CS 5433: Big Data Management

***Time*:** 4.30pm – 7.10pm Wednesday Online

Instructor*:*
Dr J. P. Thomas
email: jpt@cs.okstate.edu.
Office Hours: Wednesday: 11.00am-12.30pm - online
Inform the instructor about wanting to meet at least 24 hours in advance and the instructor will set up an online meeting.
Alternate times to meet on an individual basis may be arranged. Contact the instructor by email.

Teaching Assistant:
Ghosh, Ipsita
email: ighosh@ostatemail.okstate.edu
Office Hours: Thursday 2:00pm - 3.30pm – online.
Inform the TA about wanting to meet at least 24 hours in advance and the TA will set up an online meeting.
Alternate times to meet on an individual basis may be arranged. Contact the TA by email.

***Prerequisite*:**
CS 5423, CS 4283 or CS 5283 or equivalents
Knowledge of Programming

***Course Description*:**
Data storage and processing for big data, Map-Reduce model for big data processing within the Hadoop software framework, big data warehouse for summarization, query and analysis using Hive, Data munging and transformation using Pig; Streaming data using Flume; Transferring structured data using Sqoop; Loading, Querying and transforming data using HBase; Setting up distributed services using ZooKeeper; Data Processing using Apache Spark; Writing Data Pipeline jobs for Hadoop; Machine learning in Spark and R.

Course Objectives
This course provides an introduction to Distributed data processing using the Hadoop Framework and Apache Spark. In the Hadoop framework, the Hadoop Distributed File System (HDFS) for storage and the MapReduce programming model for processing data will be explored. Distributed analysis and patterns for big data processing will be studied. In Apache Spark RDD transformations, real-time stream processing will be explored. Some of the tools that are a part of the Hadoop Ecosystem such as Pig, Sqoop, Flume, ZooKeeper, HBase, and Kafka will be briefly covered. The Neo4j graph database for storing and processing big data will be introduced. The class will provide an introduction to Data analytics with implementation in a big data framework such as spark or Neo4j in the form of a group project.

Course Outline

Topics to be covered

A. Parallel and Distributed Data Processing Frameworks - Hadoop
- Hadoop File Systems (HDFS)
- MapReduce Programming Model
- Streaming/Pipes API

B. Distributed Analysis and Patterns for Big Data Processing
- Summarization and Joins in MapReduce Jobs

C. Parallel and Distributed Data Processing Frameworks -Spark
- Features of Spark
- RDD
- Anatomy of Spark Jobs
- Spark on YARN
- Stream-Processing
- Real-time Querying
- Graphs in Spark

D. Big Data EcoSystem
- Hive for data summarization, query, and analysis
- Analyzing large data sets using Apache Pig
- Streaming data using Apache Flume
- Importing/Exporting Structural data using Apache Sqoop
- Writing distributed services using Apache ZooKeeper
- ETL operations using HBase
- Apache Hive™ data warehouse software facilitates and quering using SQL syntax.
- Distributed real-time streaming with Kafka

E. Graph Databases
- Data Storage and Analytics with Neo4j

F. Big Data Analytics using MLlib
- Analytical methods such as Clustering, Regression and Classification

G. Visualization with Matplotlib

Textbooks

There is no textbook for this class. The following serve as reference texts:

- Tom White, *Hadoop: The Definitive Guide,* 4th Edition, O'Reilly, 2014
- Edward Capriolo, Dean Wampler, Jason Rutherglen, *Programming Hive*: *Data Warehouse and Query Language for Hadoop,* O'Reilly, 2012
- Alan Gates, *Programming Pig,* O'Reilly, 2011
- Holden Karau, Andy Konwinski, Patrick Wendell, and Matei Zaharia, *Learning Spark,* O'Reilly, 2014

- Mark Grover, Ted Malaska, Jonathan Seidman & Gwen Shapira, *Hadoop Application Architectures: Designing real-world big data applications,* O'Reilly, 2015
- Mark Needham and Amy E. Hodler, *Graph Algorithms: Practical Examples in Apache Spark and Neo4j*, O'Relly, 2019
- Donald Miner and Adam Shook, *MapReduce Design Patterns,* O'Reilly, 2013
- Jake VanderPlas, *Python Data Science Handbook*, O'Reilly, 2017

Communication medium
All notes and class announcements will be on Canvas

***Grading*:**
- 2 Quizzes = 50
- Individual Programming Assignments = 150
- Group Project = 100
- Finals = 75

Total – 375 marks

***Letter Grades*:**
Grade A: 90 - 100 %
Grade B: 80 – 89 %
Grade C: 70 –79 %
Grade D: 60 - 69 %
Fail (Grade F): 0-59 %

Attendance Policy:
Attendance is strongly encouraged, but not required. Students are responsible for any material covered in class. Some of the material covered in class will not be in the required textbook. Announcements about tests etc. will be made in class and/or by email. Students are to check their emails regularly (using their class accounts).

Late submission penalty:
1 calendar day late: 10% penalty - date based on submission
2 calendar days late: 20% penalty - date based on submission
3 calendar days late: 30% penalty - date based on submission
4 calendar days late: 40% penalty - date based on submission
5 calendar days late: 50% penalty - date based on submission
6 or more calendar days late: 100% penalty - date based on submission

***Examinations/Tests*:** No discussion of any kind (except with the instructor) is allowed. No access to any type of written material is allowed. Students who **do not** comply with the described collaboration policy will receive a grade of F in the course. Furthermore, the case will be reported to the University Officials.

Drop and Add Policy: Students will be allowed to drop as long as the University permits them to do so. A grade of W or F will be determined on the basis of the points earned until that time.

Academic Dishonesty/misconduct: The Computer Science departmental policy for academic dishonesty and misconduct applies to this class. In addition, a student attempting to gain unfair advantage by keeping an examination paper longer than the time permitted is guilty of academic misconduct.

Computer Usage: The Computer Science departmental policy for computer usage applies to this class. Exceptions will be made for students whose companies permit use of company machines for academic work. Students taking advantage of the exception must have two-way email access.

Americans with disabilities act: The Computer Science departmental policy for students with disabilities applies to this class. Anyone who has a need for examinations by special arrangements should see the instructor as the earliest possible opportunity during scheduled office hours.