

## Model Development Phase Template

Date	20 july 2024
Team ID	Team-739770
Project Title	Predicting the energy output of wind turbine based on weather conditions.
Maximum Marks	10 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 a summary and training and validation performance metrics for multiple models, presented through respective screenshots.

### Initial Model Training Code (5 marks):

```
[45] x = df.drop("Output_Energy",axis=1)
     y = df["Output_Energy"]

[46] print(x.shape, y.shape)
... (4447, 5) (4447,)

[47] x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2, random_state=42)

     linreg = LinearRegression()
     linreg.fit(x_train, y_train)
     linreg_y_pred = linreg.predict(x_test)
     print("Mean Squared Error:",mean_squared_error(y_test, linreg_y_pred))
     print("R2 Score:",r2_score(y_test, linreg_y_pred))

... Mean Squared Error: 360119.0424705109
     R2 Score: 0.8116999208318117
```

### Model Validation and Evaluation Report (5 marks):

Model	Summary	Training and Validation Performance Metrics
Randomforest	<pre>rf = RandomForestRegressor() rf.fit(x_train, y_train) y_pred = rf.predict(x_test) print("Mean Squared Error:", mean_squared_error(y_test, y_pred)) print("R2 Score:", r2_score(y_test, y_pred))</pre> <p>Mean Squared Error: 63937.1698857864 R2 Score: 0.965715782318059</p>	<pre>svm = SVR() svm.fit(x_train, y_train) y_pred = svm.predict(x_test) print(mean_squared_error(y_test, y_pred)) print(r2_score(y_test, y_pred))</pre> <p>1830923.5162077667 0.04264145353809867</p>
GradientBoostingRegression	<pre>from sklearn.ensemble import GradientBoostingRegressor gbr = GradientBoostingRegressor() gbr.fit(x_train, y_train) y_pred = gbr.predict(x_test) print(r2_score(y_test, y_pred))</pre> <p>0.9593288014515995</p>	<pre>svm = SVR() svm.fit(x_train, y_train) y_pred = svm.predict(x_test) print(mean_squared_error(y_test, y_pred)) print(r2_score(y_test, y_pred))</pre> <p>1830923.5162077667 0.04264145353809867</p>