Algorithm and Steps:

1. **Data Preprocessing:**
    - **Load the dataset using Pandas.**
    - **Extract relevant features (`Age`, `EstimatedSalary`) and target variable (`Purchased`).**
    - **Split the data into training and testing sets using `train_test_split`.**
    - **Apply feature scaling using `StandardScaler` for better model performance.**
2. **Model Training:**
    - **Instantiate and train the Logistic Regression model using the `LogisticRegression()` class from scikit-learn.**
    - **Fit the model on the training data.**
3. **Prediction:**
    - **Use the trained model to predict the output on test data.**
    - **Compare predicted and actual values.**
4. **Model Evaluation:**
    - **Generate a Confusion Matrix to get values of TP, TN, FP, FN.**
    - **Calculate key metrics:**
        - **Accuracy = (TP + TN) / Total**
        - **Error Rate = (FP + FN) / Total**
        - **Precision = TP / (TP + FP)**
        - **Recall (Sensitivity) = TP / (TP + FN)**

- **Specificity = TN / (TN + FP)**
- **F1 Score = Harmonic mean of Precision and Recall**

5. **Visualization:**
   - **Display confusion matrix using `ConfusionMatrixDisplay`.**
   - **Print the classification report.**

Conclusion:

Logistic Regression effectively performs binary classification on the dataset. The model's performance can be quantitatively assessed through evaluation metrics derived from the confusion matrix. This helps understand how well the model distinguishes between the two classes—purchasing and not purchasing—based on age and estimated salary.