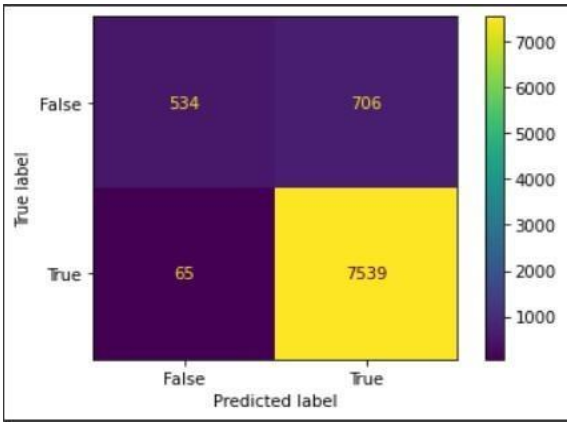
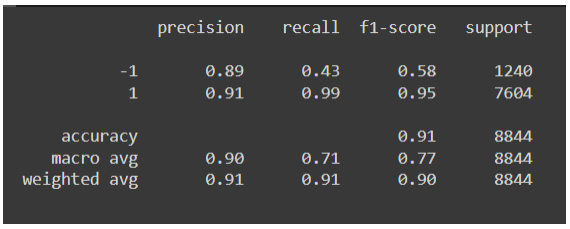
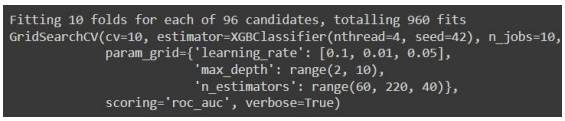
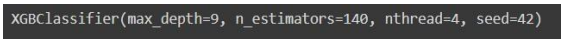
Project Development Phase
Model Performance Test

Date	18 November 2022
Team ID	PNT2022TMID23787
Project Name	Project - Web phishing detection
Maximum Marks	10 Marks

S.no	Parameter	Values	Screenshot
1.	**Metrics**	**Classification Model**	
		Confusion Matrix confusion_matrix = metrics.confusion_matrix(y_train,y_train_xgb) cm_display = metrics.ConfusionMatrixDisplay(confusion_matrix = confusion_matrix, display_labels = [False, True]) cm_display.plot() plt.show()	
		Accuracy Score acc_train_xgb = accuracy_score(y_train,y_train_xgb) acc_test_xgb = accuracy_score(y_test,y_test_xgb) print("XGBoost: Accuracy on training Data: {:.3f}".format(acc_train_xgb)) print("XGBoost : Accuracy on test Data: {:.3f}".format(acc_test_xgb))	XGBoost: Accuracy on training Data: 0.913 XGBoost : Accuracy on test Data: 0.905

		Classification Report classification_report=metrics.classification_report(y_train,y_train_xgb) print(classification_report)	

2.	**Tune the model**	**Hyperparameter tuning**
		
		from xgboost import XGBClassifier
		from sklearn.model_selection import GridSearchCV
		estimator = XGBClassifier(
		objective= 'binary:logistic',
		nthread=4,
		seed=42
		)
		parameters = {
		'max_depth': range (2, 10, 1),
		'n_estimators': range(60, 220, 40),
		'learning_rate': [0.1, 0.01, 0.05]
		}
		grid_search = GridSearchCV(
		estimator=estimator,
		param_grid=parameters,
		scoring = 'roc_auc',
		n_jobs = 10,
		cv = 10,
		verbose=True
		)
		grid_search.fit(X_train, y_train)
		grid_search.best_estimator_
			
			
			 

		Validation method- k-Fold Cross Validation from sklearn.model_selection import KFold from sklearn.model_selection import cross_val_score kfold = KFold(n_splits=10, random_state=7,shuffle=True) results = cross_val_score(xgb, X_train, y_train, cv=kfold) print("Accuracy: %.2f%% (%.2f%%)" % (results.mean()*100, results.std()*100))