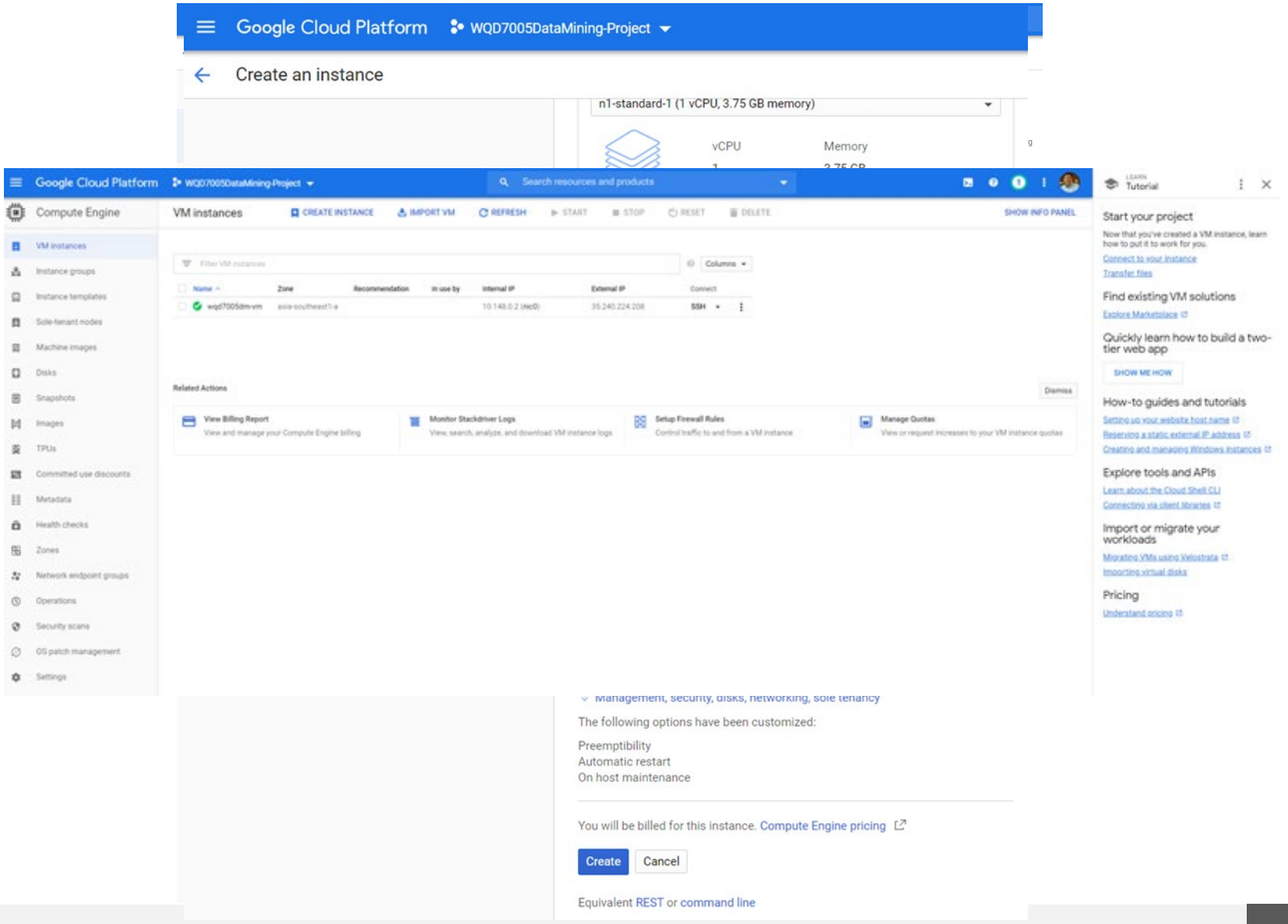
creating Cloud Storage (Data Lake) in Google Cloud Platform (GCP)



- Scroll to the **Storage** section in GCP **Navigation menu**, then click on **Browser**.
- Click on **Create Bucket**, to create the storage container to store the data captured from web crawling.
- Name the storage container as **wqd7005dm-covid19-ds**.

Accessing and Processing the Datasets

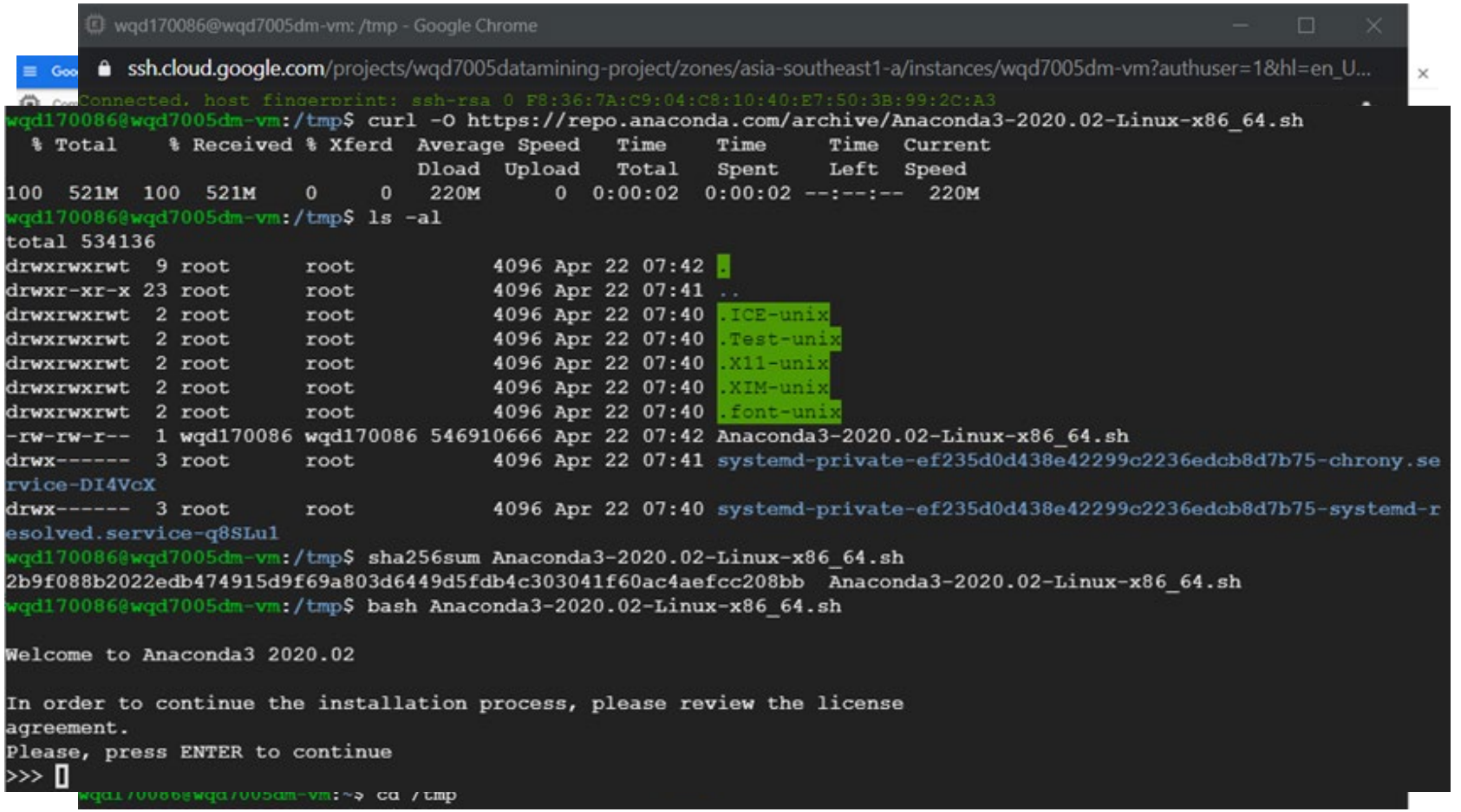
creating Virtual Machine Instances in Google Cloud Platform (GCP)



- In the **Compute** section, select **Compute Engine** and then click on **VM instances**.
- Click on the **Create** button in the new pop-up window.
- Use the minimal settings and named the VM instances as **wqd7005dm-vm**.
- Our group chose to use **Ubuntu** as the OS with **minimal setting**. Then click on the **Create** button to provision the **VM instances** to install **Anaconda** with **Jupyter Lab** for **Python**.

Accessing and Processing the Datasets

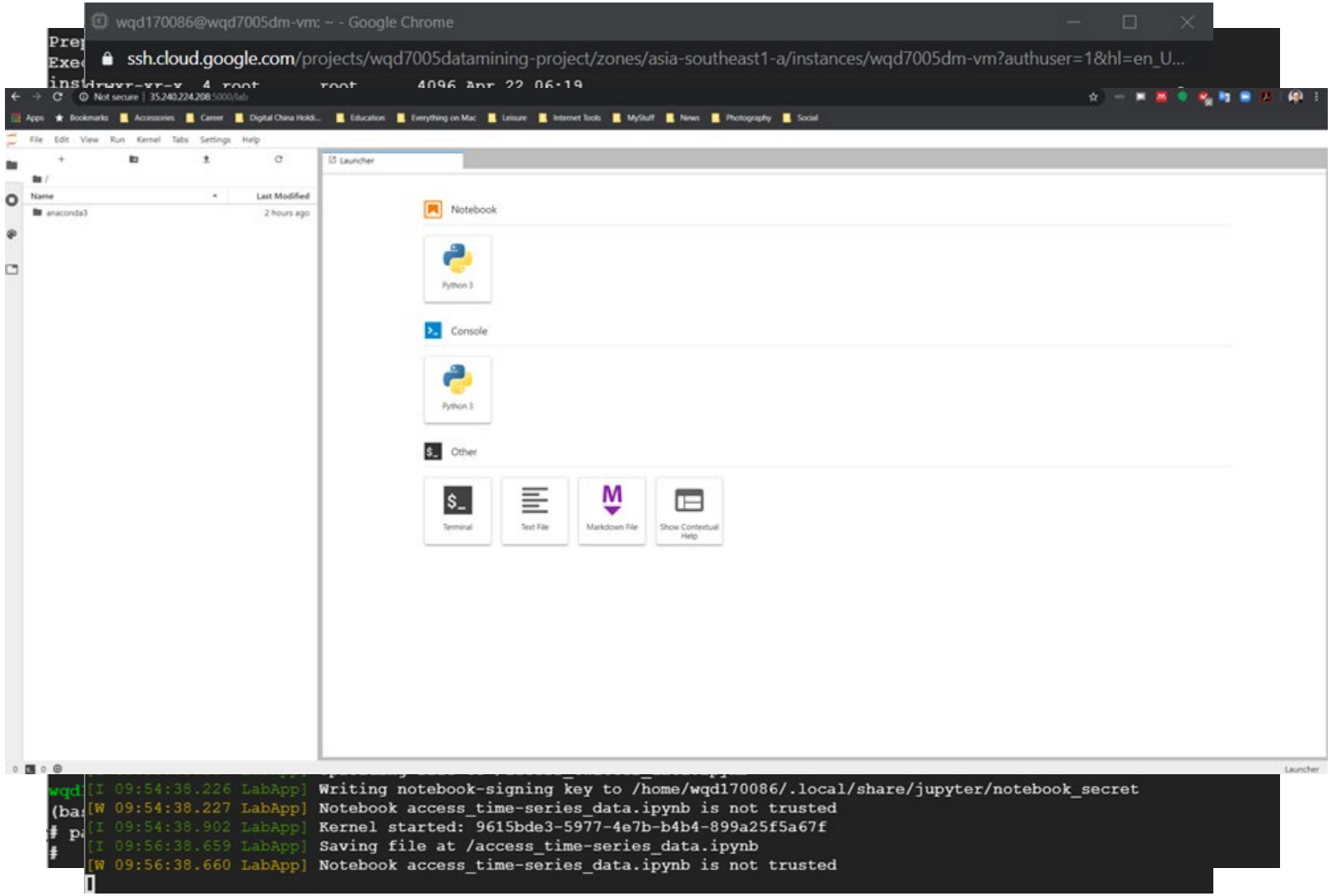
installing Anaconda and Jupyter Lab in Google Cloud Platform (GCP) VM instances



- Click on the **SSH** pull down menu, select **Open in browser window**.
- The **shell command prompt** running on Ubuntu Linux will appear.
- To install Anaconda, use the web browser to connect and view the Anaconda Distribution site @ *https://www.anaconda.com/distribution/*
- Choose the latest Linux OS version and from the command prompt, type the following command to download the basch script: *curl -O https://repo.anaconda.com/archive/Anaconda3-2020.02-Linux-x86_64.sh*
- Use the **SHA-256** ***checksum*** command before running the **Anaconda** ***installation bash script***.

Accessing and Processing the Datasets

to run Jupyter Lab in Google Cloud Platform (GCP) to start coding Python

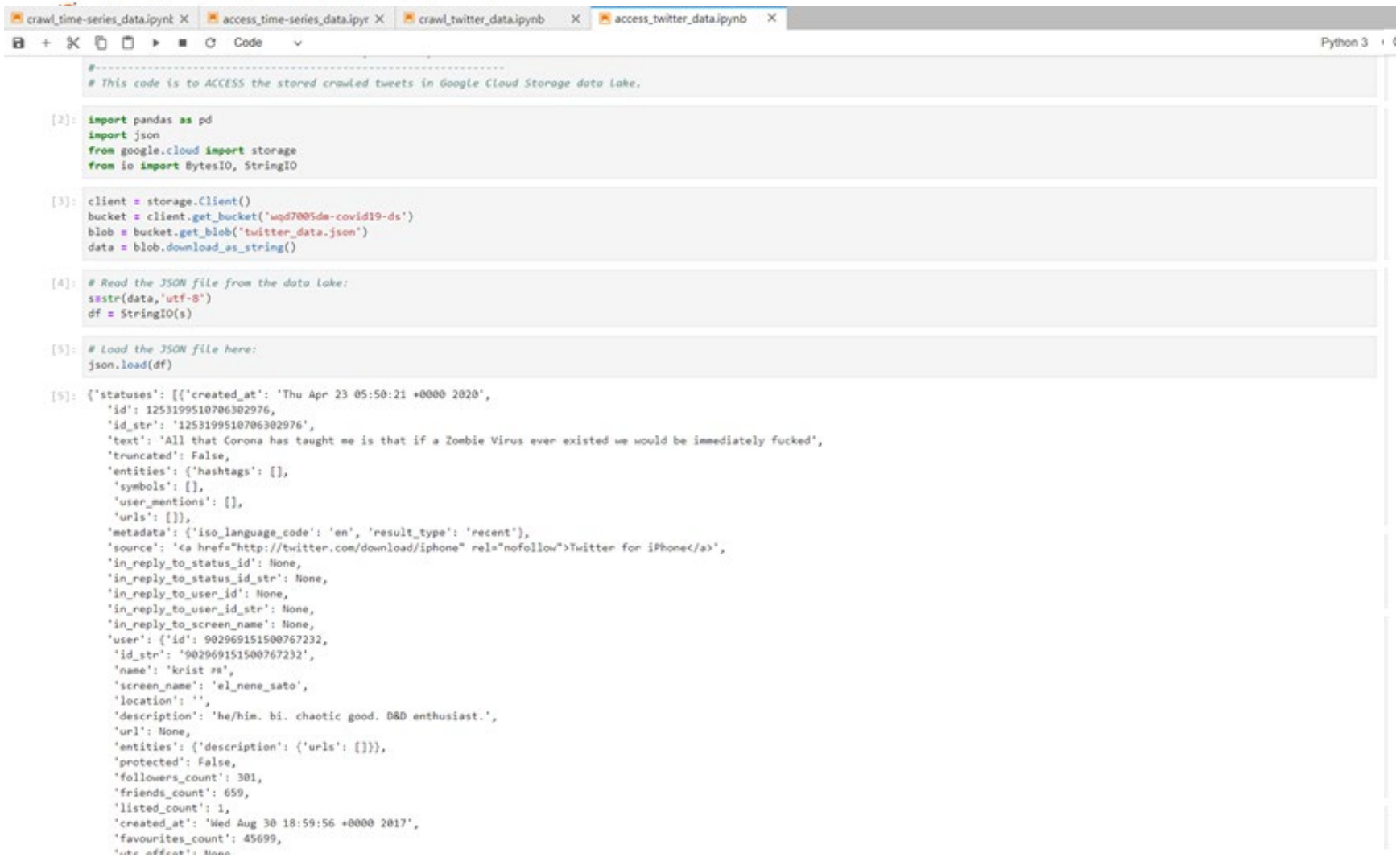


- To activate the installation, type **source ~/.bashrc**.
- To test the installation and activation, type **conda list**.
- Ensure to install the following packages if this is the **first time running** Anaconda and Jupyter Lab
  - To support importing storage in GCP: ***pip install --upgrade google-cloud-storage***
  - To support importing Twitter module: ***pip install tweepy***
- To start Jupyter Lab from the Ubuntu command prompt (based on the **port #** set up during configuration), type ***jupyter-lab --ip=0.0.0.0 --port=5000 --no-browser***
- To access to the Jupyter Lab notebook, use a web browser to key in the token generated from the above command and the static IP address configures, i.e. ***http://<Static IP addr>:<Port #>/?token=f9823a614febd8fb5997ab9bb7d98536db463d4de8000c4d***

Note: *The token to use is different each time the jupyter lab is executed.*

Accessing and Processing the Datasets

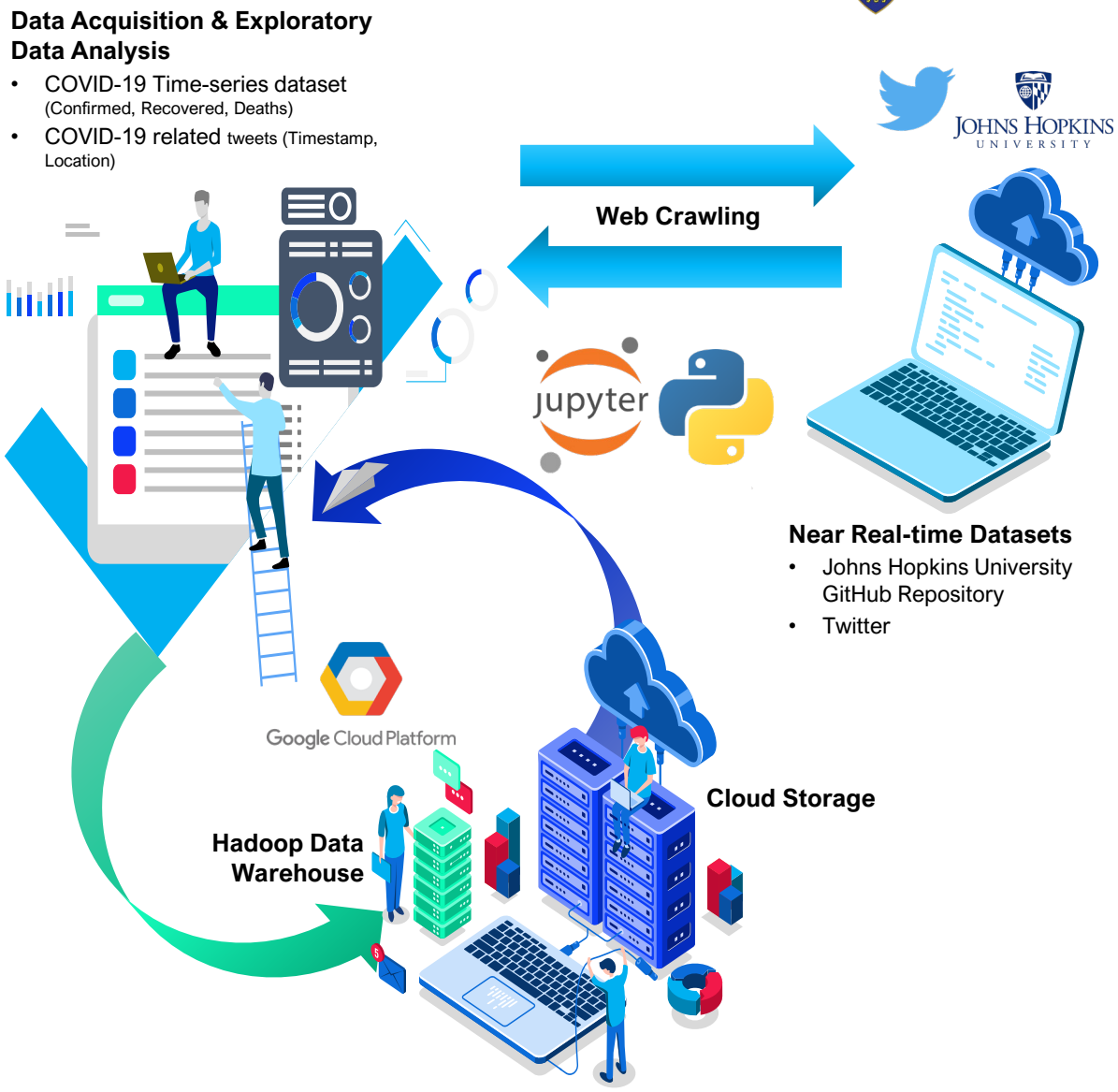
using Python on Jupyter Lab notebook in Google Cloud Platform (GCP) VM instances



- In this assignment, our group have created 4 sets of Python programming codes:
  a. Crawl for time-series data (***csv*** data sets)
  b. Access time-series data
  c. Crawl for Tweets from Twitter on relevant topics (***json*** data set)
  d. Access Tweets data
- For this assignment, we have demonstrated the following:
  I. Using web crawling method with Python codes to acquire varieties of data sets;
  II. Accessing the data sets acquired using Python programming from our Cloud Storage (i.e. Data Lake)

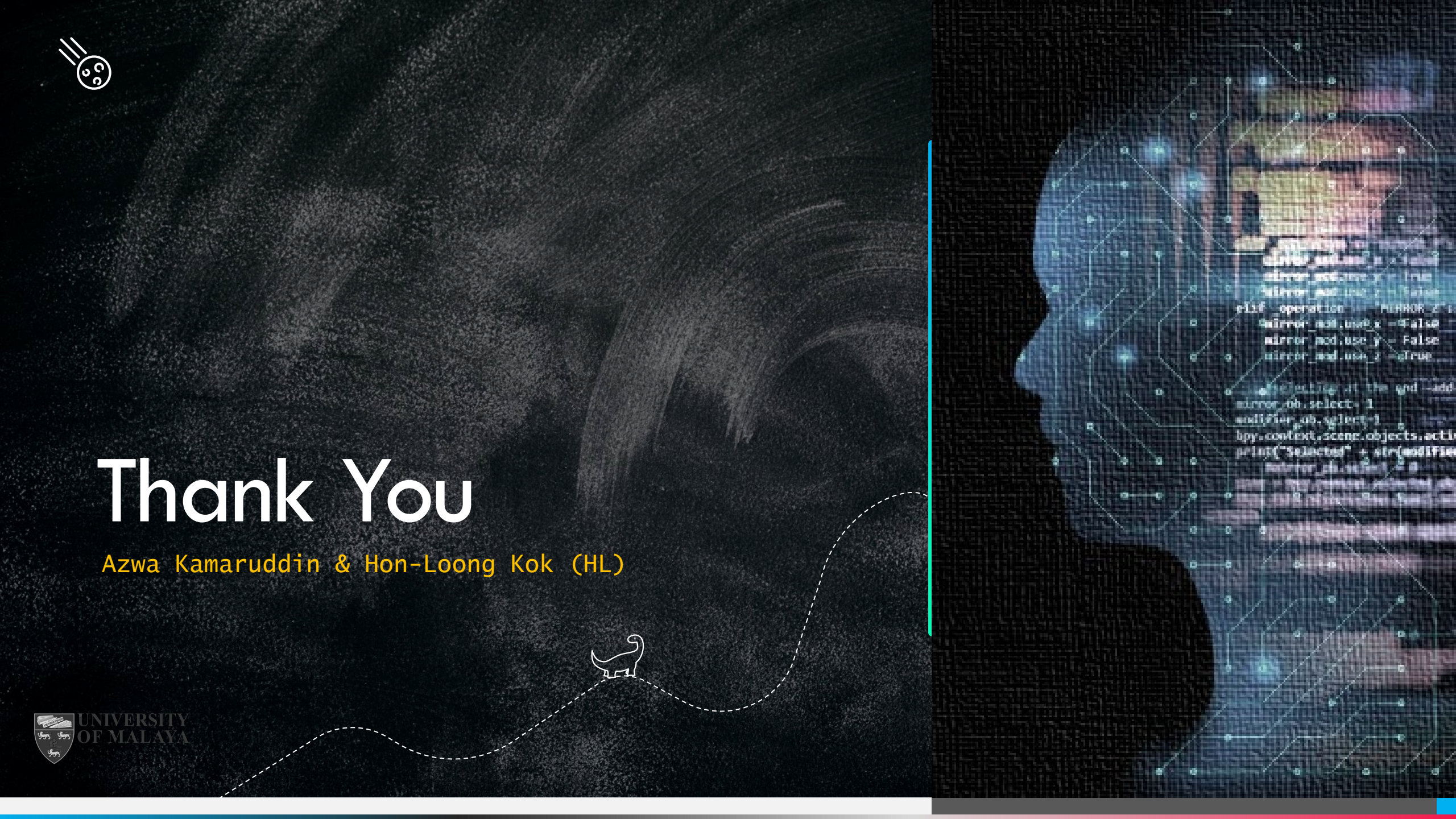
Summary

- Within a typical data mining project, there are usually many different types of operations management software with many different data storage as well.
- As the number of data sources multiplies, having data scattered all over in various formats prevents the data analysts from seeing the full and clear picture of their current state.
- This creates the necessity for integrating data in a unified storage system where data is collected, reformatted, and ready for use.



Thank You

Azwa Kamaruddin & Hon-Loong Kok (HL)



UNIVERSITY OF MALAYA