**MAKERERE     UNIVERSITY**

**COLLEGE OF COMPUTING AND INFORMATICS**

**SYSTEMS**

**DEPARTMENT OF NETWORKS.**

**GROUP7**

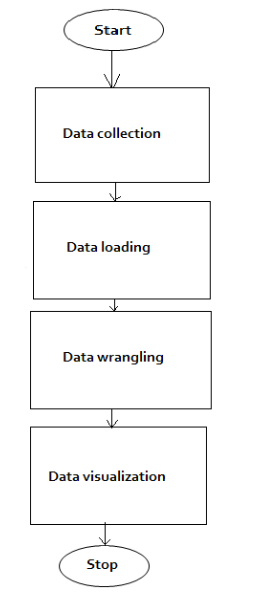
**SYSTEM DESIGNSPECIFICATION DOCUMENT (SDD)**

*GROUP7 MEMBERS:*

|  |  |  |  |
| --- | --- | --- | --- |
| **NO** | **NAMES** | **Reg No** | **Stud No** |
| 1 | Niwandiinda John Martin | 17/U/9138/PS | 217010696 |
| 2 | Miiro Henry | 17/U/6112/EVE | 217009093 |
| 3 | Onyang Joshua | 17/U/9702/PS | 217016961 |
| 4 | Odongo Abraham | 17/ U/9494/PS | 217002258 |

**SOFTWARE SPECIFICATION DESIGN**

Below is a flowchart that shows the data pipeline of from data collection up to visualization.



***Data collection.***

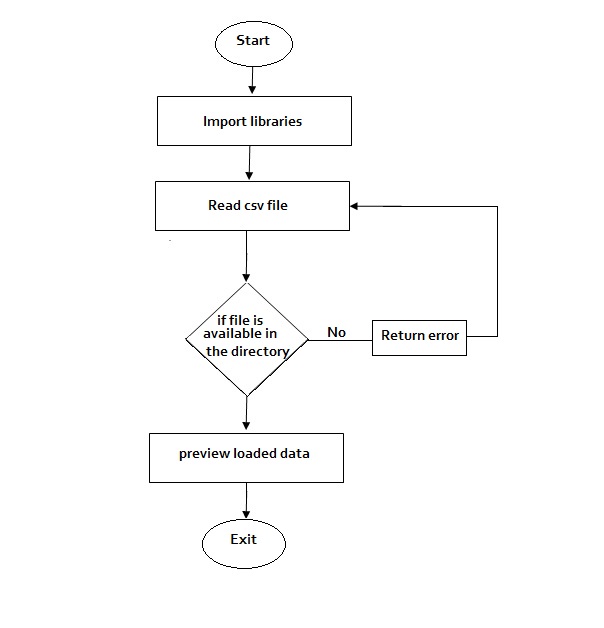
Data collection is a process of collecting information from all relevant sources to find answers to the stated problem. This is divided into primary; where the data is collected from the sources using mathematical methods and secondary; where data is obtained from already published sources like the books, online portals, etc.

We used secondary method of data collection in which we obtained the house price dataset from https//.www.kaggle.com. Kaggle also obtained the data from house sales in King county, USA which includes the house sale prices in King county and Seattle. We chose this method because it would be hard and expensive to go and collect data rather than using the already collected data.

***Data loading.***

Data loading is the process of copying and loading data or data sets from a source file, folder or application to a database or similar application. We break the data loading into the following steps

**The diagram shows data loading workflow.**



**Import libraries:**

* **Pandas.** This is a python package that provides fast, flexible and expressive data structures designed to make working with rational or labeled data both easy and intuitive***.*** pandas library help us to load our csv file into data frame using the read\_csv () function.
* **Numpy.** This is the core library for scientific computing in python. It provides high performance multidimensional array object and tools for working with these arrays.

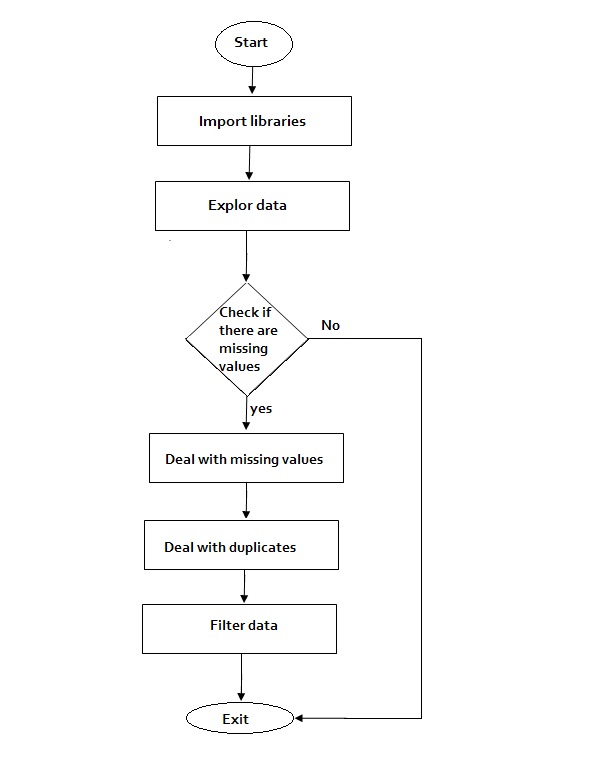
**read csv file.** Data must be loaded before every machine learning project is started.

We load our data as a csv files using pandas that is *pd.read\_csv(“fileName”).* This makes the csv file to be loaded in a table format for easy reading.

***Data wrangling.***

This is one of the steps used in preparing data for analysis that involves use of algorithms like sorting algorithm. It involves transforming data through cleaning and unifying them into suitable formats for analysis through machine learning algorithms.

Diagram showing data wrangling workflow



**Data exploration.** This is done by reading a csv file/dataset into pandas and displaying the column names along with their data types using *df.dtypes* or *df.describe().* we use the result to study the nature of all the columns.

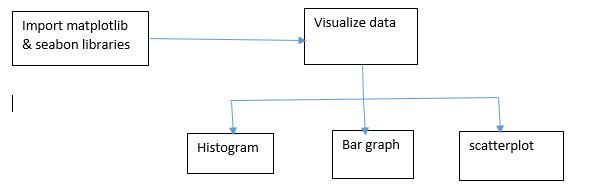
**Dealing with missing values.** With every dataset, it is vital to evaluate the missing values. How many are there? Is it an error? Are there too many missing values? Does a missing value have a meaning relative to its context? In this case we check for missing values using *df.isnull().sum().* Since there are no missing values, then we have nothing to deal with that.

**Dealing with duplicates.** To view the repeated rows, we can start off by looking at the number of unique values in each column using *df.nunique().* Basing on our result, we see that although there are 21613 rows, there are only 21436 unique ids. This means that some ids appear more than once in our dataset. To isolate these ids and view their data, we use the following. *df [df.duplicated(subset = ‘id’, keep = False).sort\_values(‘id’)].* This displays all the duplicated ids in order

**Data filtering.** We now remove ids that appear more than once in our dataset *repeat\_id[repeat\_id >1].to\_frame().reset\_index().*

***Visualization***

This is a discipline of trying to understand data by placing it in a visual context so that patterns, trends and correlation that might not otherwise be detected can be exposed.



Python provides a multiple great graphing library that come packed with lots of different features. To get little overview here are a few popular plotting libraries: *matplotlib, pandas visualization, seabon, ggplot, plotly.*

We use matplotlib and seaborn libraries to do visualization of our data.

***Matplotlib*** is a python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms.

***Seaborn*** is a python data visualization library that provides a high level interface for drawing attractive and informative statistical graphics.

We are using the following tools to visualize our data;

**Bar charts:** we use bar graphs to compare quantities of different categories. Bars represent category values and these can be vertical. The height of the bars represents the value of the category.

**Scatter plots**: This is a graph which uses Cartesian coordinates to display values for typically two variables for a set of data. The data are displayed as a collection of points, each having a value of one variable determining the position of the horizontal axis and the value of the other variable determining the position of the vertical axis. A scatter plot can suggest various kinds of correlations and skewness between the variables with a certain confidence interval.

**Histogram:** This represents the distribution of a continuous variable over a given interval. Data is plotted by grouping it together into intervals. It gives an insight into the underlying frequency distribution, outliers, skewness, etc.