Course Code	Course Title
COMP 4651	**Cloud Computing and Big Data Systems**

Course Description

Big data systems, including Cloud Computing and parallel data processing frameworks, emerge as enabling technologies in managing and mining the massive amount of data across hundreds or even thousands of commodity servers in datacenters. This course exposes students to both the theory and hands-on experience of this new technology. The course will cover the following topics. (1) Basic concepts of Cloud Computing and production Cloud services; (2) MapReduce - the de facto datacenter-scale programming abstraction - and its open source implementation of Hadoop. (3) Apache Spark - a new generation parallel processing framework - and its infrastructure, programming model, cluster deployment, tuning and debugging, as well as a number of specialized data processing systems built on top of Spark. By walking through a number of hands-on labs and assignments, students are expected to gain first-hand experience programming on real world clusters in production datacenters. *Prerequisite(s):* COMP 2011 OR COMP 2012H

List of Topics

1. Introduction and Cloud Fundamentals
2. Service Models and Cloud Challenges
3. Virtualization
4. Storage Systems: GFS and HDFS
5. MapReduce Programming Model
6. Hadoop
7. MapReduce Algorithm Design
8. From MapReduce to Spark
9. Spark Internals
10. Spark Programming
11. Advanced Spark Programming: From RDD to DataFrames
12. Graph Analytics
13. Container Orchestration
14. Resource Management and Scheduling
15. Serverless Computing
16. Other Advanced Topics

Textbooks

Since Cloud computing and big data systems are emerging technologies under heavy development, there is no official textbook.

Reference books

The followings books are good references to learn Hadoop and Spark programming:

- T. White, ‘‘Hadoop: The Definitive Guide,’’ 4th Eds, O'Reilly, 2015.
- H. Karau, A. Konwinski, P. Wendell and M. Zaharia, ‘‘Learning Spark,’’ O'Reilly, 2015.

In addition to the reference books, some course materials come from seminal papers published in recent-years’ top conferences.

Grading Scheme

Homework and Lab	30%
Course project	20%
Final Exam	50%
Total	100%

Course Intended Learning Outcomes

1. Describe the motivation, objectives, and architecture of cloud computing and big data systems.
2. Understand the use of production cloud computing platform.
3. Understand the general architecture and the use of Hadoop Distributed File System (HDFS).
4. Understand the general programming model of MapReduce and the use of Hadoop.
5. Understand Resilient Distributed Dataset (RDD) and the use of Spark programming model based on RDD.
6. Describe the major architecture difference between Hadoop and Spark.
7. Write a MapReduce/Spark program with tens to hundreds lines of code to solve common data analytics problems.
8. Use software tools to develop and debug a program written in Hadoop and Spark.
9. Understand the requirement of cluster management and container orchestration systems
10. Understand the benefits of serverless computing

Assessment Rubrics

5 Assignments (30 points):

- A1 (4 points): Amazon EC2
- A2 (6 points): Hadoop MapReduce Programming
- A3 (8 points): Spark Programming
- A4 (8 points): Advanced Spark Programming
- A5 (4 points): Container Orchestration

1 Course Project (20 points):

A term-long, open-ended project, where <=4 students team up working as a group.

The topic of the project can be tech, business, or research related. Example topics include but not limited to

- Reimplement an AWS serverless application. The applications include but are not limited to web application, image recognition, mobile backend, file processing, stream processing, IoT backend, and Chatbot. The students are expected to implement a partial or a complete set of functionalities of a particular application with the AWS components replaced with open-source alternatives, e.g., Apache Kafka substituted for Kinesis Stream, PostgreSQL for DynamoDB, OpenFaaS for AWS Lambda, Redis for S3, TensorFlow for Amazon Rekognition, Nginx for API Gateway, etc. The deliverable should be directly deployable to a Kubernetes cluster, such as a set of Docker images.
- Analyzing a dataset, public or self-collected, and drawing some insights from it (see Sample 1 and Sample 4). A few public datasets you can use: AWS Public Datasets, SF OpenData, Google cluster workload traces;
- Tackling a public challenge (as done by Sample 1): there are many data analytics competitions held on platforms such as Kaggle, TianChi, KDDCup, and CrowdAnalytics. Students could choose their interested one and give an attempt. Neither full participation nor complete solution is required.
- Implementing some non-trivial distributed algorithms on Spark (or other big data analytics framework). For instance, Sample 3 implements three distributed algorithms to achieve k-anonymity.
- Performance measurement of EC2, Spark, or Hadoop;
- A literature survey on a specific topic covered in the course (the surveyed papers are not limited to the reading list);
- ...

The delivery of the project is a project report <= 5 pages

Final Exam (50 points)