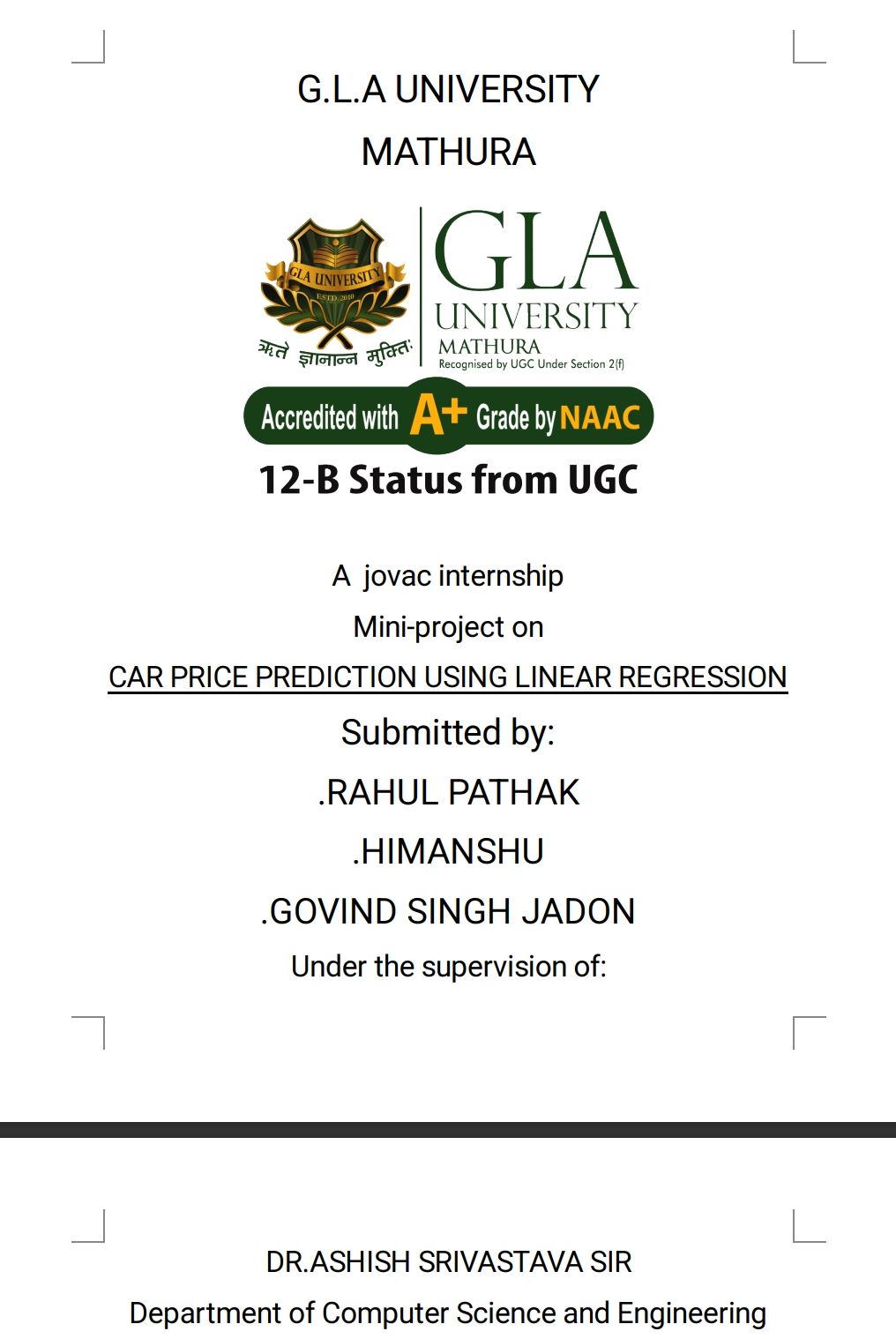
****

**CAR PRICE PREDICTION USING THE LINEAR REGERESSION**

**Project Description:**

**Objective: The goal of this analysis is to predict car prices based on various features of the cars using a linear regression model. And the final selling price is to be calculated.**

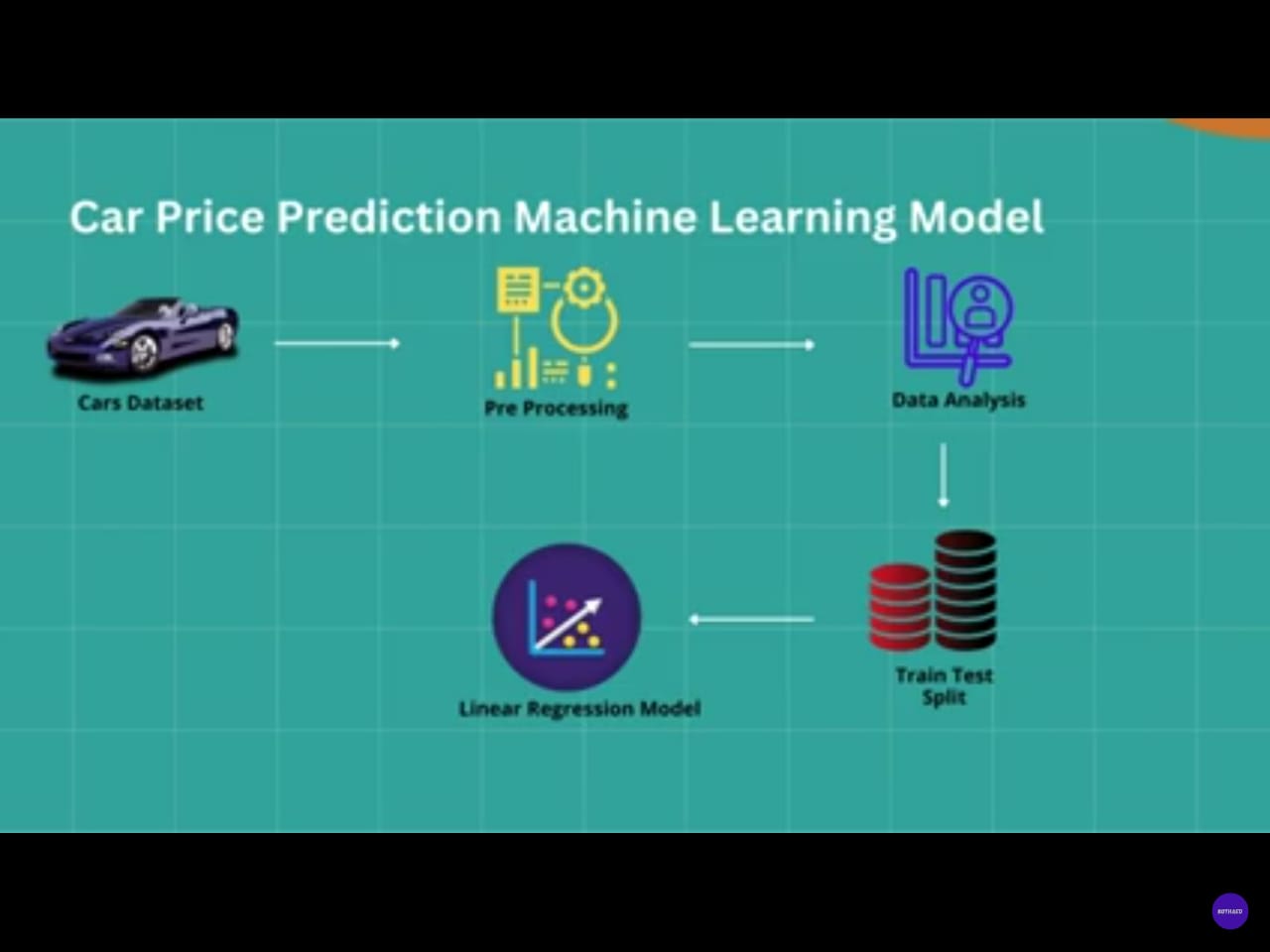
**Dataset: The dataset comprises various features of cars, including**

| **name** | **year** | **Selling price** | **Km driven** | **fuel** | **Seller type** | **transmission** | **owner** | **mileage** | **engine** | **Max**  **power** | **torque** | **seats** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

**. The target variable for prediction is the car’s selling price.**

****

**Methodology: The project employs linear regression, a fundamental machine learning technique, to model the relationship between car features and prices. The dataset is preprocessed, and the model is trained and evaluated to ensure its predictive accuracy and generalizability.**

****

**Workflow**

* **Important libraries**

1. **Pip install pandas**
2. **Pip install numpy**
3. **Pip install pickle**
4. **Pip install Scikit-learn**
5. **Pip install streamlit**

* **Data Collection**

1. **Source: Obtain the dataset from a reliable source (e.g., CSV file).**

**Cardetails.csv**

1. **Content: The dataset includes features like name, fuel type, seller type, km driven, transmission type, mileage, max-power and car specifications.**

* **Data Preprocessing**

1. **Exploration: Inspect the dataset for initial insights, including dimensions, data types, and missing values.**
2. **Handling Missing Values: Address any missing data appropriately.**
3. **Encoding Categorical Variables: Convert categorical features (e.g., fuel type, seller type, transmission) into numerical values using encoding techniques.**
4. **Feature Selection: Drop irrelevant features and separate features (X) from the target variable (Y).**

* **Model Development**

1. **Train-Test Split: Divide the dataset into training and testing sets (e.g., 90% training, 10% testing) to evaluate model performance.**
2. **Training: Apply linear regression to the training data**
3. **Evaluation: the car price when we change any of the feature accordingly.**

* **Model Evaluation**

1. **Training Data: Analyze model performance on training data and visualize actual vs. predicted prices.**
2. **Testing Data: Evaluate the model on test data to check its ability to generalize to new data. Compare actual vs. predicted prices.**

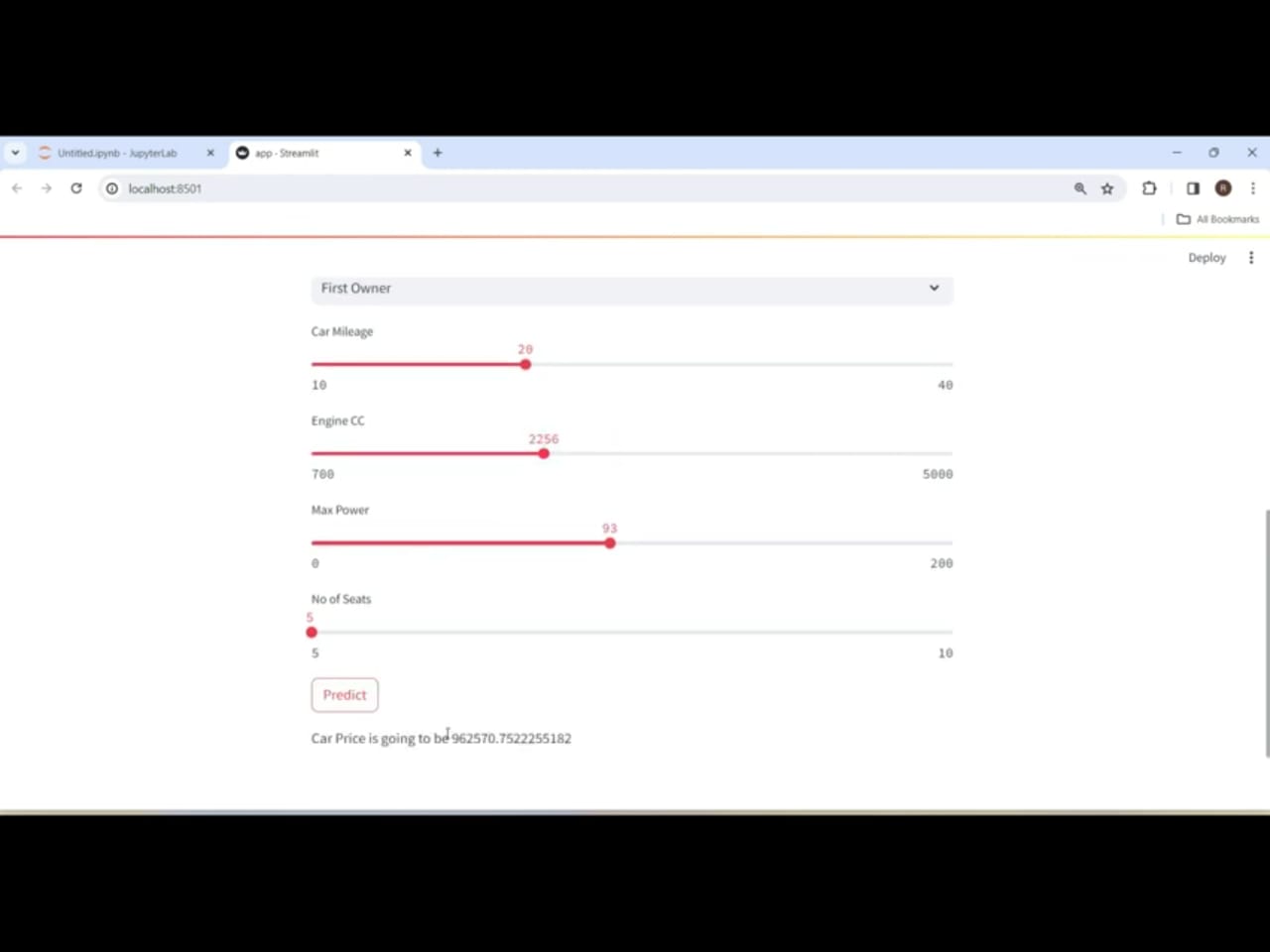
* **Deploy the result**

1. **Deploy the result and see the result by running the “streamlit” by streamlit app.py.**

* **Results Analysis**

**Feature Importance: Identify significant features influencing car prices.**

**Future Work: Suggest improvements, such as incorporating additional features or exploring advanced models**

**Output:**

**References:**

[**www.kaggle.com**](http://www.kaggle.com)

[**www.jupyter.com**](http://www.jupyter.com)

[**www.codingblocks.com**](http://www.codingblocks.com)