# SWE 307 BIG DATA PROJECT - 3 DESCRIPTION

## Kafka-Spark-Cassandra Data Path

**Due date:** 27.11.2025 Thursday, in class.

In this project study, you are asked to create a data path that continuously retrieve, process and store data. The architecture of the system is given in Figure 1. Since we do not have real POS (Point Of Sale) devices, a simple POS device simulator server will be provided to you. This POS server will be started up in the background before your project servers start running and stay alive thorough the project.

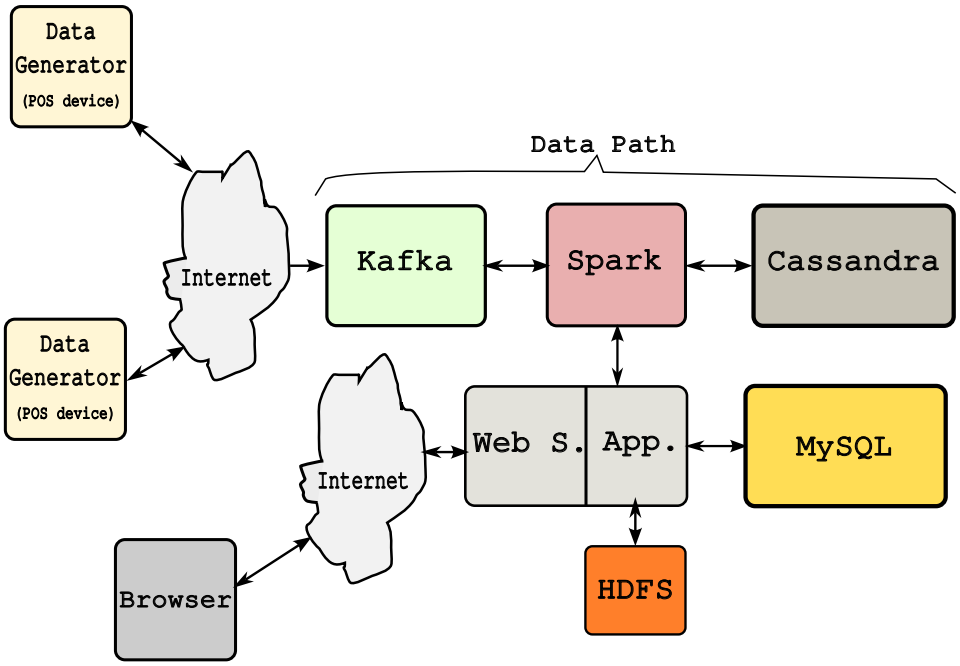


**Figure 1.** The architectural block diagram of the project 3 study.

Here are the descriptions of each block in the figure:

**Data generators:** This modules generate random credit card expenses for the people recorded in MySQL-DB. In this project only one generator server will run in the background. Every second a new expense record will be generated in CSV format like: <user_id, date_time, description, type, count, payment>

**Definitions:**

| Label | Explanation |
|---|---|
| **user_id** | empno in emp table. |
| **date_time** | Date and time of the payment |
| **description** | Example: macaroni, fridge, Cinema etc. |
| **type** | Example: food, appliances, entertainment etc. |
| **count** | Number of item bought |
| **payment:** | Amount of money paid, float number, 2 digits precision after the decimal point. |

**Kafka:** This module collects the expenses at producer input generated by the POS server. There will be four topics in Kafka server named with departments: accounting, research, sales, operations. Expense records are collected under topics based on department name where the person in.

**Spark**: This module performs two tasks:

1) Gets every data record from Kafka consumer channel and immediately store them into the Cassandra DB.

2) Calculates cumulative expense report for a user when queried. For example, if a query sent from the browser for user “Scott” then all information along with total expenses must be shown in the browser. In this case, Spark will fetch/filter relevant data from Cassandra-DB, calculate total expense amount and send the answer to the app-controller.

**Cassandra**: This is a distributed NoSQL database that stores anything coming from Spark module quickly. Spark module fetches data infrequently when an expense report is asked from browser.

The lower layer of the system (see Figure 1) is just a copy of the Project 2 (Redis may be removed in this project). The user information for this project is stored in the "emp.csv" file, you can import them into MySQL-DB. The department information is stored in the "dept.csv" file. These file contents are also given below.

What is required from you is as follows:

1) Hadoop-DFS, Spark and Cassandra clusters must be installed as single node in your computer.

2) A simple Java Spring-Boot application will be developed to perform the following tasks:

a) Personnel and department data will be fetched from MySQL.

b) Expenses will be fetched by Spark and calculations will be done in Spark. Total will be sent to controller.

b) Personnel images will be stored to/fetched from HDFS.

c) Expected web page will show the following information:

<*Image of employee, ename, mgr. name, salary, commission, department, total_expense*>

Examples will be done in the class.

**PS:**
**1)** You are free to use G-Drive or AWS-S3 as file storage instead of HDFS.
**2)** Example image files and csv files will be provided on Github repository, you can clone/download everything provided.

Here you are text data as well:
emp.csv
empno,ename,job,mgr,hiredate,sal,comm,deptno,img
7369,SMITH,CLERK,7902,17-DEC-1980,800,,20,smith.jpg
7499,ALLEN,SALESMAN,7698,20-FEB-1981,1600,300,30,allen.jpg
7521,WARD,SALESMAN,7698,22-FEB-1981,1250,500,30,ward.jpg
7566,JONES,MANAGER,7839,2-APR-1981,2975,,20,jones.jpg
7654,MARTIN,SALESMAN,7698,28-SEP-1981,1250,1400,30,martin.jpg
7698,BLAKE,MANAGER,7839,1-MAY-1981,2850,,30,blake.jpg
7782,CLARK,MANAGER,7839,9-JUN-1981,2450,,10,clark.jpg
7788,SCOTT,ANALYST,7566,09-DEC-1982,3000,,20,scott.jpg
7839,KING,PRESIDENT,\000,17-NOV-1981,5000,,10,king.jpg
7844,TURNER,SALESMAN,7698,8-SEP-1981,1500,0,30,turner.jpg
7876,ADAMS,CLERK,7788,12-JAN-1983,1100,,20,adams.jpg
7900,JAMES,CLERK,7698,3-DEC-1981,950,,30,james.jpg
7902,FORD,ANALYST,7566,3-DEC-1981,3000,,20,ford.jpg
7934,MILLER,CLERK,7782,23-JAN-1982,1300,,10,miller.jpg

dept.csv
deptno,dname,loc
10,ACCOUNTING,NEW YORK
20,RESEARCH,DALLAS
30,SALES,CHICAGO
40,OPERATIONS,BOSTON