Project Development Phase

Model Performance Test

Date	13 November		
Team ID	PNT2022TMID43816		
Project Name	Project -Web phishing detection		
Maximum Marks	10 marks		

Model Performance Testing:

Project team shall fill the following information in model performance testing template.

S.NO.	Parameters	Values	Screenshots Random Forest Regression / Classification			
	Metrics	CLASSIFICATION				
1.		MODEL: Random Forest Regression Accuracy score – 97%	In [24]: from sklearn.ensemble import RandomForestClassifier			
			<pre>In [25]: RF=RandomForestClassifier() RF.fit(x_train,y_train)</pre>			
			Out[25]: RandomForestClassifier()			
			<pre>In [26]: #prediction on the test data RF_pred_test=RF.predict(x_test)</pre>			
			<pre>RF_pred_test Out[26]: array([-1, 1, -1,, -1, -1], dtype=int64)</pre>			
			In [27]: #predccition in the train data			
			RF pred train=RF.predict(x train)			
			RF_pred_train			
			Out[27]: array([-1, 1, 1,, 1, 1, 1], dtype=int64)			
2.	Accuracy	Accura7cy -Test	In [28]: ## Accuracy of DT model from sklearn.metrics import accuracy_score			
			accuracy_score(y_test,RF_pred_test)			
			Out[28]: 0.9701492537313433			
			<pre>In [30]: #Random Forest Regression RF_train = accuracy_score(y_train,RF_pred_train) RF_test = accuracy_score(y_test,RF_pred_test)</pre>			
			<pre>print("Accuracy on training Data: {:.3f}".format(RF_train)) print("Accuracy on test Data: {:.3f}".format(RF_test))</pre>			
			Accuracy on training Data: 1.000 Accuracy on test Data: 0.970			
			<pre>In [31]: #confusion matrix from sklearn import metrics metrics.confusion_matrix(y_test,RF_pred_test)</pre>			
			Out[31]: array([[958, 41],			

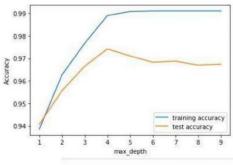
1. METRICS:

CLASSIFICATION REPORT:

Random Forest Regression / Classification

```
In [24]: from sklearn.ensemble import RandomForestClassifier
In [25]: RF=RandomForestClassifier()
RF.fit(x_train,y_train)
Out[25]: RandomForestClassifier()
In [26]: #prediction on the test data
RF_pred_test=RF.predict(x_test)
RF_pred_test
Out[26]: array([-1, 1, -1, ..., -1, -1], dtype=int64)
In [27]: #predccition in the train data
RF_pred_train=RF.predict(x_train)
RF_pred_train
Out[27]: array([-1, 1, 1, ..., 1, 1], dtype=int64)
```

PERTORMANCE:



Out[83]:		ML Model	Accuracy	f1_score	Recall	Precision
	0	Gradient Boosting Classifier	0.974	0.977	0.994	0.986
	1	CatBoost Classifier	0.972	0.975	0.994	0.989
	2	Random Forest	0.969	0.972	0.992	0.991
	3	Support Vector Machine	0.964	0.968	0.980	0.965
	4	Decision Tree	0.958	0.962	0.991	0.993
	5	K-Nearest Neighbors	0.956	0.961	0.991	0.989
	6	Logistic Regression	0.934	0.941	0.943	0.927
	7	Naive Bayes Classifier	0.605	0.454	0.292	0.997
	8	XGBoost Classifier	0.548	0.548	0.993	0.984
	9	Multi-layer Perceptron	0.543	0.543	0.989	0.983

2.Accuracy