Quick Start Guide

T316 – Capstone Project - AutoGuru

A Report
Written For

AutoGuru

Prepared by
Alvin Luu

Submitted in partial fulfillment of project requirements for
QUT - T316 - Capstone Project - AutoGuru

Document Control

Document Title	Quick Start Guide
Project	Vantage
Revision	A
Date	22/09/25
Prepared By	Alvin Luu
Checked By	JL, DH, TD
Approved By	

Revision History

Revision	Date	Author	Description	Approved
A	22/09/2025	Alvin Luu	Prelim Issue	

Table of Contents

1 Definitions......3

1.1 Glossary......3

3 Document Purpose......4

4 Scope......4

5 Quick Start Guide......4

6 Requirements......4

6.1 Hardware Requirements......4

6.2 Software Requirements......5

7 Cloning the repository......5

8 Environment Configuration......6

8.1.1 Backend and Model Training Configuration......6

8.1.2 Front End......7

9 Operation......7

9.1 Training Models......7

9.2 Running the Backend......8

9.3 Running the Frontend......8

10 Ingress and Egress......9

1 Definitions

1.1 Glossary

Term	Definition
Mean	The average of the data set calculated by summing all values then dividing by count of those values.
Median	The middle value within the data set that has been ordered from least to greatest.
IQR Low	The value at the 25th percentile of the data set, point at which less than 75% of the data falls.
IQR High	The value at the 75th percentile of the data set, point at which 75% of the data falls.
Within IQR	A true or false statement that checks if the predicted price in within the low and high IQR range.
Z from Median	Z score is relative to the median and measures how far a data point is from the mean
Confidence	The qualitative assessment of how much trust can be placed on the outputted predicted price. This is calculated based on the performance of previously mentioned metrics.

3 Document Purpose

This document provides clear and streamlined instructions for implementing and operating the Project Vantage solution. It is designed to guide users with no prior experience to successful operation in the shortest possible time. Rather than serving as a comprehensive manual, this guide focuses on the essential steps and requirements needed to quickly set up and run the solution.

4 Scope

This document provides clear instructions for the initial setup of the solution, including all necessary hardware and software requirements. It also outlines the environment configuration process and specifies the required input data locations. The setup instructions in this guide apply exclusively to the Windows operating system. While not intended as a comprehensive manual or full reference, this guide delivers a streamlined process focused on listing the technical requirements and preparing the environment to run the Project Vantage solution.

5 Quick Start Guide

To run the final developed solution, numerous requirements must be met alongside an environment configuration. A step-by-step process is outlined below with additional rules to ensure a smooth implementation process.

6 Requirements

6.1 Hardware Requirements

While powerful hardware is not required to run the front end or back end of the solution locally, it is recommended for model training, as stronger hardware can significantly reduce training times. The models used in this project were trained on a Ryzen 7 7700X processor, yielding the following training durations:

- **Log**: 26 minutes
- **Capped**: 25 minutes
- **Prescribed**: 5 minutes
- **Repaired**: 66 minutes

6.2 Software Requirements

The following software components are required to set up the environment and run the solution:

- Python 3.12 or higher: Required for running the backend services and performing model training. Download
- pip: Python’s package manager, used to install and manage backend and model training dependencies.
- Node.js 22**:** Required for running the React-based frontend of the solution. Download
- npm**:** Node.js package manager, used to install and manage frontend dependencies.
- Git: Required for cloning the repository. Download

Verify Installation:

```
python --version
pip --version
node -v
npm -v
git --version
```

7 Cloning the Repository and Downloading Data

Before setting up the environment, you need to clone the Project Vantage repository to your local machine. Please refer to the README at https://github.com/tdela12/CAPSTONE_T316# for a comprehensive breakdown of the repository and structure.

Steps to clone:

1. Open a Command Prompt terminal (can be done on other terminals however some commands may differ)
2. Navigate to folder where you want to store the project:

```
cd C:\path\to\your\desired\folder
```

3. Clone the repository using the provided URL:

```
git clone https://github.com/tdela12/CAPSTONE_T316.git
```

4. Navigate to the cloned repository:

```
cd CAPSTONE_T316
```

5. Verify that the repository has been cloned successfully by listing the files

```
dir
```

6. Navigate to Preprocessing directory

```
cd preprocessing
```

7. Create data directory

```
mkdir data
```

8. Download the following data sets from the [Goolge Drive]
   - cpi_data.csv
   - mid_data.xlsx
   - registration_data.csv
   - ticket_data.csv
9. Move the data sets to the new data directory

8 Environment Configuration

With the repository cloned, the following steps can be followed to setup the projects environment

8.1 Backend and Model Training Configuration

Configuration Steps:

1. Open a Command Prompt terminal (can be done on other terminals however some commands may differ)
2. Navigate to cloned repository:

```
cd path\to\CAPSTONE_T316
```

3. Create a virtual python environment :

```
python3 -m venv env
```

4. Activate the environment :

```
.\env\Scripts\activate.bat
```

5. Install the following dependencies: numpy, pandas, scipy, scikit-learn, catboost, matplotlib, shap, jupyterlab, ipykernel, fastapi and uvicorn.

```
pip install -r requirements.txt

or

pip install numpy pandas scipy scikit-learn catboost matplotlib
shap jupyterlab ipykernel fastapi uvicorn openpyxl jupyter pytest
httpx pytest-asyncio
```

8.2 Setting up the data

The raw datasets must be pre-processed before the application is used. To do this:

1. Open a Command Prompt terminal (can be done on other terminals however some commands may differ)
2. Navigate to the preprocessing directory

```
cd preprocessing
```

3. Lauch Jupyter notebooks

```
jupyter notebook
```

4. Open http://localhost:8888/tree in browser
5. Open preprocessing_pipeline.ipynb
6. Run all cells
7. Open create_registration_data.ipynb
8. Run all cells

8.3 Front End

Configuration Steps:

1. Open a Command Prompt terminal (can be done on other terminals however some commands may differ)
2. Navigate to the frontend directory inside the cloned repository:

```
cd path\to\CAPSTONE_T316\frontend
```

3. Install frontend dependencies using npm:

```
npm install
```

4. Verify successful installation by starting the development server:

```
npm run dev
```

   - Visit http://localhost:5173/ to verify if the page loads correctly

9 Operation

9.1 Retraining Models

The notebooks used to train and create the models used in this project have been provided. If necessary, the models can be retrained and updated using these notebooks.

1. Open a terminal (Command Prompt, PowerShell, or Git Bash)

2. Navigate to notebook directory inside the cloned repository:

```
cd path\to\CAPSTONE_T316\training_notebooks
```

3. Launch Jupyter Notebook

```
jupyter notebook
```

4. Open the notebook for the model you want to train:
   Example files:
   - train_log_model.ipynb
   - train_capped_model.ipynb
   - train_prescribed_model.ipynb
   - train_repaired_model.ipynb
5. Verify the input data paths in the notebook cells:
   - Ensure the notebook points to the correct folders for training datasets.
   - Update paths if necessary.
6. Run all cells in the notebook to start training:
   - In Jupyter, click Kernel → Restart & Run All, or Cell → Run All.
   - The training process will begin, and progress/output will be displayed in the notebook cells.
   - The retrained model will be saved to the models/ directory

9.2 Running the Backend

1. Open a Command Prompt terminal (can be done on other terminals however some commands may differ)
2. Navigate to the cloned repository:

```
cd path\to\CAPSTONE_T316
```

3. Activate the virtual environment:

```
.\env\Scripts\activate.bat
```

4. Navigate to the backend directory cloned repository:

```
cd backend
```

5. Run the server:

```
uvicorn server:app --reload
```

6. Verify Server is running by accessing swagger docs at http://127.0.0.1:8000/docs

9.3 Running the Frontend

1. Open a Command Prompt terminal (can be done on other terminals however some commands may differ)

2. Navigate to the frontend directory inside the cloned repository:

```
cd path\to\CAPSTONE_T316\frontend
```

3. Run the development server

```
npm run dev
```

4. Interact with the front end at http://localhost:5173/

9.4 Running Unit Tests

1. Open a Command Prompt terminal (can be done on other terminals however some commands may differ)
2. Navigate to the cloned repository:

```
cd path\to\CAPSTONE_T316
```

3. Activate the virtual environment:

```
.\env\Scripts\activate.bat
```

4. Navigate to the backend directory cloned repository:

```
cd backend
```

5. Run the test suite

```
pytest -v
```

10 Ingress and Egress

To view the ingress and egress of the solution model, please refer to the swagger documented API. This can be accessed at http://127.0.0.1:8000/docs when running the backend.