AI601: DATA ENGINEERING FOR AI SYSTEMS

Used Car Price Prediction, Search and Analysis

Kashaf Gohar | Eeman Adnan | Muhammad Annus Shabbir

PROJECT OVERVIEW

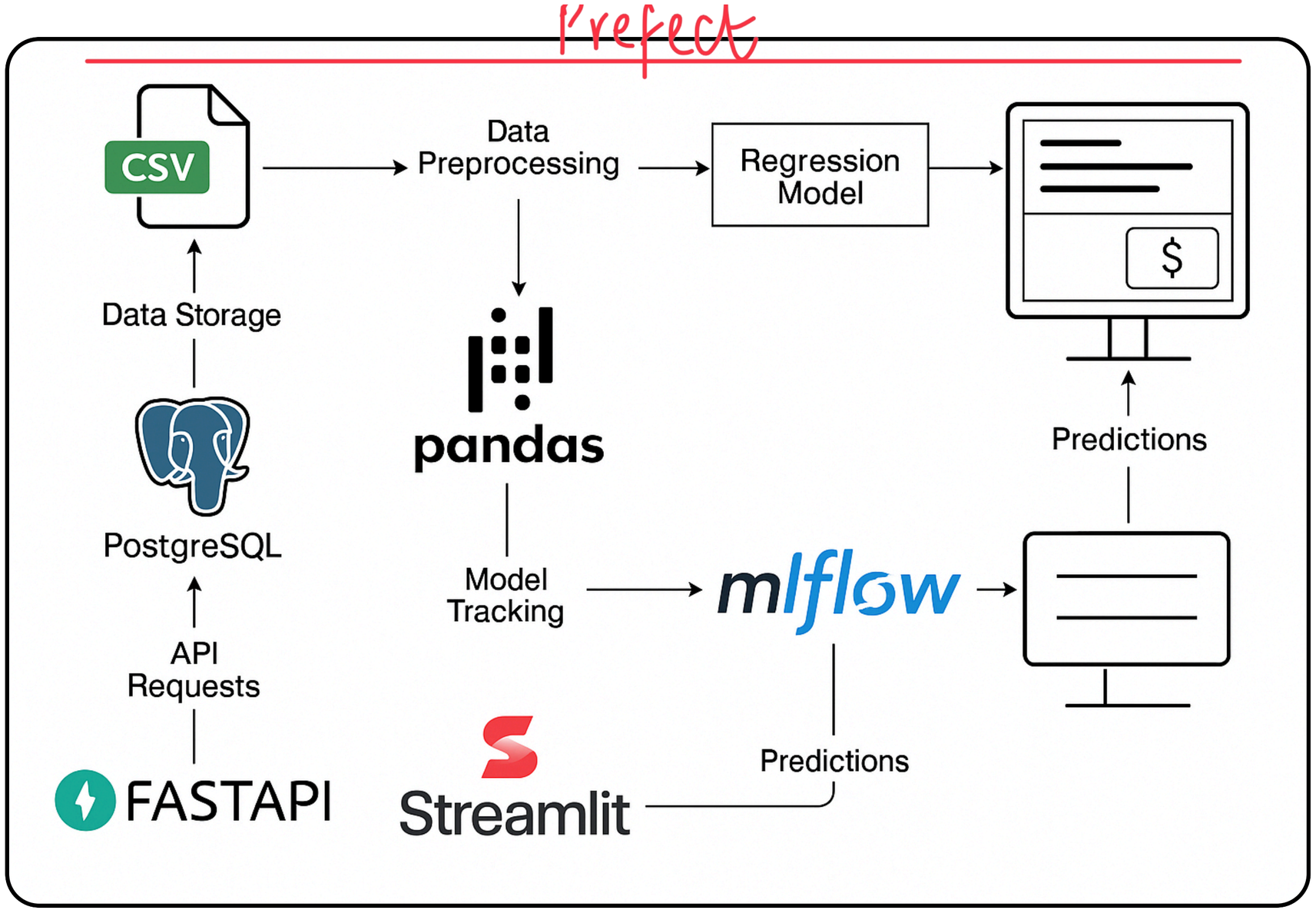
Goal: The aim of the project is to build an end-to-end data engineering pipeline to predict the price of used cars by preprocessing the data, then train machine learning models for price prediction, storage of data using structured datasets and giving output with analytics and an interactive UI.

Domain/Theme: Data Engineering, Machine Learning and MLOps

Data:

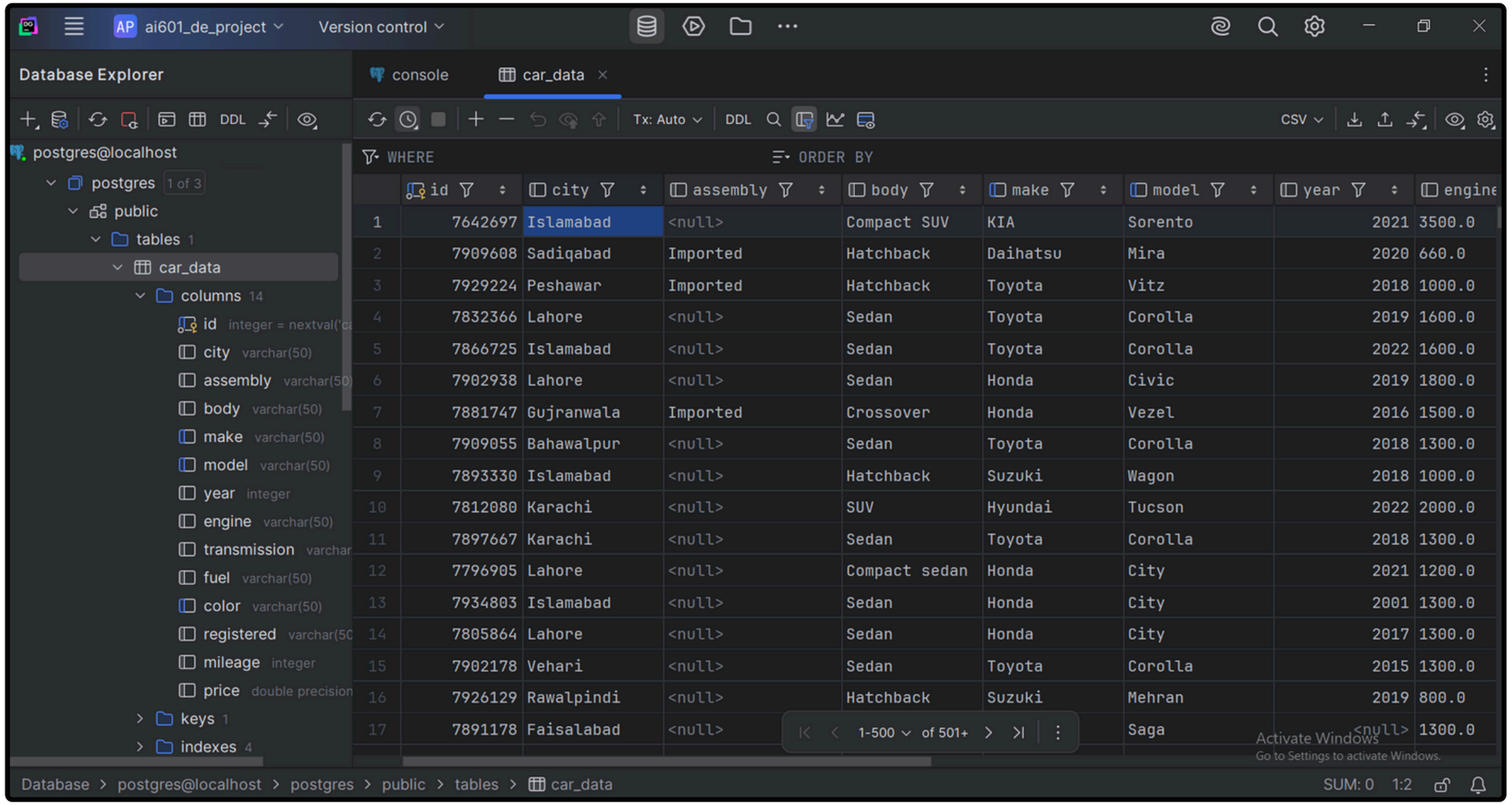
Source: PakWheels used cars listings data..
Type: Structured data.
Attributes: City, Make, Model, Year, Engine, Fuel, Mileage, Transmission, Price, etc.
Additions: Synthetic data added for robustness and variation.

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



AI Usage

- Synthetic data generation to add to data source
- FastAPI integration with Database

CHALLENGES

- ***Database Connection and Integration***

Establishing a reliable and persistent connection between FastAPI and PostgreSQL was challenging, especially during local development and Docker deployment.

- ***FastAPI Model Serving***

Integrating MLflow models into FastAPI and ensuring consistent model performance was a non-trivial task. The MLflow models loaded dynamically using mlflow.sklearn.load_model() which caused delays on initial API call (cold start problem) and there was dependency mismatch (e.g., NumPy or scikit-learn versions) during loading trained models.

- ***API–Frontend Integration (FastAPI + Streamlit)***

Connecting Streamlit with FastAPI (via requests.post() and requests.get()) caused CORS issues and broken API endpoints during testing. Localhost ports (8000 for FastAPI and 8501 for Streamlit) had to be configured manually. Streamlit displayed generic errors when API call failed without helpful debug info.

LEARNINGS

- End-to-End ML and Data Engineering Pipeline Architecture
- Model Tracking with MLflow
- Database Management
- Frontend–Backend Communication
- API Development with FastAPI

DEMO