

## PROJECT DEVELOPMENT PHASE

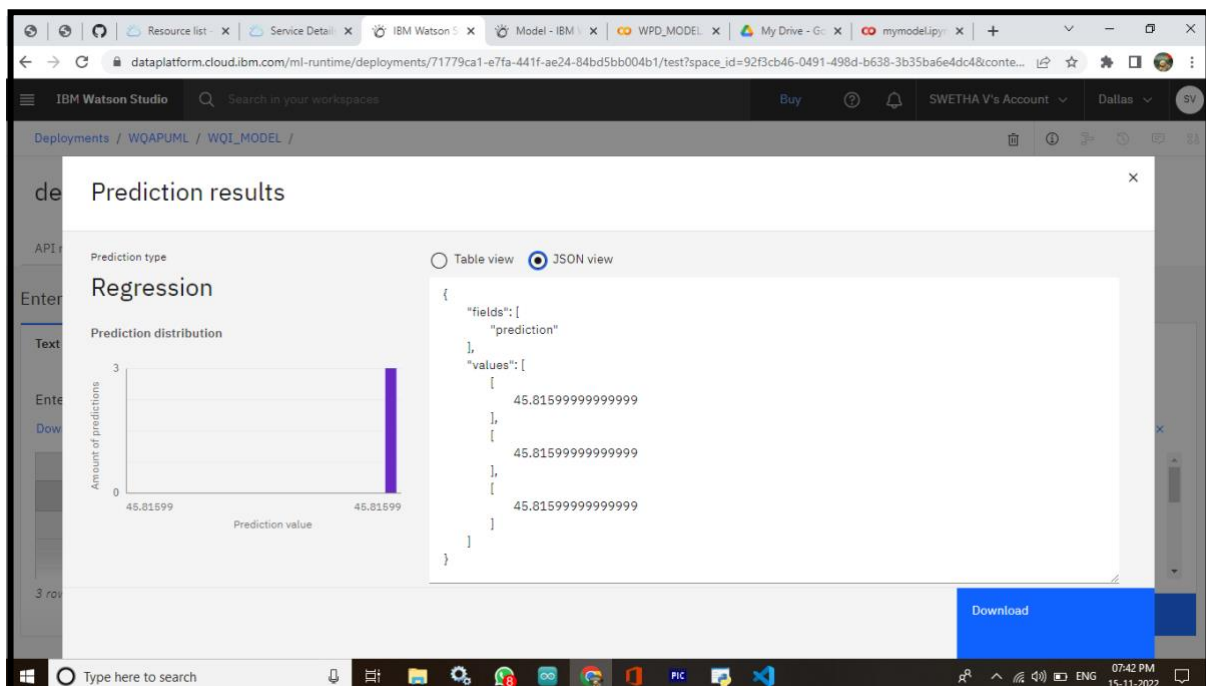
### Performance Testing

Date	10 November 2022
Team ID	PNT2022TMID29722
Project Name	Efficient Water Quality Analysis & Prediction Using Machine learning
Maximum marks	10 Marks

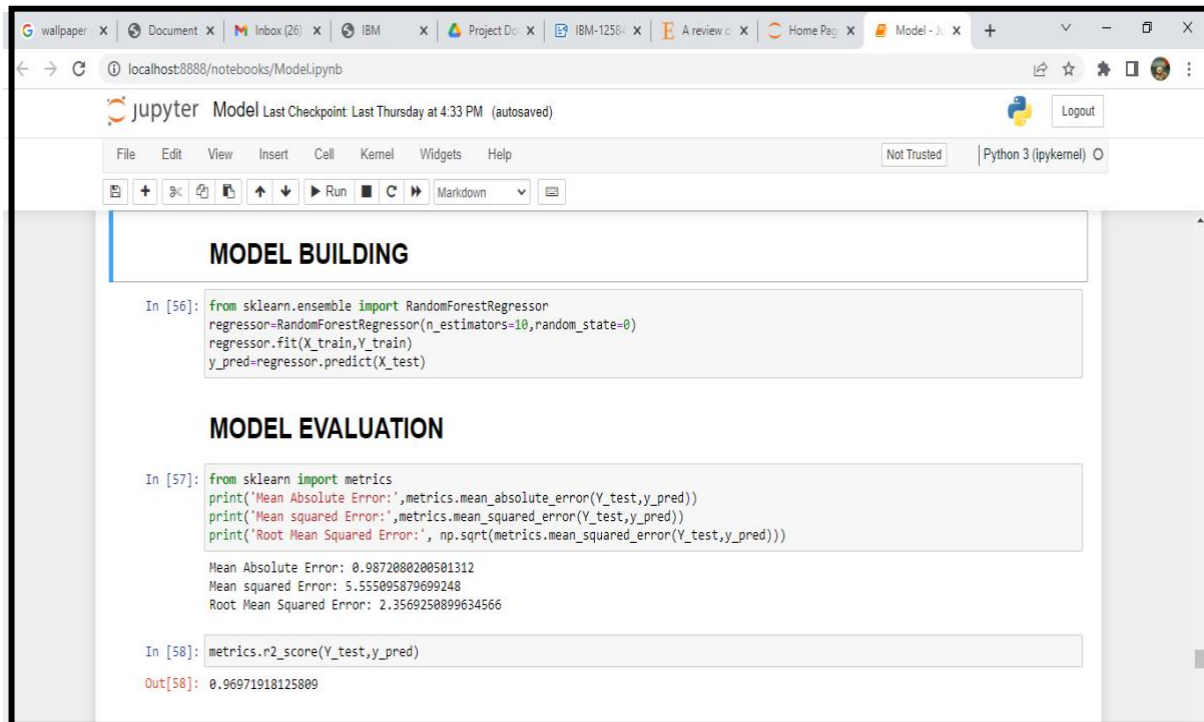
#### Model Performance Testing:

S.NO	Parameter	Values
1.	Metrics	<p><b>Regression Model:</b></p> <p><b>RANDOM FOREST REGRESSOR:</b></p> <p>Random Forest Regression is a supervised learning algorithm that uses ensemble learning method for regression. Ensemble learning method is a technique that combines predictions from multiple machine learning algorithms to make a more accurate prediction than a single model.</p> <p><b>Classification Model:</b></p> <ul style="list-style-type: none"><li>➤ Confusion Metrics</li><li>➤ Accuracy Scoring</li></ul>

#### PREDICTION RESULTS:



# ACCURACY RESULTS:



The screenshot shows a Jupyter Notebook interface in a web browser. The browser's address bar displays 'localhost:8888/notebooks/ModelIpyynb'. The Jupyter interface includes a top bar with the 'jupyter' logo, the notebook name 'Model', and a 'Last Checkpoint' timestamp. Below this is a menu bar with options like 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', 'Widgets', and 'Help'. A toolbar with icons for file operations and execution is also present. The notebook content is divided into two main sections: 'MODEL BUILDING' and 'MODEL EVALUATION'. The 'MODEL BUILDING' section contains a code cell (In [56]) that imports 'RandomForestRegressor' from 'sklearn.ensemble', fits the model to training data, and makes predictions on test data. The 'MODEL EVALUATION' section contains two code cells. The first (In [57]) imports 'metrics' and prints the Mean Absolute Error, Mean Squared Error, and Root Mean Squared Error. The second (In [58]) prints the R-squared score. The output of the second cell (Out[58]) shows an R-squared score of 0.96971918125809.

```
In [56]: from sklearn.ensemble import RandomForestRegressor
regressor=RandomForestRegressor(n_estimators=10,random_state=0)
regressor.fit(X_train,Y_train)
y_pred=regressor.predict(X_test)

MODEL BUILDING

In [57]: from sklearn import metrics
print('Mean Absolute Error:',metrics.mean_absolute_error(Y_test,y_pred))
print('Mean Squared Error:',metrics.mean_squared_error(Y_test,y_pred))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(Y_test,y_pred)))

Mean Absolute Error: 0.9872080200501312
Mean Squared Error: 5.555095879699248
Root Mean Squared Error: 2.3569250899634566

In [58]: metrics.r2_score(Y_test,y_pred)

Out[58]: 0.96971918125809
```