**IST 615 – Cloud Management**

**Car Retail Website**

**Project Proposal**

**Due Date:** 10/24/2022

**Re-submission Date:** 10/31/2022

**Number of Pages: 03**

**Team members:**

* Mikhail Pravin Pinto ([mpinto01@syr.edu](mailto:mpinto01@syr.edu))
* Kirat Saran ([ksaran@syr.edu](mailto:ksaran@syr.edu))

**Purpose and Goals:**

Since the epidemic forced individuals to buy vehicles to avoid contact with other people, the demand for cars has increased substantially. This has increased market demand for used vehicles. We will build a car retail website where buyers can filter based on manufacturers, mileage, distance traveled, etc. to assist them to determine exactly what they want in a car. We'll use Microsoft Azure Cloud Services to implement this.

**Dataset:**

The dataset is from the site CarDekho.com which was uploaded to Kaggle.com.

The following details of cars are included in the dataset:

1. ID  
2. URL (to the listing)  
3. Region (the region where the car was listed)  
4. Price (selling price)  
5. Year (the car was listed)  
6. Manufacturer (brand name of the car)  
7. Model (model of the vehicle)  
8. Condition

9. Number of Cylinders

10. Fuel (fuel type)

11. odometer (Miles traveled)

12. title of the car

13. Transmission

14. VIN

15. Drive

16. Size

17. Type

18. paint color

19. image of the car (URL)

20. description

21. state

22. lat (latitude)

23. long (longitude)

24. posting date

**Link to the dataset:**

<https://www.kaggle.com/datasets/austinreese/craigslist-carstrucks-data?datasetId=62920>

**Size**: 1.5 GB

The data needs to be cleaned for null values in some of the columns. We will also delete that columns that would not be useful in our analysis. This will be done before the loading of the data into Azure Blob Storage.

**Design/Solution Architecture:**

Diagram

Description automatically generated

**Cloud Services**

***Azure Blob Storage***: The dataset which is in CSV format will be uploaded to the Azure blob service and stored in a container.

***Azure SQL Server***: By integrating Azure blob storage and SQL server we can create a pipeline job that will extract data from CSV transform it into a database format and finally store it in the destination SQL server created. The process will be manually, or time triggered.

***Azure Databricks / Power Bi Dashboards***: Data will be imported to Databricks if necessary to work on visualizations and models using Python. If it is not necessary, we will directly integrate SQL Server data into Power BI to create dashboards where users will have insights about available cars, features, estimated price, car sale trends and more.

***App Services (WEB + SQL)***: Using App Services the SQL server and the Website hosted will be integrated. Whenever the user clicks on the button the query in the SQL server will be triggered that will retrieve data according to the user’s choice.

***Implementation of front-end:***

**Page 1: (Retrieving vehicle information according to inputs by user)**

This page will have user inputs like the brand of the car, the transmission of the car, the type of car, the color of the car, price of the car. This will retrieve car data for that brand as well as cars with similar characteristics which might not be from the same manufacturer or of the same color.

**Page 2: (Selling Price estimate) (If time persists)**

According to the price of the cars already present in the data set we will estimate the price of cars if a user wants to know how much their car is valued in the market. If the user likes the estimated price, they can list their car on our website. This would add the data to the database in the SQL server.

**Dashboards:**

To build an analytic model using our database where we have multiple cars of the same brand and type and model, we can use that data to give an option different option to the user based on the condition of the car and the title of the car. We can show the user the average price for the cars they are looking for. We will further come up with a better model and tweak it on the go if we have new ideas. We have not yet come up with a plan to integrate it with our frontend.

To code and connect the database to the front-end ***Visual studio*** will be used. To interact with the database ***Azure data studio*** will be used.