**Data Science full-stack developer – Case Study**

As a full stack Data science engineer you are expected to build/test & support large-scale data processing systems and be an expert in the latest Big Data, NoSQL, advanced analytic tools & AI concepts. You should be able to drive a complete data application lifecycle - data ingestion, modelling, processing & prediction solutions using AI techniques (Deep Learning/Machine Learning) to meet business needs of the organisation. You should be a creative problem solver, resourceful in getting things done and productive working independently or collaboratively and well versed in Agile/DevOps methodology and behaviours.

**Use-Case: Airline on-time performance**

**Reference Link:** <http://stat-computing.org/dataexpo/2009/>

Have you ever been stuck in an airport because your flight was delayed or cancelled and wondered if you could have predicted it if you'd had more data? This is your chance to find out.

**The data:**

The data consists of flight arrival and departure details for all commercial flights within the USA, from October 1987 to April 2008. This is a large dataset: there are nearly 120 million records in total, and takes up 1.6 gigabytes of space compressed and 12 gigabytes when uncompressed.

**Environment:**

* **Option1:** We recommend using Google Cloud Big Data platform (dataproc) to address this uses cases. The attached document has the guidelines on how to create Google cloud account and provision a new Hadoop cluster. When you first sign-in, you will be provided with a 300$ credit to use for 365 days, which will be highly sufficient to develop the use cases mentioned.
* **Option2:** Use your own environment



1. **Batch Ingestion & Processing**
   1. **Hadoop directory Structure to be created.**

|  |  |  |
| --- | --- | --- |
| **Layer** | **Directory Path** | **File Format** |
| RAW | /data/raw/ | As is (e.g. TXT, CSV, XML, JSON, etc.,) |
| Decomposed | /data/decomposed/ | Avro |
| Modelled | /data/modelled/ | Parquet |
| Schema (Meta data) | /data/schema/ | AVSC schema |

* 1. **Source data details**

Download the stats created for year 2008 & 2007.

<http://stat-computing.org/dataexpo/2009/the-data.html>

**Supplemental Data:** <http://stat-computing.org/dataexpo/2009/supplemental-data.html>

* 1. **Data preparation**
  + Create a Kafka cluster
  + Create the following Topics in Kafka
    - Airports
    - Carriers
    - Planedate
    - OTP
  + Download the stats created for year 2008 & 2007 and load the data into a Kafka cluster under the relevant topics. Use any options of your choice to load the data to Kafka topics.
  1. **Batch Ingestion (HDFS)**
* **Raw layer (Store data AS-IS)**
  + Use *Apache Flume* to consume messages from Airports & Planedate Kafka Topic to HDFS Raw folder
  + Use *Spark Streaming* to consume messages from Carriers and OTP Kafka Topic to HDFS Raw folder
* **Decomposed layer (Append UUID and timestamp to the AS-IS data)**
  + For each message in the Airports & Planedate data from raw directory, append UUID and timestamp using Pig Latin.
  + For each message in the Carriers & OTP data from raw directory, append UUID and timestamp using Pig Latin.
  1. **Modelling and processing**
* Cleanse the data (trim, null, removing duplicates) and load it in Parquet format as modelled using Spark/Scala
  1. **Develop a solution using R/Python/Spark MLlib tools and Big Data platform to answer the following questions. Demonstrate your expertise in each tools.**
* Which carrier performs better?
* When is the best time of day/day of week/time of year to fly to minimise delays?
* Do older planes suffer more delays?
* Can you detect cascading failures as delays in one airport create delays in others? Are there critical links in the system?
* Create a model to predict flight delays
* How well does weather predict plane delays?

**Note:**

* Prepare a technical presentation along with Architecture Diagram/UML to explain the solution for the above case-study and share
* Share the code & configuration details
* Walk us through the solution, model, tools and code during assessment