School of Information Technology and Electrical Engineering
INFS3208 – Cloud Computing

Project (20 Marks)

Due at 1 PM 21/10/2022 (Friday in Week 12)

Overview and objectives

The **goal of this project** is to *propose a cloud-based application in a propo*sal **(5 marks)** and *implement the application* **(15 marks)** using learned cloud computing technologies. You could choose one of two types of projects below and find a specific topic suitable for the type. Some project ideas will be demonstrated in the video recordings of the previous student projects that were the winners of the competition last year. A novel project with excellent implementations will be nominated for the student project competition awards this year. In this open project, the **understanding of cloud computing's concepts and the ability to use the respective cloud technologies** will be assessed. As an individual project, you are required to complete both the proposal and implementation all by yourself. The proposal submission will be automatically checked against the existing documents including academic publications, books, and articles on the web in the Turnitin databases. The case with a HIGH similarity score will be sent to Academic Ethics and Integrity Committee for further investigation. The implementation will be marked by tutors in person in the practical and tutorial sessions. You must submit the proposal and the source code on the blackboard before 1:00 PM 21/10/2022. It is your responsibility to ensure the submission is on time.

Proposal (5 Marks)

The proposal aims to test your understanding of the concepts, characteristics, and relevant technologies in Cloud Computing. Moreover, your ability to design a cloud-based application with a reasonable budget will be assessed. The project proposal must NOT exceed two A4 pages and it should include the following sections:

- Introduction:
    - Give background information about this project: *What is this project about?*
    - Explain the motivation of this project: *Why is this project important?*
    - Describe the overall objective and features of the project: *What features does this project have?*
    - Explain the limitations of traditional computing solutions: *Why doesn't traditional computing solve the problem well?*
    - Explain the benefits brought by cloud computing: *How does cloud computing fit in this project?*

- Technical Solutions:
    - Describe what cloud technologies you've used in this project (e.g., k8s for the front-end app, NoSQL for the back-end data storage).
    - Provide a monthly cost estimation of all the cloud resources used in this project (e.g., costs of VMs, K8s cluster, networking, Load Balancers, etc.).

- Architecture Design:
    - Depict the workflow or framework of the project in a figure (e.g., micro-service framework).

The teaching team will follow up on your implementation to ensure that the project is feasible.

To broadly accommodate students with different backgrounds, the projects are categorised into two types:

Type I: a web-based application using Micro-service architecture and related technologies.

- e.g., PHP/JavaScript + MySQL/NoSQL + Docker/K8s.
- *This project type is focusing on developing and deploying scalable, reliable, and resilient applications in the cloud*.
- Scalability and resiliency of the application should be demonstrated.

Type II: a data-analytic application using computing models for Big Data.

- e.g., Jupyter Notebook + Scala/Python/Java + Spark SQL / Streaming / MLlib
- *This project type is focusing on solving a complicated real-world problem by analysing a large amount of data.*
- The real-world problem should be solved by analysing the large-scale data.

You can only choose one type and find out the specific topic that suits the type according to your background and preference. There are a few technical requirements that must be met when designing your project in the proposal.

Technical requirements:

If your project belongs to **Type I**,

- For the **front-end** design, you should have a functional user interface allowing users to interact with the application (e.g., login, search, etc.).
- To support the overall objectives of this project, you should have a **back-end** design (e.g., database) working together with the front-end UI.
- You should use docker technologies to **containerise** your applications with multiple containers running in a **microservice architecture**.
- To make your application scalable, reliable, and resilient, you should use either **Docker Swarm** or **Kubernetes** to orchestrate multiple containers across hosts for your application.

If your project belongs to **Type II**,

- You need to perform a **challenging** analytic problem that requires large-scale data processing (e.g., big data queries, classification, regression, clustering, association rule mining over big data). You need to pick up a **large dataset in the real world.** Some publicly big datasets are recommended below. To persist the data, you need to **store** the data in either database (relational/non-relational databases, e.g., MySQL/Redis/MongoDB/Cassandra) or distributed file system (e.g., HDFS).
- To avoid “Garbage in Garbage out”, you need **data pre-processing** (e.g., removing/replacing NaN values, converting data types, imputation, etc.) to ensure the data quality for the downstream analytical task.
- You should run the analytical task on a Spark cluster using **Spark programming** techniques and the related **Spark built-in libraries** (i.e., SQL, MLlib, Streaming, and GraphX). You can choose either Python or Scala as the programming language.
- To demonstrate the analysis results, your program should **visualise** results with some tools (e.g., matplotlib in Python) in Jupyter/Zeppelin Notebook.

No matter which types your project belongs to, the cloud infrastructure costs should be reasonable. You can use Google Cloud Platform Pricing Calculator when making the budget plan in the proposal.

Implementation (15 Marks)

The implementation aims to test your ability to build and deliver a cloud application that is proposed in your project proposal. The implementation should be technically consistent with the proposed features and functionalities in the proposal. You are ONLY allowed to use your own code in the previous programming/analytical courses. For example, your previous web development project in INFS3202 can be re-used but you need to containerise the application and deploy it on the cloud. If the previous project was a group one, it's your responsibility to have the team members' consent and you need to clearly inform the team members that the workload of the individual project in INFS3208 is to deploy the application in the cloud with the taught technologies. To demonstrate the implementation, you need to present it to your tutor in the tutorial and practical sessions in Week 13. To avoid the absence of your presentation, you could alternatively prepare a **10-minute video presentation** of your project. The presentation should include but not be limited to the following parts:

- Background & Motivation;
- Project architecture and applied technologies;
- For Type I projects, database design (for database users) or data storage (for file system users);
- For Type II projects, descriptions of the used data and analytic models (Spark programming);
- Results & Discussion

You MUST back up the implementation code and data regularly (at least every week). It is your responsibility to ensure that you fully understand the technical requirements of this individual project. If you have any concerns or problems, please contact your session's tutor or ask questions on Ed Discussion.

You should use the GCP credit wisely and avoid all unnecessary expenses on cloud services during the project. Also, you should monitor the balance of the credits during the implementation. Please contact your tutor as soon as possible when the credit is running out (less than $10). **Note that the GCP credit is not unlimited. It is strongly recommended to develop, test, and debug locally before deploying the project in the cloud.**

Submission

You should make an online submission before 1 PM 21/10/2022 (Friday in Week 12):

- Your proposal is no more than 1,000 words in two pages.
- The source code and the relevant data should be compressed into a single file and should be submitted to Blackboard.
- If the data is too large for uploading, you should provide an external link (e.g. OneDrive or Dropbox link) in the submission and ensure the data source is valid.

You are welcome to discuss your proposal and implementation with the teaching team during tutorial/practical sessions or on Ed Discussion.

Publicly available datasets:

1. Queensland Government Open Data Portal, https://www.data.qld.gov.au/
2. Datasets on Kaggle, https://www.kaggle.com/datasets
3. Free Public Data Sets for Analysis on Tableau, https://www.tableau.com/learn/articles/free-public-data-sets
4. Awesome Public Dataset list on Github, https://github.com/awesomedata/awesome-public-datasets

Note: some small datasets should NOT be used.