Model Development Phase Template

Date	27 June 2025
Team ID	LTVIP2025TMID44004
Project Title	TrafficTelligence: Advanced Traffic Volume Estimation with Machine Learning
Maximum Marks	4 Marks

Initial Model Training Code, Model Validation and Evaluation Report

The initial model training code will be showcased in the future through a screenshot. The model validation and evaluation report will include classification reports, accuracy, and confusion matrices for multiple models, presented through respective screenshots.

Initial Model Training Code:

Model 1(Random Forest Regressor)

```python
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x_scaled, y_value, test_size=0.2)

from sklearn.ensemble import RandomForestRegressor
regressor = RandomForestRegressor(n_estimators = 100, max_depth = 9,random_state = 0)
model=regressor.fit(x_train,y_train)
y_pred=model.predict(x_test)

from sklearn.metrics import mean_squared_log_error, r2_score
r2_score(y_pred, y_test)
```

Mode 2 (SVR)

```python
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x_scaled, y_value, test_size=0.2)

from sklearn.svm import SVR
regressor = SVR(kernel = "rbf")
model=regressor.fit(x_train,y_train)

y_pred=model.predict(x_test)
from sklearn.metrics import r2_score
r2_score(y_pred, y_test)
```

Model 3 (Polynomial Regression)

```python
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x_poly, y_value, test_size=0.2)

from sklearn.linear_model import LinearRegression
model=LinearRegression()
model.fit(x_train, y_train)

y_pred=model.predict(x_test)
from sklearn.metrics import mean_squared_log_error, r2_score
r2_score(y_pred, y_test)
```

Model 4 (XG Boost)

```python
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x_scaled, y_value, test_size=0.2)

import xgboost as xgb
model=xgb.XGBRegressor()
model.fit(x_train, y_train)

y_pred=model.predict(x_test)
from sklearn.metrics import mean_squared_log_error, r2_score
r2_score(y_pred, y_test)
```

Model Validation and Evaluation Report:

| Random Forest Regressor | ```python
from sklearn.metrics import mean_squared_error, r2_score, mean_absolute_error
print("Mean Square Error: ", mean_squared_error(y_test, y_pred))
print("Mean Absolute Error: ", mean_absolute_error(y_test, y_pred))
print("R-square Score: ", r2_score(y_test, y_pred))

Mean Square Error:  198008.8816750678
Mean Absolute Error:  280.9588373446771
R-square Score:  0.9277791335225944
``` | 92% | Not Applicable |
| Support Vector Regressor | ```python
from sklearn.metrics import mean_squared_error, r2_score, mean_absolute_error
print("Mean Square Error: ", mean_squared_error(y_test, y_pred))
print("Mean Absolute Error: ", mean_absolute_error(y_test, y_pred))
print("R-square Score: ", r2_score(y_test, y_pred))

Mean Square Error:  2104039.9111552383
Mean Absolute Error:  1256.7067758496808
R-square Score:  0.23504232546490522
``` | 23% | Not Applicable |
| Model | Regression Report | R2 Score | Confusion Matrix |

Polynomial Regression		76%	Not Applicable
	```from sklearn.metrics import mean_squared_error, r2_score, mean_absolute_error print("Mean Square Error: ", mean_squared_error(y_test, y_pred)) print("Mean Absolute Error: ", mean_absolute_error(y_test, y_pred)) print("R-square Score: ", r2_score(y_test, y_pred)) Mean Square Error: 646496.8829842781 Mean Absolute Error: 588.6552844192978 R-square Score: 0.7605639174654056```		
XG Boost		95%	Not Applicable
	```from sklearn.metrics import mean_squared_error, r2_score, mean_absolute_error print("Mean Square Error: ", mean_squared_error(y_test, y_pred)) print("Mean Absolute Error: ", mean_absolute_error(y_test, y_pred)) print("R-square Score: ", r2_score(y_test, y_pred)) Mean Square Error: 120958.54825379612 Mean Absolute Error: 228.5786688810355 R-square Score: 0.9563201748182905```		