Group 13

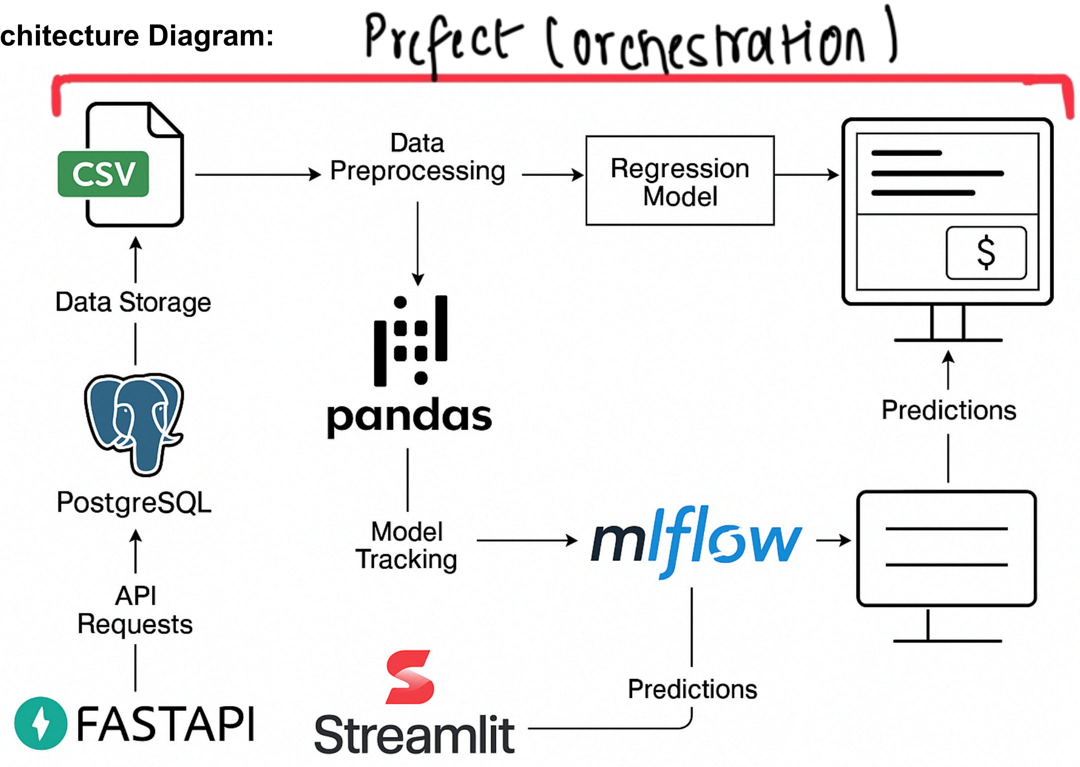
Title: Used Car Price Prediction, Search and Analysis

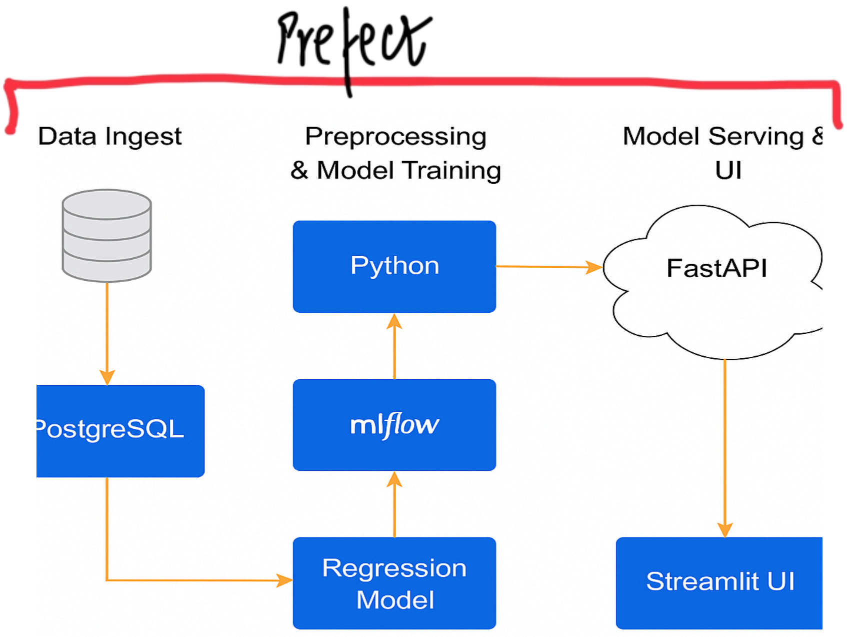
Goal: The aim of the project is to build an end-to-end data engineering pipeline to predict the price of used cars by preprocessing the data, then train machine learning models for price prediction, storage of data using structured datasets and giving output with analytics and an interactive UI.

Domain/Theme: Data Engineering, Machine Learning and MLOps

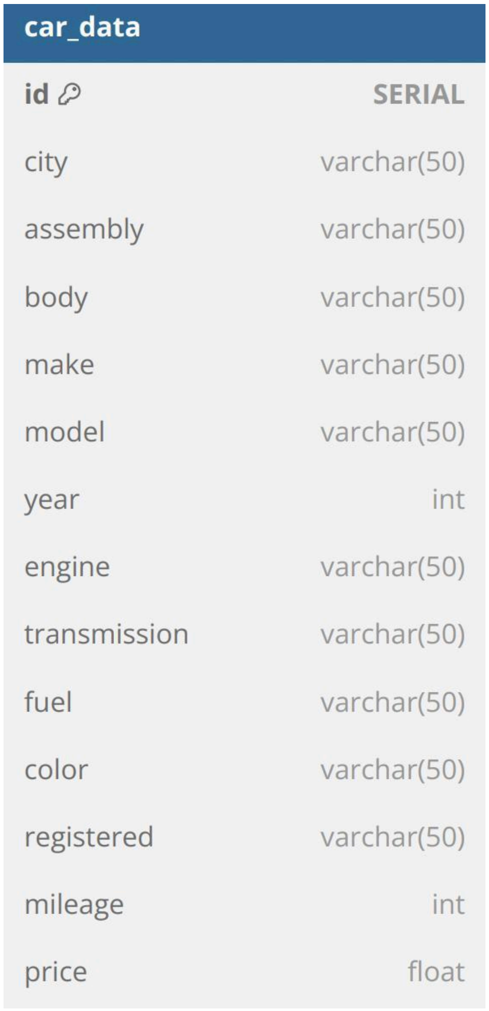
Data:
Source: PakWheels used cars listings data..
Type: Structured data.
Attributes: City, Make, Model, Year, Engine, Fuel, Mileage, Transmission, Price, etc.
Additions: Synthetic data added for robustness and variation.

Architecture Diagram:





Schema Diagram:



car_data	
id 🔑	**SERIAL**
city	varchar(50)
assembly	varchar(50)
body	varchar(50)
make	varchar(50)
model	varchar(50)
year	int
engine	varchar(50)
transmission	varchar(50)
fuel	varchar(50)
color	varchar(50)
registered	varchar(50)
mileage	int
price	float

Data Engineering Lifecycle: First we took data in raw .csv format. Then to ingest that data we used Python and Pandas library to parse the CSV file and that data was stored in PostgreSQL database. The data had some inconsistencies and values that were redundant or needed processing. The transformation and cleaning of data was done in python. Cleaning and preprocessing of the data was done by preparing a DataFrame of car listings for training by handling missing values, encoding of categorical features, scaling of numeric features and creation of some new features. For the missing year of the car we estimated the car's age. The assembly of cars was encoded as local or imported. Fills missing fuel with the most common value and maps fuel types to numerical values. Applied ordinal encoding on transmission. After this we ran multiple iterations of regression models such as linear and polynomial regression with varying polynomial degrees and for each iteration ran and removed and dropped columns that didn’t give accuracy during validation data testing. This helped us finalise that polynomial regression of degree 2 gave lowest mean square error and highest R2 value. The model was monitored using MLflow and use of scikit-learn to implement. Then we used a lightweight REST API server of FastAPI to serve the model to our UI deployment of Streamlit where we created an interactive interface.

Render link: *https://car-price-1.onrender.com/*

Github repository link: *https://github.com/eeman122/car_price.git*

Names	*Roll No.*	*Contributions*
M. Annus Shabbir	24280015	Machine Learning model testing FastAPI with ML model for predictions
Kashaf Gohar	24280009	Machine Learning model testing Streamlit UI
Eeman Adnan	24280022	Machine Learning model testing FastAPI with DB for result fetching Render Deployment (frontend and backend) Orchestration