**Use Case**

(Completely Fictional)

MarketData, Inc. is a financial technology company headquartered in Manhattan, New York that collects stock price information from various US and Global Exchanges and and uses it to generate analytics for its portfolio management division.

Because their client base is growing very fast and the stock price database is now storing historical information collected over 20 years, the CIO and CTO of the company have decided to move their infrastructure to the cloud. They have engaged you, the Cloud Database Architect, to implement the transition to the following:

1. Migrate their real-time database to **MongoDB** running in a cost-effective, secure, scalable, and redundant infrastructure on the cloud that will scale on demand, distribute load efficiently, and serve a very large number of users across one or more geographic continents.
2. At the end of the day, the close prices of all actively trading stocks in the real-time MongoDB will be exported and imported to a **Redshift** cluster that stores all historical data, since MarketData was founded.

1. Deploy **QuickSight**as the data visualization solution to generate data analysis dashboards for the company’s clients.

Provide the following:

**[50%] 1. Cloud Solution Architecture and Data Flow**

In a Word document provide a detailed explanation & architectural diagrams of the solution. Include the three architectural tiers listed above (MongoDB, Redshift, QuickSight) and networking components (e.g. Region, Availability Zones, VPC, Public Subnets, Private Subnets) that would make the solution redundant, secure, and scalable.

**If needed, AWS architecture icons can be downloaded from**[**https://aws.amazon.com/architecture/icons/ (Links to an external site.)**](https://aws.amazon.com/architecture/icons/)

Feel free to make assumptions and come up with recommendation on how and when data will be ingested in MongoDB and how and when data will be exported and imported to Redshift (for example, using an ETL Tool you can import to MongoDB. Using Studio3T you can export from MongoDB and then move to S3. Then from S3 import to RedShift). Make sure those are included in your diagrams.

**[50%] 2. Proof of Concept Implementation**

For a proof of concept implementation use free-tier resources on AWS as much as possible. You can go big on the diagram in #1 but stay as economic as possible on the implementation.

**Data Files**

Apple - [https://dpapatho-msin618-bucket.s3.amazonaws.com/fintech/AAPL.csv (Links to an external site.)](https://dpapatho-msin618-bucket.s3.amazonaws.com/fintech/AAPL.csv)

Microsoft - [https://dpapatho-msin618-bucket.s3.amazonaws.com/fintech/MSFT.csv (Links to an external site.)](https://dpapatho-msin618-bucket.s3.amazonaws.com/fintech/MSFT.csv)

Cisco - [https://dpapatho-msin618-bucket.s3.amazonaws.com/fintech/CSCO.csv (Links to an external site.)](https://dpapatho-msin618-bucket.s3.amazonaws.com/fintech/CSCO.csv)

Amazon - [https://dpapatho-msin618-bucket.s3.amazonaws.com/fintech/AMZN.csv (Links to an external site.)](https://dpapatho-msin618-bucket.s3.amazonaws.com/fintech/AMZN.csv)

Verizon - [https://dpapatho-msin618-bucket.s3.amazonaws.com/fintech/VZ.csv (Links to an external site.)](https://dpapatho-msin618-bucket.s3.amazonaws.com/fintech/VZ.csv)

Oracle - [https://dpapatho-msin618-bucket.s3.amazonaws.com/fintech/ORCL.csv (Links to an external site.)](https://dpapatho-msin618-bucket.s3.amazonaws.com/fintech/ORCL.csv)

Facebook - [https://dpapatho-msin618-bucket.s3.amazonaws.com/fintech/FB.csv (Links to an external site.)](https://dpapatho-msin618-bucket.s3.amazonaws.com/fintech/FB.csv)

Netflix - [https://dpapatho-msin618-bucket.s3.amazonaws.com/fintech/NFLX.csv (Links to an external site.)](https://dpapatho-msin618-bucket.s3.amazonaws.com/fintech/NFLX.csv)

Twitter - [https://dpapatho-msin618-bucket.s3.amazonaws.com/fintech/TWTR.csv (Links to an external site.)](https://dpapatho-msin618-bucket.s3.amazonaws.com/fintech/TWTR.csv)

Google  - [https://dpapatho-msin618-bucket.s3.amazonaws.com/fintech/GOOG.csv (Links to an external site.)](https://dpapatho-msin618-bucket.s3.amazonaws.com/fintech/GOOG.csv)

**Implementation Steps**

1. Import the CSV files above to a collection, for example stockprices, in a MongoDB database.
2. In MongoDB the CSV files will be imported as JSON documents. Export the JSON documents and upload them to an S3 Bucket.
3. Deploy a Redshift cluster and create a table to hold the prices from the JSON files. Decide on a distribution key and sort key for this table.
4. Import into that table the JSON data that resides on the S3 Bucket. The following article is very helpful explaining how to import to Redshift from JSON: [https://docs.aws.amazon.com/redshift/latest/dg/copy-usage\_notes-copy-from-json.html (Links to an external site.)](https://docs.aws.amazon.com/redshift/latest/dg/copy-usage_notes-copy-from-json.html)

**Note: This import is the most time-consuming task of this assignment. Allow yourself enough time for trial and error.**

1. Once the data is on Redshift, create a new analysis and data sources from it on QuickSight and generate the following visualizations:

* **Price comparison chart** for the following stocks: AMZN, GOOG.
* **Price comparison chart** for the rest of the stocks: MSFT, FB, APPL, VZ, ORCL, TWTR, NFLX, CSCO.
* **Sample Portfolio Performance Chart:**Assume in January 2019, an investor bought $100,000 in the following stocks:  
  25% MSFT

5% AMZN

5% GOOD

25% APPL

15% CSCO

25% FB

 Provide a chart that shows how the portfolio' value moved from the day the investor bough it until now. How much is this portfolio worth now?