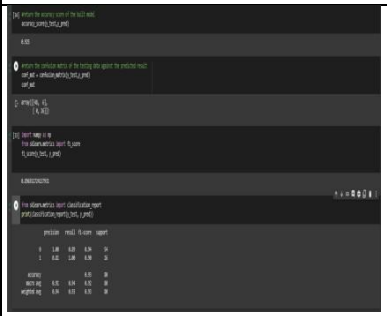
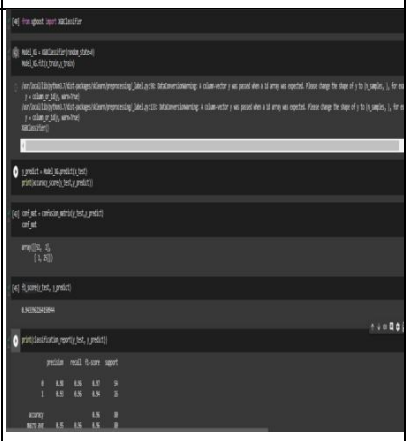
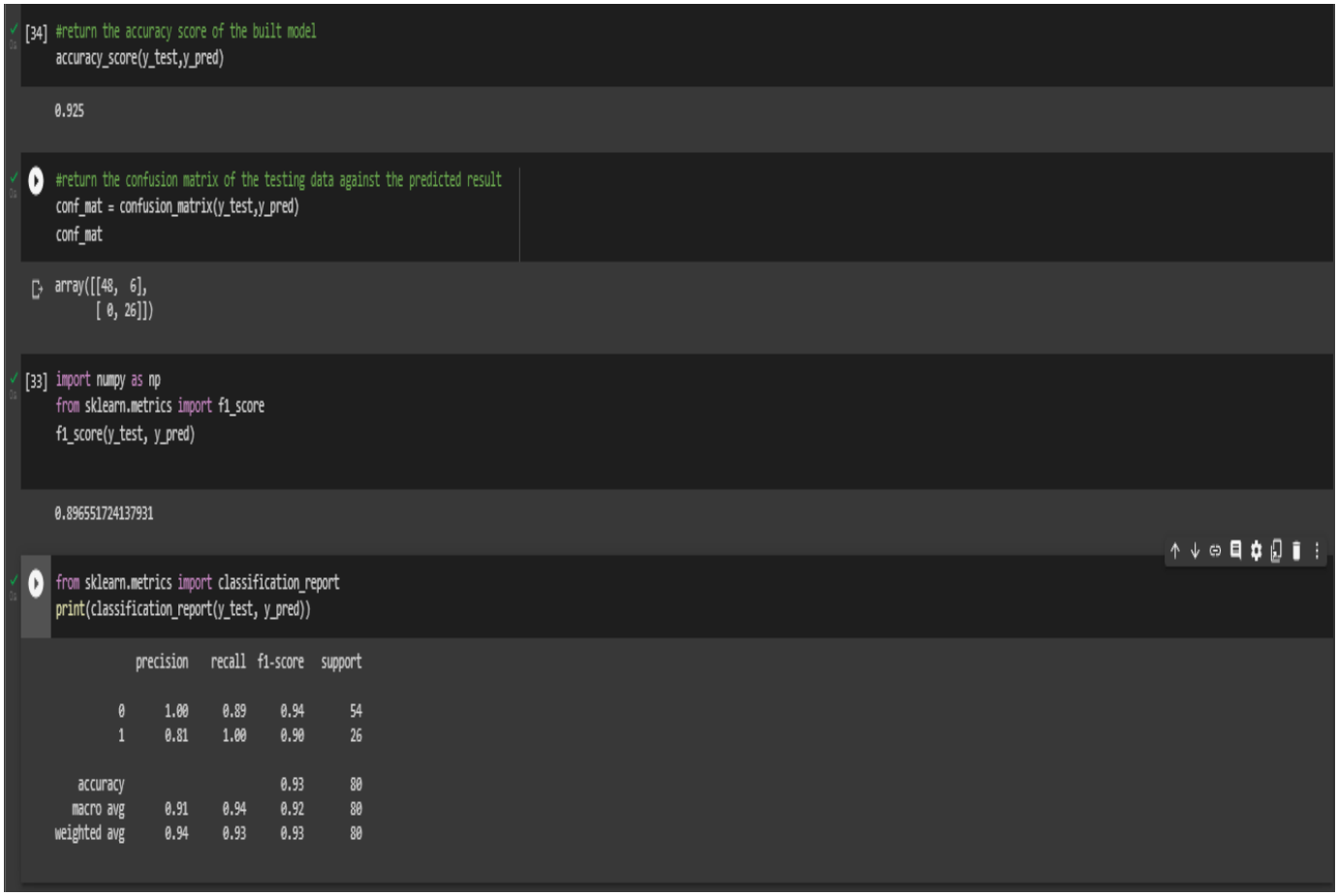
Project Development Phase
Model Performance Test

Date	15November 2022
Team Lead	Swetha S
Team Members	Uma B Vishnu Priya S Yuvashri V
Project Name	Early Detection of Chronic Kidney Disease using Machine Learning

Model Performance Testing:

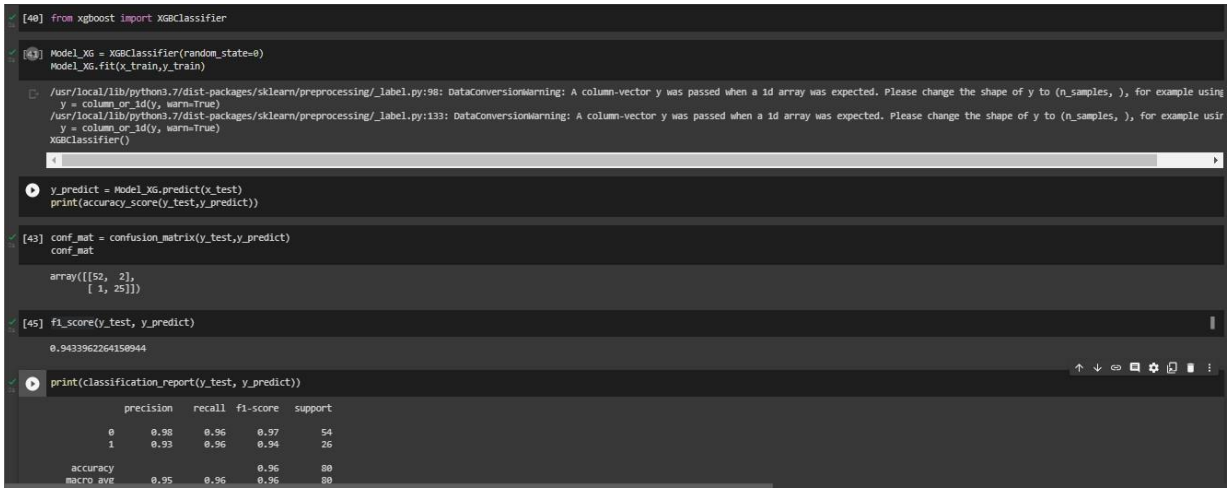
S.No.	Parameter	Values	Screenshot
1.	Metrics	**Classification Model:** Confusion Matrix , F1 Score, Accuracy Score & Classification Report	
2.	Tune the Model	Hyperparameter Tuning - Validation Method -	

1. Metrics



```
[34] #return the accuracy score of the built model
     accuracy_score(y_test,y_pred)

     0.925

     #return the confusion matrix of the testing data against the predicted result
     conf_mat = confusion_matrix(y_test,y_pred)
     conf_mat

     array([[48,  6],
            [ 0, 26]])

[33] import numpy as np
     from sklearn.metrics import f1_score
     f1_score(y_test, y_pred)

     0.896551724137931

     from sklearn.metrics import classification_report
     print(classification_report(y_test, y_pred))

                   precision    recall  f1-score   support

                0       1.00      0.89      0.94        54
                1       0.81      1.00      0.90        26

         accuracy                           0.93        80
        macro avg       0.91      0.94      0.92        80
     weighted avg       0.94      0.93      0.93        80
```

2. Tune the Model



```
[40] from xgboost import XGBClassifier

[ ] Model_XG = XGBClassifier(random_state=0)
    Model_XG.fit(x_train,y_train)

/usr/local/lib/python3.7/dist-packages/sklearn/preprocessing/_label.py:98: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.7/dist-packages/sklearn/preprocessing/_label.py:133: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example usir
  y = column_or_1d(y, warn=True)
XGBClassifier()

y_predict = Model_XG.predict(x_test)
print(accuracy_score(y_test,y_predict))

[43] conf_mat = confusion_matrix(y_test,y_predict)
     conf_mat

array([[52,  2],
       [ 1, 25]])

[45] f1_score(y_test, y_predict)

0.9433962264150944

print(classification_report(y_test, y_predict))

              precision    recall  f1-score   support

           0       0.98      0.96      0.97        54
           1       0.93      0.96      0.94        26

    accuracy                           0.96        80
   macro avg       0.95      0.96      0.96        80
```