

INTRODUCTION TO BIG DATA ANALYTICS	
Course Code: CSOE10	Credits: 3:0:0
Prerequisite: Programming Knowledge	Contact Hours: 45L
Course Coordinator: Dr. Ganeshayya I Shidaganti	

Course Contents

Unit I

UNDERSTANDING BIG DATA: What is big data – why big data – Data!, Data Storage and Analysis, Comparison with Other Systems, Rational Database Management System, Grid Computing, Volunteer Computing, convergence of key trends – unstructured data – industry examples of big data – web analytics – big data and marketing – fraud and big data – risk and big data – credit risk management – big data and algorithmic trading – big data and healthcare – big data in medicine – advertising and big data – big data technologies – introduction to Hadoop – open source technologies – cloud and big data – mobile business intelligence – Crowd sourcing analytics – inter and trans firewall analytics.

- Pedagogy / Course delivery tools: Chalk and talk, Power point presentation, Videos
- Lab Component / Practical Topics: Nil

Unit II

NOSQL DATA MANAGEMENT: Introduction to NoSQL – aggregate data models – aggregates – key-value and document data models – relationships – graph databases – schema less databases – materialized views – distribution models – shading – version – map reduce – partitioning and combining – composing map-reduce calculations.

- Pedagogy / Course delivery tools: Chalk and talk, Power point presentation, Videos
- Lab Component / Practical Topics: Nil

Unit III

BASICS OF HADOOP: Data format – analyzing data with Hadoop – scaling out – Hadoop streaming – Hadoop pipes – design of Hadoop distributed file system (HDFS) – HDFS concepts – Java interface – data flow – Hadoop I/O – data integrity – compression – serialization – Avro – file-based data structures.

- Pedagogy / Course delivery tools: Chalk and talk, Power point presentation, Videos
- Lab Component / Practical Topics: Nil

Unit IV

MAPREDUCE APPLICATIONS: MapReduce workflows – unit tests with MRUnit – test data and local tests – anatomy of MapReduce job run – classic Map-reduce – YARN – failures in classic Map-reduce and YARN – job scheduling – shuffle and sort – task execution – MapReduce types – input formats – output formats.

- Pedagogy / Course delivery tools: Chalk and talk, Power point presentation, Videos
- Lab Component / Practical Topics: Nil

Unit V

HADOOP RELATED TOOLS: Hbase – data model and implementations – Hbase clients – Hbase examples – praxis. Cassandra – Cassandra data model – Cassandra examples – Cassandra clients –Hadoop integration. Pig – Grunt – pig data model – Pig Latin – developing and testing Pig Latin scripts. Hive – data types and file formats – HiveQL data definition – HiveQL data manipulation – HiveQL queries.

- Pedagogy / Course delivery tools: Chalk and talk, Power point presentation, Videos
- Lab Component / Practical Topics: Nil

Text Books:

1. Tom White, "Hadoop: The Definitive Guide", Third Edition, O'Reilley, 2012.
2. Eric Sammer, "Hadoop Operations", O'Reilley, 2012.

Reference Books:

1. Vignesh Prajapati, Big data analytics with R and Hadoop, SPD 2013.
2. E. Capriolo, D. Wampler, and J. Rutherglen, "Programming Hive", O'Reilley, 2012.
3. Lars George, "HBase: The Definitive Guide", O'Reilley, 2011.
4. Alan Gates, "Programming Pig", O'Reilley, 2011

Course Outcomes (COs):

At the end of the course, the student should be able to:

1. Describe the Role and Importance of Big Data in various Business Context
2. Describe the Aggregate Data Models and Distribution Models in managing the Big Data Storage.
3. Discuss the Architecture and Features of Hadoop Distributed File System (HDFS) for handling Big Data.
4. Discuss the Map Reduce Fundamentals and Workflows for handling Big Data.
5. Interpret on various Hadoop related Tools and explore the Operations for Analyzing Big data Applications.

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE) : 50 Marks		
Assessment Tools	Marks	Course Outcomes addressed
Internal Test-I (CIE-I)	30	CO1, CO2 & CO3
Internal Test-II CIE-II)	30	CO4 & CO5
Average of the two CIE will be taken for 30 marks		
Other Components		
Case Study	10	CO1, CO2, CO3, CO4 & CO5
Report	10	
The Final CIE out of 50 Marks = Average of two CIE tests for 30 Marks+ Marks scored in Case Study +Marks scored in Reports		
Semester End Examination (SEE)		
Course End Examination (Answer One full question from each Unit-Internal Choice)	100	CO1, CO2, CO3, CO4 & CO5

Project Management with Git	
Course Code: CSOE12	Credits: 3:0:0
Pre-requisites: Programming	Contact Hours: 42
Course Coordinator: Brunda G	

Course Content

Unit I

Introduction to Project Management: Introduction: What is a Project, What is Project Management, Project Characteristics, Project Objectives, Project Scope and Stakeholders, **Project Management Software:** Jira, Trello, Asana, Task Management and Issue Tracking, **Software Development Life Cycle:** SDLC Phases, Waterfall, Iterative, Incremental, Agile and DevOps, **Agile Frameworks:** Agile Manifesto, Scrum, Kanban, Crystal and Dynamic Systems Development Method, **Industry Practices:** User Stories, Product Backlog, Sprint Planning, Daily Stand-up, Sprint Review, Retrospective, Bug Tracking and Risk Management.

Unit II

Git: Version Control with Git: Understanding version control concepts — why we need it, centralized vs distributed VCS. **Installing Git and First-Time Setup:** Configuring username and email. **Getting a Repository:** Init and clone. **Recording Changes to the Repository:** Add, commit, status, diff. **Viewing the Commit History:** Log. **Undoing Things:** Checkout, revert, basic reset. **Working with Remotes:** Push, pull, fetch. **Tagging:** Basic tags, semantic versioning. **Git Branching:** Branches in a nutshell, basic branching and merging, branch management.

Unit III

GitHub: Account Setup and Configuration: Setting up a GitHub account and configuring it for use. **Contributing to a Project:** Forking, cloning, opening a pull request. **Maintaining a Project:** Managing releases, reviewing and merging contributions. **Branching Workflows:** GitHub Flow, trunk-based development, Gitflow. **Remote Branches:** Understanding and working with remote-tracking branches. **Rebasing:** Rebasing a feature branch instead of merging.

Unit IV

Distributed Workflows: Understanding distributed Git workflows, contributing to and maintaining a project in a distributed setup. **Git Tools:** Revision Selection, Interactive Staging, Stashing and Cleaning, Searching, Rewriting History, Reset Demystified, Advanced Merging, Debugging with Git and Submodules.

Unit V

The GitHub EcoSystem: Exploring the GitHub Marketplace - Introducing the GitHub Marketplace, Listing Your App on the Marketplace. **Attending Events-** Exploring Types of Events, Knowing What to Expect at Events, Becoming Familiar with GitHub Events, Speaking at Events. **The Parts of TENS:** **Ten Ways to Level Up on GitHub** - GitHub Help Docs, GitHub Learning Labs, GitHub In-Person Training, Project-Specific Documentation, External Community Places, Online Coding Tutorials, Online Courses and Tutorials, Blogs and Twitter, Community Forum. **Ten Ways to Improve Your Development Workflow** - Drafting Pull Requests, Git Aliases, Run Tests Automatically, Take Breaks, Prototype User Interfaces, Scaffold Apps with Yeoman, Chrome Web Developer Tools, StackOverflow, Code Analysis Tools, Project Boards.

Textbooks:

1. Pro Git: Scott Chacon and Ben Straub, 2nd Edition, 2014, Apress, ISBN: 978-1-4842-0076-6 (available free online at git-scm.com/book under Creative Commons Attribution Non-Commercial Share Alike 3.0 license)
2. GitHub® For Dummies® Published by: John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030-5774, www.wiley.com
3. Ken Schwaber and Jeff Sutherland, The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game.

Reference Book:

1. Directing the Agile Organization-By Evan Leybourn
2. <https://www.geeksforgeeks.org/software-engineering/what-are-agile-frameworks/>

Course Outcomes (COs):

At the end of the course, the students will be able to:

1. Demonstrate software project management practices and Agile methodologies. (PO-1, 2, -3, 4, 5,9,11)
2. Apply Git commands and version control techniques for software development. (PO-1,2, 3, 4, 5, 6,9,11)
3. Demonstrate GitHub-based collaboration using branches, pull requests, and distributed workflows. (PO-1, 2, 3,4, 5,6,9,11)
4. Apply advanced Git techniques for merging, conflict resolution, debugging, and repository management. (PO-1, 2, -3, 4, 5, 6,9,11)
5. Utilize GitHub ecosystem tools and best practices to improve development workflows. (PO-1, 2, -3, 4, 5,9,11)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE) : 50 Marks		
Assessment Tools	Marks	Course Outcomes addressed
Internal Test-I (CIE-I)	30	CO1,CO2 & CO3
Internal Test-II CIE-II)	30	CO4 & CO5
Average of the two CIE will be taken for 30 marks		
20 Marks Components		
GitHub Project Assignment	20	CO1, CO2, CO3, CO4 & CO5
The Final CIE out of 50 Marks = Average of two CIE tests for 30 Marks+ Marks scored in Demonstration		
Semester End Examination (SEE)		
Course End Examination (Answer One full question from each Unit-Internal Choice)	100	CO1, CO2, CO3 CO4 and CO5