**Subject:** Capstone Machine Learning Deployment Assignment

**Project :** Car Price Prediction API with FastAPI & Google Cloud Run

**1. Executive Summary**

This project aims to build and deploy a machine learning model to predict car prices based on customer demographics and vehicle specifications. The solution uses Python's FastAPI framework for serving predictions and is deployed using Google Cloud Run for scalability and availability. The model is trained on a car sales dataset, preprocessed for quality, and exposed via a REST API.

**2. Problem Statement & Dataset Description**

**Problem:** Predict the price of a car based on inputs like age, annual income, engine type, company, etc.

**Dataset Source:** Kaggle - [Car Sales Dataset](https://www.kaggle.com/datasets/missionjee/car-sales-report)

**Features Used:**

* Age
* Annual Income
* Company
* Model
* Engine
* Transmission
* Dealer Region

**Target Variable:**

* Price ($)



**3. Model Development Process**

**Preprocessing:**

* Missing value imputation
* Label encoding for categorical variables
* Feature scaling with StandardScaler

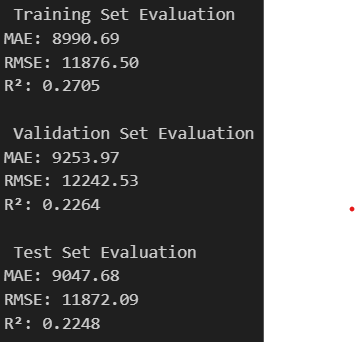
**Model:**

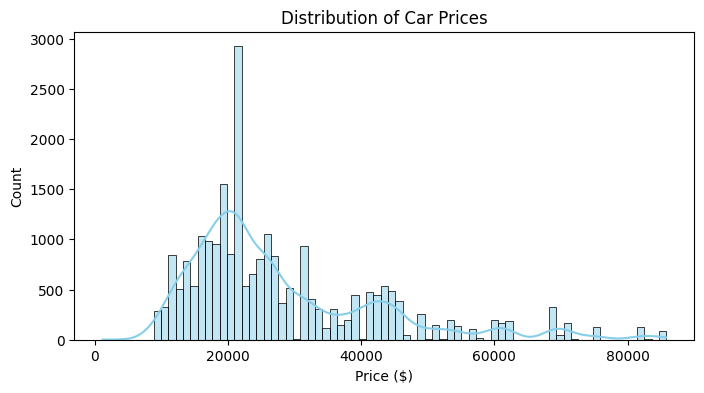
* RandomForestRegressor (with GridSearchCV)

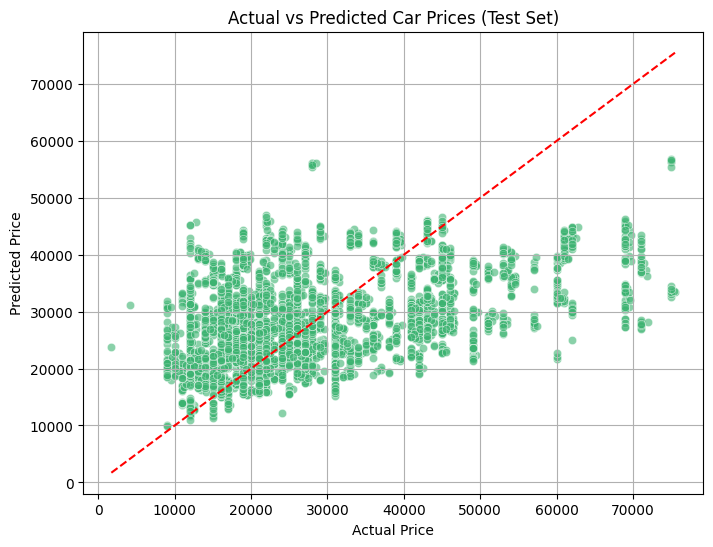
**Evaluation Metrics:**

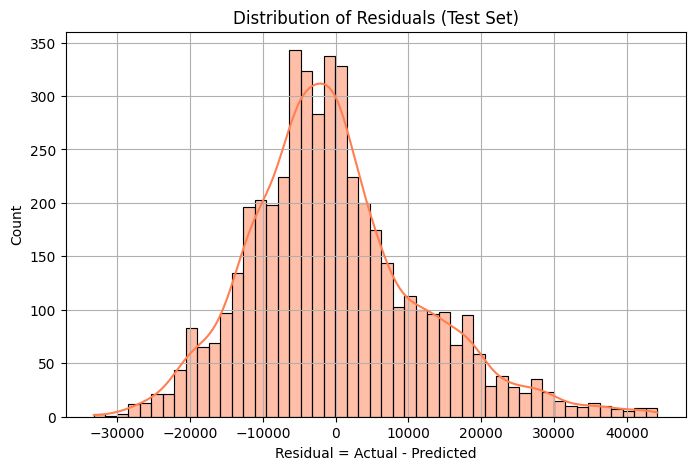
* R² Score
* Mean Absolute Error (MAE)

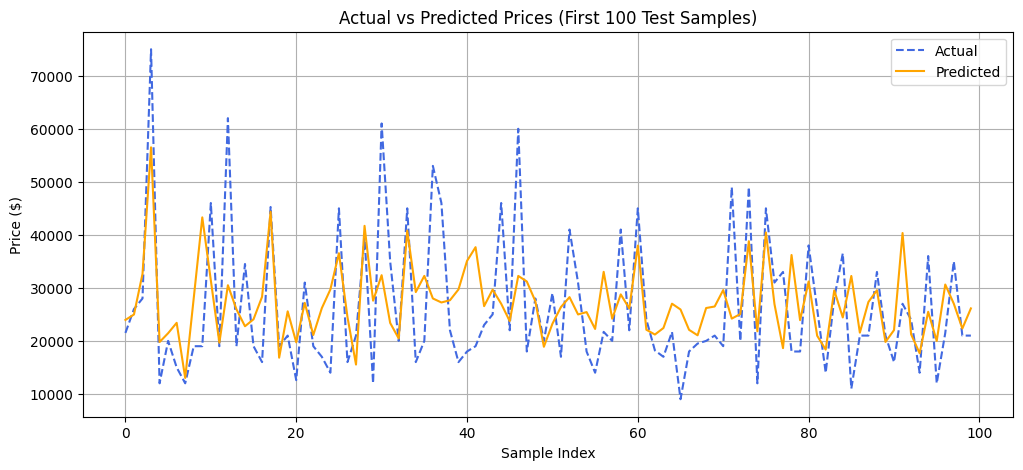
**Performance:**

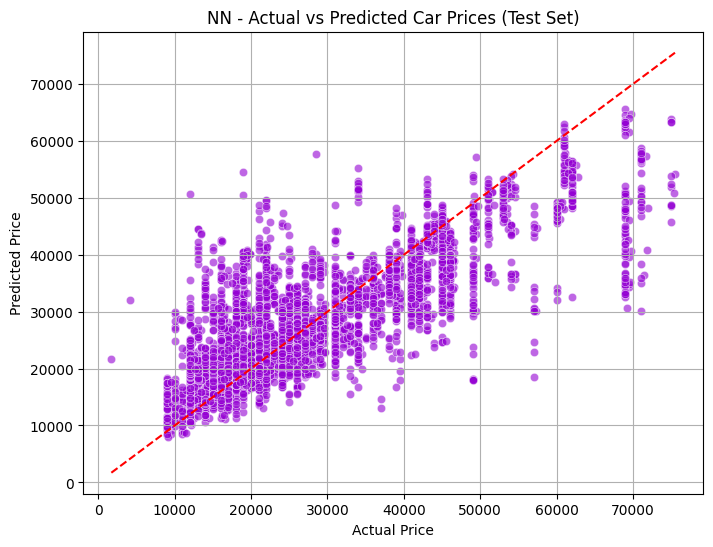


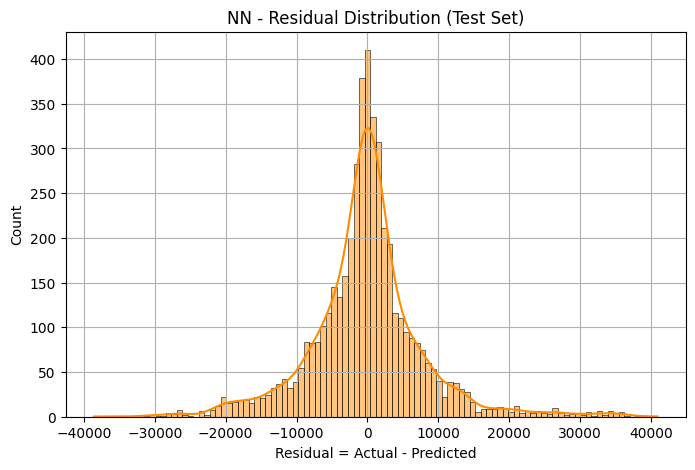


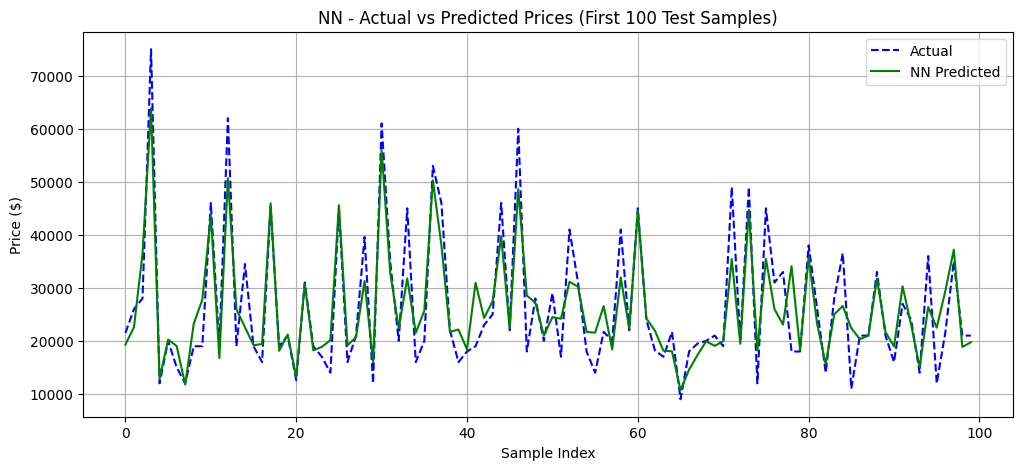


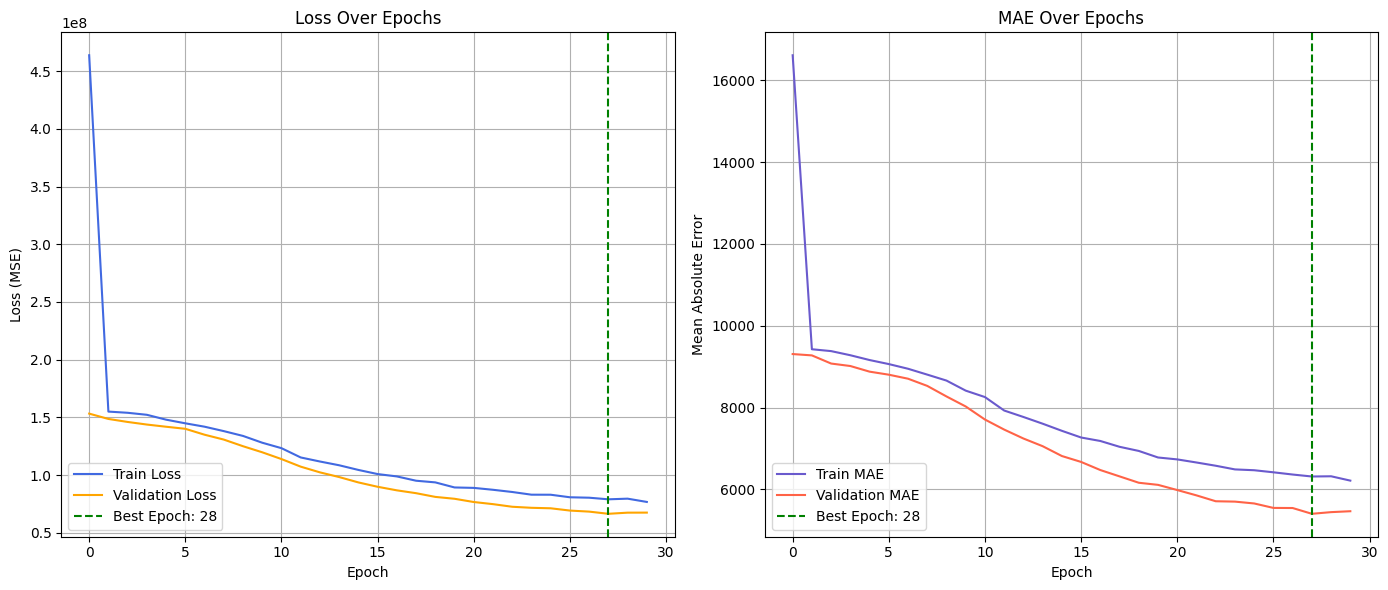












**4. Deployment Architecture**

**Tools & Services:**

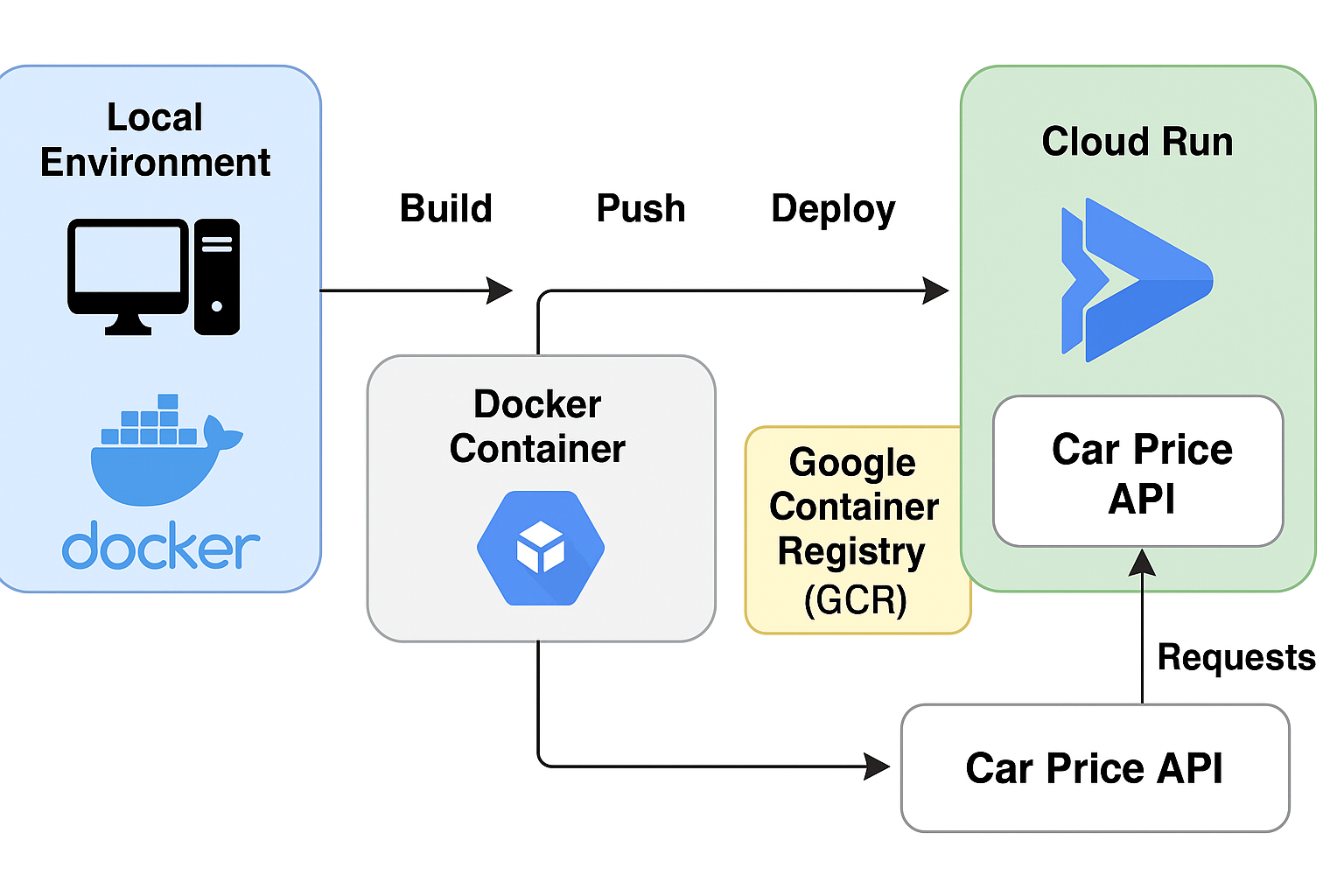
* FastAPI
* Docker
* Google Cloud Build
* Google Cloud Run
* Cloud Container Registry

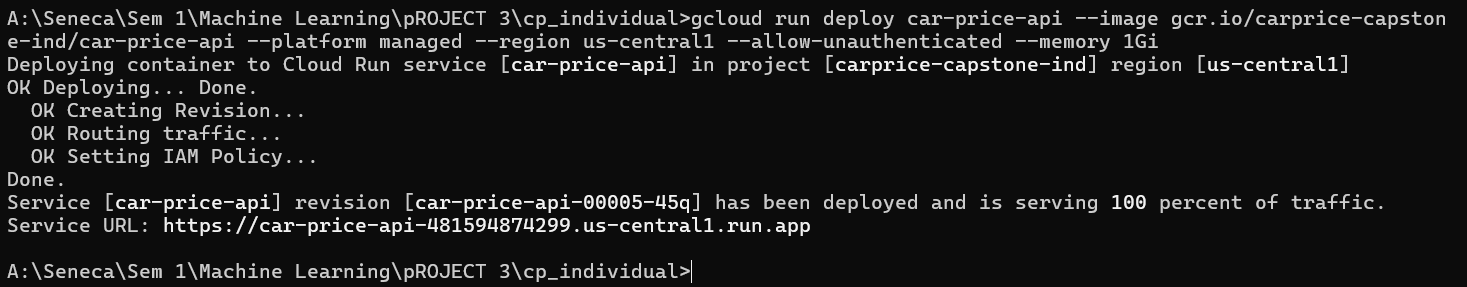
**Deployment Steps:**

1. Train and pickle model
2. Create FastAPI app in main.py
3. Write Dockerfile
4. Build and push image to GCR
5. Deploy to Cloud Run with memory set to 1Gi

**Live API Endpoint:**  
https://car-price-api-481594874299.us-central1.run.app/docs

Diagram showing architecture (local -> Docker -> GCR -> Cloud Run)



Screenshot of successful deployment confirmation in terminal  


1. **Challenges & Solutions**

**Challenge:** Memory limit exceeded 512MiB

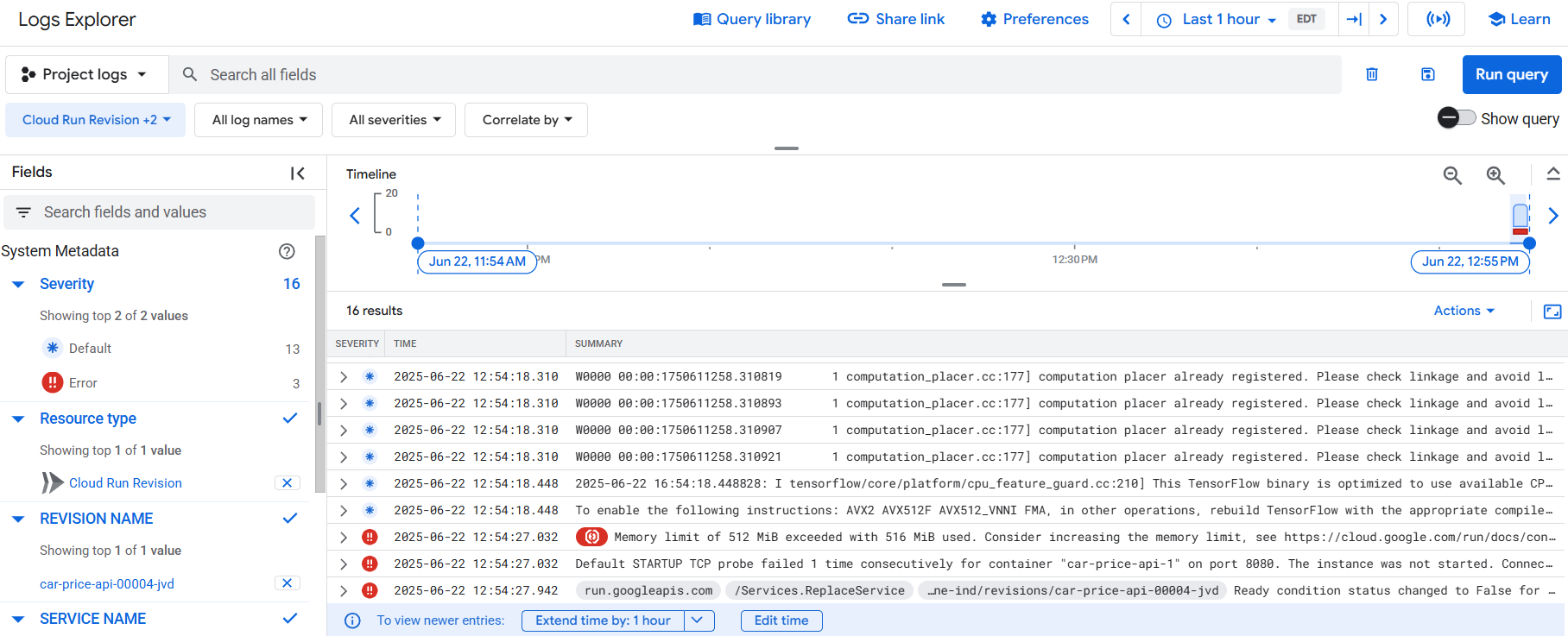
* **Solution:** Increased to –memory 1Gi in deploy command

**Challenge:** App not listening on port 8080

* **Solution:** Ensured FastAPI uses port=8080 in CMD and EXPOSE in Dockerfile

**Challenge:** Timeout errors

* **Solution:** Verified logs and updated startupProbe configuration if needed



Error logs or deployment failure messages from Cloud Console

1. **Conclusion & Future Work**

This project demonstrates the ability to train, serve, and deploy an ML model using modern cloud tools. The app is now scalable and available publicly via REST API. Future enhancements include:

* Adding authentication
* UI dashboard for inputs
* Training on additional features like fuel type, mileage, etc.

Working screenshot of API on Swagger UI or Postman.

