# **CSC 405-01: Progress Report 1**

Our group has been given data on two different Volvo Vehicles (Trucks). These trucks have been equipped with many, many sensors that log data on everything from transmission oil temperature to the time the driver spends on the accelerator, for that given vehicle. One of the main areas that is in question surrounds a new power unit called the "APU". It does not depend on power from the motor, and general assumptions would say that would call for better efficiency and longevity of the truck's motor. Using time series manipulations on the data given should give insight on whether these assumptions are possibly true. These trucks are based in areas with variations in elevation, so the outcome should be interesting.

The goals of this project include analyzing Volvo vehicle long haul and short haul data, use of time series to aid in visualizations for analysis, determining if different metrics logged by sensors make a difference in relation to an "APU" module that is not dependent on power from the motor, and the reporting any discovered anomalies detected by data manipulations.

**Creation of file hierarchy and implementation of Pandas DataFrames:**

This has been completed by James Polk. The follow folders have been created: src/, doc/, /data, and util/. Within the data directory, an empty folder trucks/ for truck1.csv and truck2.csv can be found. Since these files are too large to be uploaded to GitHub, each team member is responsible for ensuring that the .csv files reside here in their local directories. The dictionary files for the two .csv files can be found in data/dictionary on the repository. The src/ folder houses VolvoTrucksAnalytics.ipynb, the main iPython Notebook for the project. Currently the util/ folder is unused.

Notebook blocks have been created for imports and constants, functions relating to data cleanup and wrangling, and logic for said cleanup. Functions for the creation of Pandas DataFrames for each truck have been coded as well as the culling of extraneous UTC date and time data that is already present in a DateTime stamp in the DataFrame. Finally, some basic data statistics regarding the volume of the data has been presented.

**Adjust and format the data dictionary to metrics being used:**

This was completed by Christopher Thacker. The data that's been provided isn't very useful if we can't understand what it means, so the provided column names needed to be changed. These changes included removing unnecessary column name parts, changing title formats, and including the unit of measurement, if provided, to the title for easier readability. Ambiguous titles were clarified to the best of the team's knowledge to-date, but slight changes may come in the future. The created function changes the column names with a Python dictionary which is currently being generated by CSV files.

**Determine whether there are rows or columns with all NaN values, if so remove them:**

This was completed by Wahab Ehsan. In order to find rows that are not useful because of the excessive amounts of NaN types, the following conditions were taken into consideration. The number of columns for each truck is different. The times NaN type in truck1 does not equal to the number of time NaN types appear on truck2. Columns with all NaN types are useless and are deleted using pandas. Rows are deleted by the threshold value percent. If the Non-NaN values are less than the percent given, then the row would be deleted. Initially the percent threshold is set to 75 at the moment as a global variable and can be decremented or incremented depending on the data needed.

**Merge weights together to determine how weight affects performance of the vehicles:**

This was completed by Ioannis Batsios. The columns with the same weight were merged together by the weight while the other columns that included numeric data were merged by using the mean of the data. Ioannis also looked at summing the data, but that appeared to be fruitless for most of the statistics, but it may be valuable when determining total mileage per vehicle as a vehicle begins to break down the more miles it has driven.