**Project Execution Process :**

**Step 1:** Set Up Your Environment

1. \*Install Python\*: Ensure Python is installed on your system.

2. \*Install Required Libraries\*: Use pip to install necessary libraries. These include TensorFlow, Pandas, NumPy, Scikit-learn, Joblib, Flask, and Matplotlib.

**Step 2:** Prepare the Dataset

1. \*Load the Dataset\*: Obtain the ECG dataset and save it as ECG-Dataset.csv.

2. \*Preprocess the Data\*: Ensure your dataset is preprocessed correctly and has a binary target column named target.

**Step 3:** Train the Model

1. \*Split the Data\*: Divide the dataset into training and testing sets (80% training, 20% testing).

2. \*Standardize the Data\*: Use a scaler to standardize the data to have a mean of 0 and a standard deviation of 1.

3. \*Build the Neural Network Model\*: Construct a deep learning model using TensorFlow.

4. \*Compile and Train the Model\*: Compile the model with the appropriate optimizer and loss function, then train it with the training data, incorporating early stopping and model checkpoint callbacks.

**Step 4:** Save the Model and Scaler

1. \*Save the Trained Model\*: Save the trained model to a file named heart\_disease\_model.keras.

2. \*Save the Scaler\*: Save the scaler used for data standardization to a file named scaler.save.

**Step 5:** Evaluate the Model

1. \*Evaluate on Test Data\*: Test the model on the testing dataset and obtain the accuracy.

2. \*Save the Accuracy\*: Save the accuracy to a file for future reference.

**Step 6:** Set Up Flask for Web Interface

1. \*Create Flask Application\*: Set up a Flask application to create a web interface for making predictions.

2. \*Define Routes\*: Define routes for displaying an input form, processing the form data to make predictions, and showing the results.

3. \*Create HTML Templates\*: Create HTML templates for the input form, result display, and performance plots.

**Step 7:** Run the Flask Application

1. \*Start the Local Server\*: Run the Flask application to start a local web server.

2. \*Access the Application\*: Open a web browser and navigate to http://127.0.0.1:5000.

**Step 8:** Enter Inputs and Make Predictions

1. \*Fill the Input Form\*: Enter the required input values for prediction on the web interface.

2. \*Submit the Form\*: Submit the form to make a prediction.

3. \*View the Results\*: The application will display the prediction result on the results page.

**Step 9:** Display Performance Plots

1. \*Access the Plots Page\*: On the results page, click the button to view performance plots.

2. \*View Accuracy and Loss\*: The plots page will show the model's accuracy and loss over the training epochs.

**Summary:**

- \*Training and Testing\*: You trained and tested a deep learning model to predict heart disease based on ECG data.

- \*Model Deployment\*: You deployed the model using a Flask web application, allowing users to input data and receive predictions.

- \*Result Visualization\*: The application also provides visualization of the model's performance through plots.

By following these steps, you can run the project, input test data, obtain predictions, and visualize the model's performance.