# Problem Statement

Set up a Hadoop data lake using Apache distribution and prepare operational and monitoring strategy and perform routine DevOps tasks similar to a realistic use case

# Overview

As part of this case study, team needs to set up a Hadoop data lake and implement a sample Hadoop and Spark use case. The team will have to develop a reference use case using below technologies:

1. Apache Hadoop/Yarn
2. Pig
3. Hive
4. Apache Spark
5. Ranger
6. Nagios
7. Ganglia
8. ELK

For e.g., the team can use the reference movie lens data and create use case for the following:

|  |
| --- |
| * List all the movies and the number of ratings |
| * List all the users and the number of ratings they have done for a movie |
| * List all the Movie IDs which have been rated (Movie Id with atleast one user rating it) |
| * List all the Users who have rated the movies (Users who have rated atleast one movie) |
| * List of all the User with the max,min,average ratings they have given against any movie |
| * List all the Movies with the max,min,average ratings given by any user |

# Requirements

Considering the above application, the following are the high level requirements:

1. Set up an Apache Hadoop environment (4 node cluster).
2. Enable HA for Name node, Resource Manager
3. Enable Kerberos
4. Set up data access using Ranger
5. Prepare 1 oozie and Azkaban workflow based on reference use case
6. Set up Monitor console using appropriate tools (Nagios/Ganglia)
   * Infrastructure monitoring (CPU, Memory etc.)
   * Platform monitoring (HDP services and parameters)
   * Application monitoring (MR Jobs, Queries etc.)
   * Log monitoring (ELK stack)
7. Prepare the following artifacts
   * Monitor dashboard
   * Weekly Health reports
8. Demonstrate decommissioning of node and impact on cluster data
9. Commission the node again and impact on cluster data

# Deliverables

1. Reference use case source code
2. Test environment set up document
3. Solution architecture for setting up cluster
4. Monitoring configurations and reports