** Bootcamp Project**

**INTRODUCTION**

The goal of the project is to create an e-commerce system for the food service industry that enables on-line consumers to purchase foods and restaurant supplies from a single source. The system will provide the consumer with basic point-of-sale functionality while enabling the provider to optimize their product offerings based on various product and consumer information.

From the perspective of the on-line consumer, the system must collect orders for each user and collate them into a shopping cart. The shopping cart must uniquely identify all products purchased by each user and reconcile the total purchase price.

From the perspective of the provider, the system must contain information that allows the provider to offer the most favorable products to its consumers while ensuring stability and bandwidth during peak times of the day. This includes information such as the most popular product, the most popular vendor, and the time of day when the most products are purchased.

Feel free to use any and all available references throughout the project.

**DATA MODELING**

The source data for the system consists of user data, product data, and an order stream (formats for each are shown below).

Task:

From this data, please provide a data model and associated query tables to yield the information listed in Table 1 in order to meet the above objectives. All performance related best practices should be considered, throughout. Also provide trace information for each.

Deliverables:

Please provide a detailed description of the solution including assumptions and/or difficulties encountered. Also provide the result-sets and keyspace description (CQL> describe keyspace <name>) output.

****

**Table 1 – Cassandra Data Modeling Project Objectives (Query Tables)**

1)   All product orders associated with each individual user (the Shopping Cart).  Include all descriptive attributes for each product.

2)   All product orders associated with each individual Vendor.

3)   All users that have purchased a specific product.

4)   All users that have purchased any product from a specific vendor.

5) All product orders associated with a given day.

6) All product orders within a given hourly range for a specified day.

7) All product orders within a given minute range for a specified day.

8) All product orders within a given second for a specified day.

9) All product orders for an individual within any specified day.

10) All product orders for an individual within a given daily hourly range.

11) All product orders for an individual within a given daily minute range.

12) All product orders for an individual within a given daily second range.

****

**HADOOP MAP-REDUCE**

An application of Hadoop in the DataStax Enterprise stack is to gather statistics associated with each product and develop recommendations based on those statistics. The most common form is to count occurrences of entities and/or attributes. The Products data contains a collective consumer recommendation associated with each product: highly recommend, may recommend, never recommend, undecided.

Task:

Develop map-reduce jobs that will yield the information listed in Table 2, below.

Deliverables:

Please provide a detailed description of the solution including assumptions and/or difficulties encountered. Also provide all results, source code files and output files.

**Table 2 – Hadoop Map-Reduce Project Objectives**

1) Total number of products for each recommendation.

2) The product(s) most highly recommended.

3) The product(s) most highly recommended for each vendor.

**HADOOP EXTRA CREDIT**

Develop a map-reduce job to compute the 10 period moving average of order prices in the order stream.

****

**SOLR**

An application of Solr in the DataStax Enterprise stack is to perform free-text and faceted searches within data fields. A noted challenge in e-commerce systems, for example, is the standardization of addresses where permutations of misspelled street names are analyzed.

Task:

This portion of the project will analyze street addresses for the letter ‘s’ and ‘S’. To this end, please accomplish the objective listed in Table 3, below.

Deliverables:

Please provide a detailed description of the solution including assumptions and/or difficulties encountered. Also provide solrconfig.xml, solrschema.xml, description of required column-families, and all results files.

**Table 3 – Solr Project Objectives**

1) Develop a schema for user data.

2) Validate schema.

3) Determine the number of occurrences of the letter 's' and the letter 'S' in all street addresses.

**SOLR EXTRA CREDIT**

The letter ‘s/S’ is said to be the most commonly occurring letter in the English language. Validate or disprove this in street addresses.

****

**MORE PROJECT EXTRA CREDIT**

Each user of the system has an associated Account. Using only the user data and order stream, develop a data model and query tables that maintain all transactions while periodically reconciling and updating the user’s balance in the account table. The user’s balance must have the ability to be reported correctly at all times.

**DEMO**

Time-permitting, please create a recorded demo of the project solution(s).

****

**APPENDIX – Sample Data Format and Specification**

Data is provided for 5000 users and approximately 5000 unique products delivered by 120 unique vendors.

The provided data format is as follows:

=================

Users

File:

user\_list.csv

Format:

user id,first name,middle name,last name,date of birth,street address,zip code,city name,state name,gender,phone number,email,country code

Sample Record:

U1,Sam,O.N.M.,Ellis,6/18/1971,619 Ranch Woods,30076,Walsh,North Carolina,M,379-100-3220,sam4956@hailmail.net,GG

=================

Products

FIle:

product\_list.csv

Format:

product name,unit price,vendor,rating,product id,recommendation,attribute 1, attribute 2,,,,attribute n

Sample Record:

Oil - Avocado,13.15,Reliant Foodservice,P1,undecided,bland,messy

=================

****

Orders

File:

order\_list.csv

Format:

order id,user id, product id, quantity, timestamp

Sample Record:

O1, U4860, P335, 1, 2013-08-09 00:01:01

=================

accounts

File:

account\_list.csv

Format:

account id, user id, balance

Sample Record:

A1, U1, 108.6

=================