Data Assignment

DESCRIPTION

The objective of this assignment is to implement a distributed system that handles csv data, transforms those and persists them in parquet format. The system should be composed of the following two modules:

1. A distributed file system (Hadoop cluster) where the csv dataset-file will be stored and the resulting parquet files will be persisted.
2. A spark cluster that will run on top of Hadoop and will process the csv data in order to produce the parquet files.

In Channel VAS we use mostly Scala for this kind of tasks, but if you are not comfortable with it, you can use Python. It does not matter for the evaluation of this task which language you will chose. Regardless of your language choice give the necessary attention to code quality and readability.

ASSIGNMENT DETAILS

The first main objective is to setup the Hadoop + Spark cluster. If you have some familiarity with dockers you may setup a multi-node cluster with the help of this technology. Otherwise you can setup a single-node cluster at a single vm. Spark can be setup over yarn or standalone.
You could find more info at the below sites and repositories:
https://hadoop.apache.org/docs/r3.2.1/
https://archive.apache.org/dist/hadoop/common/
https://spark.apache.org/releases/spark-release-3-1-1.html
https://archive.apache.org/dist/spark/

The input data is given in CSV format (the attached cvas_data.csv file). A spark job should be submitted at Spark cluster that will read the input csv data and produce the parquet output.
Each row in the CSV file is composed of three columns. The first column is the timestamp of a transaction, the second one is the amount of the transactions and the third one is the channel through which the transaction was done. To illustrate the format of the file, consider the fragment below:

2021-08-16 00:14:01+01,0.3,SMS
2021-08-16 00:54:43+01,0.15,SMS
2021-08-16 00:04:29+01,0.15,SMS

The Spark job should read the file, perform the necessary transformations and cleaning and persist parquet files back to a new output directory at hdfs. The table should contain the following three columns with specified data types:

1. timestamp: of data type *TimestampType* produced from first column of csv
2. amount: of data type DecimalType with four decimal digits from second column of csv
3. channel: of data type StringType from third column of csv

We would like to be able to perform a query that will produce to us the data from parquet with the right format, cleaned of any problematic records.

DELIVERABLE

The final deliverable should include the following:

• A **private** repo in GitHub. We only need read-only access to the repo (invite as collaborator p_salteris@yahoo.gr).

• A **README.md** file with instructions of how to setup, build and execute all modules.

Your deliverable will be examined based on the requirements of the assignment. Expect to be asked about your design and implementation choices, assumptions and issues that you may have encountered