NAAN MUDHALVAN PROJECT PHASE 2: **INNOVATION**

PROJECT TITLE: **Public Transportation Efficiency Analysis**

BY: Logesh G

DATA ANALYTICS

VALUABLE INNOVATION STEPS:

STEP1: DATA COLLECTIONS

- Review the initial design concept to ensure it aligns with the identified problem.
- Gather feedback from stakeholders and subject matter experts for improvements.
- Incorporate necessary changes to enhance the design's effectiveness.

STEP2: CLEANING DATA

- Clean and reprocess the data to remove outliers, errors, and inconsistencies.
- Ensure data quality before analysis.
- Transformation data into proper format for further processes

STEP 3: TRANSPORTATION PERFORMANCE EVALUATION

Transportation Models:

Utilize advanced transportation models to predict and analyze public transportation performance, taking into account factors like ridership, routes, and schedules.

Route Optimization:

Apply optimization algorithms to enhance route efficiency, reducing travel times and improving the overall quality of service.

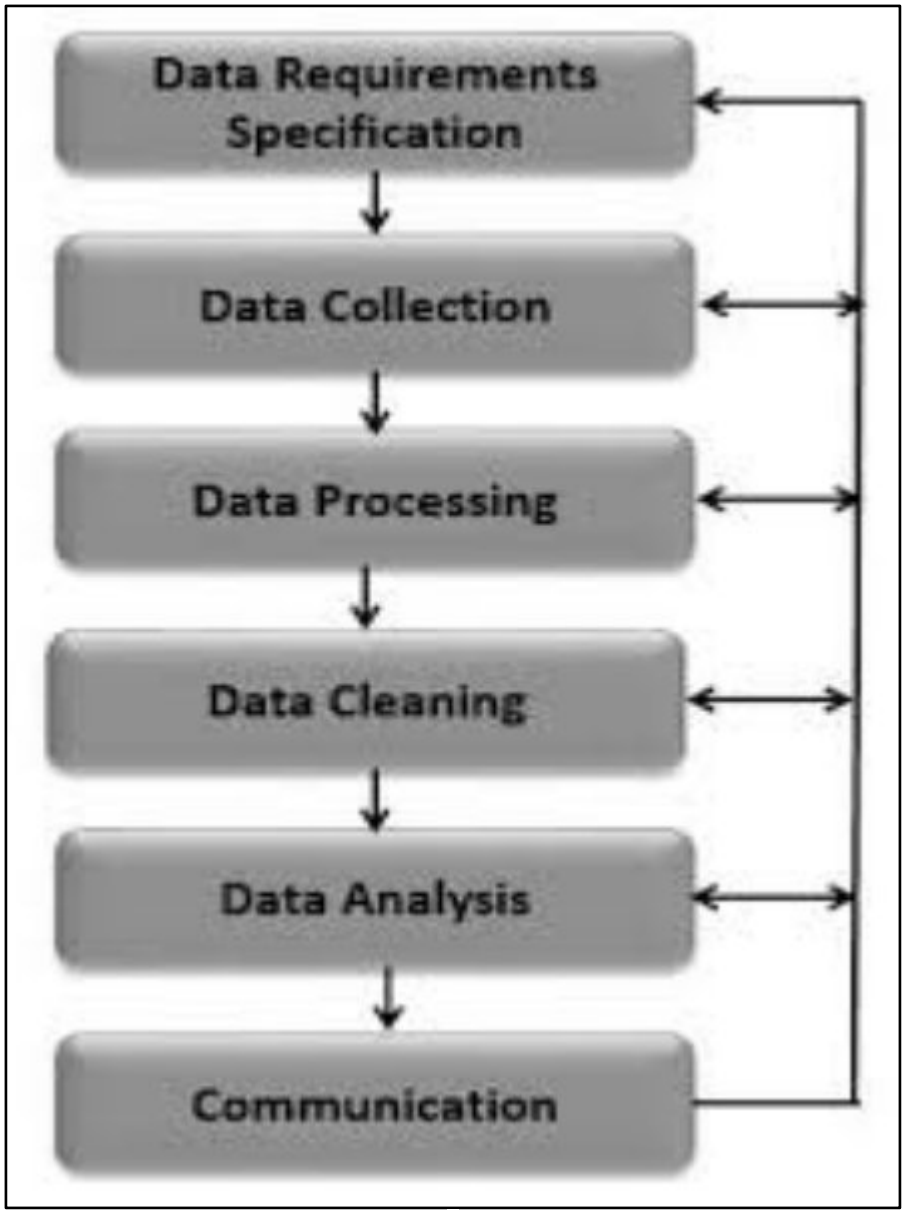
STEP 4: DATA VISUALIZATION

Create interactive maps and visual representations to convey transportation patterns and metrics, including passenger counts, on-time performance, and vehicle utilization.

STEP 5: RESULTS AND COMMUNICATION

Results are generated using appropriate calculation methods, incorporating statistical evaluations from our data collection. The final outcome highlights the efficiency and performance of the public transportation system.

PUBLIC TRANSPORTATION EFFICIENCY ANALYSIS DESIGN



Cluster:

```
import pandas as pd
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt

# Load the dataset containing transportation efficiency
metrics
data = pd.read_csv("transport-efficiency-data.csv")

# Select relevant features for clustering
selected_features = data[['efficiency_metric_1',
'efficiency_metric_2']]
```

```
# Choose the number of clusters (you can experiment with
different values)
num_clusters = 3

# Perform k-means clustering
kmeans = KMeans(n_clusters=num_clusters)
kmeans.fit(selected_features)

# Add cluster labels to the dataset
data['cluster_label'] = kmeans.labels_

# Visualize the clusters
plt.scatter(data['efficiency_metric_1'], data['efficiency_metric_2'],
c=data['cluster_label'])
plt.xlabel('Efficiency Metric 1')
plt.ylabel('Efficiency Metric 2')
plt.title('Public Transportation Efficiency Clustering')

# Show the plot
plt.show()
```