Project (35 Marks)

Due at 4:59 PM 16/10/2020 (Friday in Week 10)

Overview and objectives

The goal of this assignment is to propose a cloud-based application in a proposal **(15 marks)** and implement the application **(20 marks)** with cloud computing technologies and related programming skills. You will need to propose the project, implement the code, and present the work all by yourself. You feel free to choose any application you are interested in and utilise the right cloud computing technologies to carry it out. It's no problem to re-recreate something that you've done before, but you need to re-implement or deploy the application with learned cloud technologies. A novel project idea with excellent implementations will be nominated to the student project competition awards. Overall, your project will be open-ended, and your project should demonstrate how to make use of cloud computing technologies and what specific benefits your cloud application can get from cloud computing. You are required to individually complete the proposal and implementation. You must online submit the proposal on the blackboard before 4:59 PM 16/10/2020. It is your responsibility to ensure the submission on time.

Proposal (15 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 application will be assessed. Turn in a printed two-page project proposal that is including but not limited to 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 objectives: *What features do this project have?*
    - Explain the benefits brought by cloud computing: *How does the cloud computing fit in this project?*

- Technical Solutions:
    - Describe what front-end technologies you use in this project.
    - Describe what back-end technologies you use in this project.
    - Estimate the cloud cost (e.g. VM, cluster network, software, etc.).

- Architecture Design:
    - Depict the workflow or framework of the project in a figure.
    - Describe in detail the architecture of the project.

The teaching team will follow up with your implementation to ensure that the project is reasonably proposed with appropriate size and difficulty.

According to the previous experience, students' projects are normally categorised into two types:
1. Type I: a web-based application (e.g. PHP/JavaScript + MySQL/NoSQL + Docker) that is scalable, reliable, and resilient in the cloud.

2. Type II: a data analytic application (e.g. Jupyter Notebook + Scala/Python/Java + Spark SQL / Streaming / MLlib) that is capable of analysing big data.

You can choose either one 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 front-end UI smoothly. 3
- You should adopt a **micro-service architecture** and **containerise** your micro-services with multiple containers. You should use **docker-compose** to orchestrate all the containers.
- In your design, your application should have **scalability** and **reliability** with the help of cloud technologies (e.g. docker swarm mode and load balancer).

If your project belongs to **Type II**,
- You need to perform a **challenging** data analytic task (e.g. big data queries, classification/regression, clustering/association rule mining, etc.) to solve a real problem.
- You do not have to analyse the real big data in this task, but the dataset you pick up should be **reasonably large and complex**. To store the data, you need to **distribute** the data in either multiple relational databases (e.g. MySQL/MariaDB) or multiple non-relational databases (e.g. Redis/MongoDB/Cassandra).
- You should run this analytical task on a Spark cluster using **Spark programming** techniques and related **Spark built-in libraries** (i.e. SQL, MLlib, Streaming, and GraphX). You can choose either Python or Scala as the programming language.
- To return the analysis results, your program should be able to **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 fair and reasonable. You can use Google Cloud Platform Pricing Calculator when making the budget plan in the proposal.

Implementation (20 Marks)

The implementation aims to test your ability to build and deliver a cloud application that is proposed in your proposal. The implementation should be technically consistent with the proposed features and functionalities in the proposal. It's no problem to re-recreate something that you've done before, but you need to re-implement or deploy the application with learned cloud technologies. Please note that simply re-using the code is not allowed. If the previous project was a group one, you need to acknowledge your previous team members that this project will be re-implemented with new cloud technologies and they must confirm their consent. To demonstrate your project, you should present it to your tutor in the tutorial and practical sessions in Week 11. To avoid the absence of your presentation, you should prepare a **10-minute video presentation** of your group project. The presentation should include but not limited to the following parts:
- Background & Motivation;
- Project architecture and applied technologies;
- For Type I projects, database design (for database users) and Data storage (for file users);
- For Type II projects, descriptions of analytic models and applied data in Spark programming;
- GUI demonstration or result visualisation;
- Comparing to non-cloud development and deployment, what advantages and disadvantages does cloud computing have in your project.

You should use the credit wisely and avoid all unnecessary expenses on cloud services during the group project. Also, you should monitor the balance of credit during the implementation. Please contact your tutor as soon as possible when the credit is running out. **Note that the GCP credit is not unlimited.**

You should backup the implementation code and data regularly (at least every week). It is your own interests to ensure that you fully understand the technical requirements of this group project. If you have any concerns or problems, please contact your session's tutor or ask questions on Piazza.

Submission

You should make an online submission before 4:59 PM 16/10/2020 (Week 10):
- Your proposal should be no more than 1,000 words in two pages and should be submitted to Turnitin;
- The source code and the relevant data should be compressed into a single file and should be submitted to Blackboard.

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