AI Inventory Prediction & Restocking Agent

By :
Prutha Trivedi
Preeti Sanjay Chougule

Table of Contents

- Introduction
- Core Features
- System Workflow Overview
- ML Pipeline Execution
- API Backend - Fast API
- Streamlit UI Launch
- Dashboard Home View
- Forecast Visualization
- Restock Alerts & Recommendation
- Tech Stack

Introduction

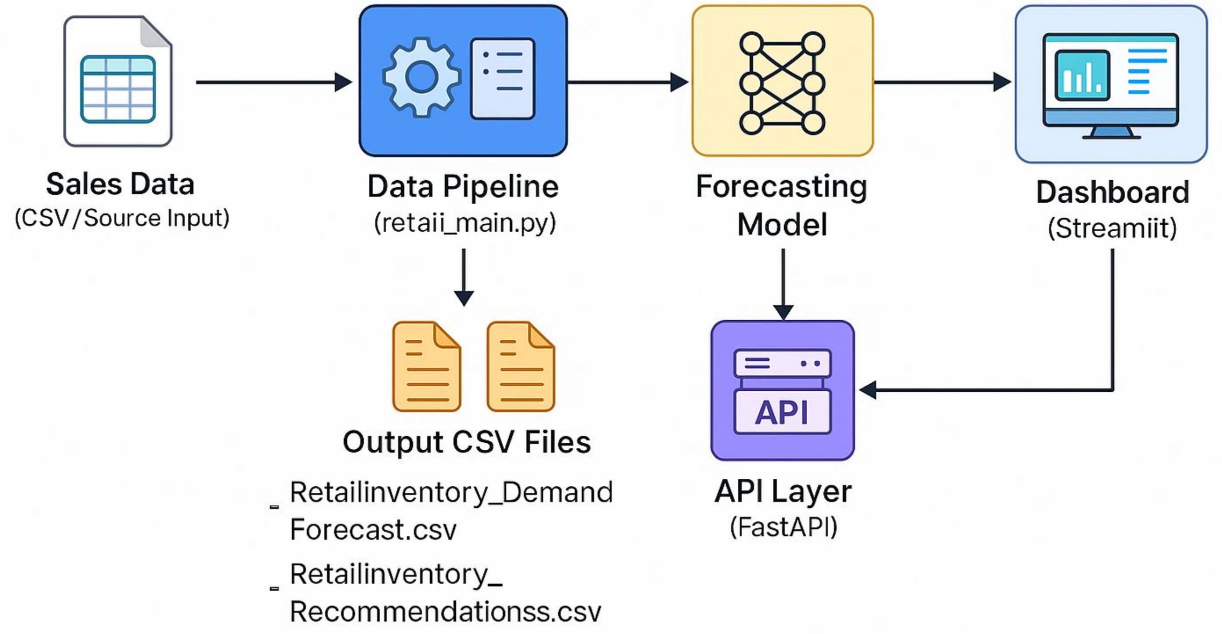
Retail businesses constantly struggle with inefficient inventory management processes, where manual tracking, inaccurate forecasts and delayed restocking decisions often lead to stockouts, excess inventory and operational bottlenecks. These challenges directly affect profitability, customer satisfaction and supply chain reliability. Traditional tools fail to provide real-time visibility or predictive insights, leaving businesses reactive rather than strategic in their inventory decisions.

To address this persistent business problem our project introduces an AI-Driven Inventory Prediction & Restocking Agent that leverages machine learning and Generative AI principles to forecast demand, identify low-stock risks early and recommend optimal reorder quantities. By integrating a complete data engineering pipeline, an automated forecasting model, an API service layer and a Streamlit dashboard, the system transforms manual inventory workflows into a predictive, autonomous and data-driven process. This solution not only enhances operational efficiency but also exemplifies how GenAI can be practically deployed to build scalable, real-world business applications.

Core Features

- Sales Data Ingestion - Automatically reads and updates sales data.
- Demand Forecasting - Predicts future product demand using ML models.
- Low-Stock Detection - Identifies items that are close to running out.
- Restocking Suggestions - Recommends when and how much to reorder.

System Workflow Overview



Sales Data
(CSV/Source Input)

Data Pipeline
(retaii_main.py)

Forecasting
Model

Dashboard
(Streamiit)

Output CSV Files

- Retailinventory_Demand Forecast.csv
- Retailinventory_ Recommendationss.csv

API Layer
(FastAPI)

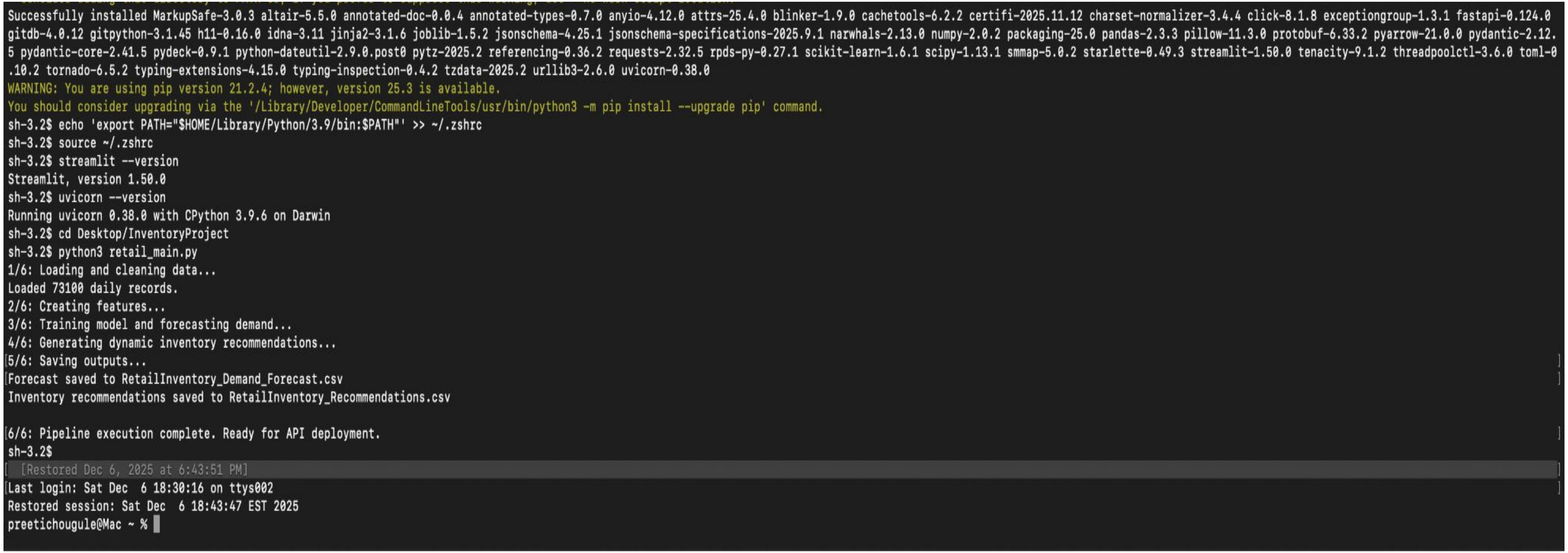
ML Pipeline Execution

- The process begins with the ML pipeline.
- The script loads historical inventory & sales data.
- It trains forecasting models (ARIMA / Random Forest / LSTM).
- It generates:

  - **RetailInventory_Demand_Forecast.csv**

  - **RetailInventory_Recommendations.csv**
- These files are used for prediction and restocking logic.

ML Pipeline Execution



```
Successfully installed MarkupSafe-3.0.3 altair-5.5.0 annotated-doc-0.0.4 annotated-types-0.7.0 anyio-4.12.0 attrs-25.4.0 blinker-1.9.0 cachetools-6.2.2 certifi-2025.11.12 charset-normalizer-3.4.4 click-8.1.8 exceptiongroup-1.3.1 fastapi-0.124.0
gitdb-4.0.12 gitpython-3.1.45 h11-0.16.0 idna-3.11 jinja2-3.1.6 joblib-1.5.2 jsonschema-4.25.1 jsonschema-specifications-2025.9.1 narwhals-2.13.0 numpy-2.0.2 packaging-25.0 pandas-2.3.3 pillow-11.3.0 protobuf-6.33.2 pyarrow-21.0.0 pydantic-2.12.
5 pydantic-core-2.41.5 pydeck-0.9.1 python-dateutil-2.9.0.post0 pytz-2025.2 referencing-0.36.2 requests-2.32.5 rpds-py-0.27.1 scikit-learn-1.6.1 scipy-1.13.1 smmap-5.0.2 starlette-0.49.3 streamlit-1.50.0 tenacity-9.1.2 threadpoolctl-3.6.0 toml-0
.10.2 tornado-6.5.2 typing-extensions-4.15.0 typing-inspection-0.4.2 tzdata-2025.2 urllib3-2.6.0 uvicorn-0.38.0
WARNING: You are using pip version 21.2.4; however, version 25.3 is available.
You should consider upgrading via the '/Library/Developer/CommandLineTools/usr/bin/python3 -m pip install --upgrade pip' command.
sh-3.2$ echo 'export PATH="$HOME/Library/Python/3.9/bin:$PATH"' >> ~/.zshrc
sh-3.2$ source ~/.zshrc
sh-3.2$ streamlit --version
Streamlit, version 1.50.0
sh-3.2$ uvicorn --version
Running uvicorn 0.38.0 with CPython 3.9.6 on Darwin
sh-3.2$ cd Desktop/InventoryProject
sh-3.2$ python3 retail_main.py
1/6: Loading and cleaning data...
Loaded 73100 daily records.
2/6: Creating features...
3/6: Training model and forecasting demand...
4/6: Generating dynamic inventory recommendations...
5/6: Saving outputs...
Forecast saved to RetailInventory_Demand_Forecast.csv
Inventory recommendations saved to RetailInventory_Recommendations.csv

6/6: Pipeline execution complete. Ready for API deployment.
sh-3.2$
  [Restored Dec 6, 2025 at 6:43:51 PM]
Last login: Sat Dec  6 18:30:16 on ttys002
Restored session: Sat Dec  6 18:43:47 EST 2025
preetichougule@Mac ~ %
```

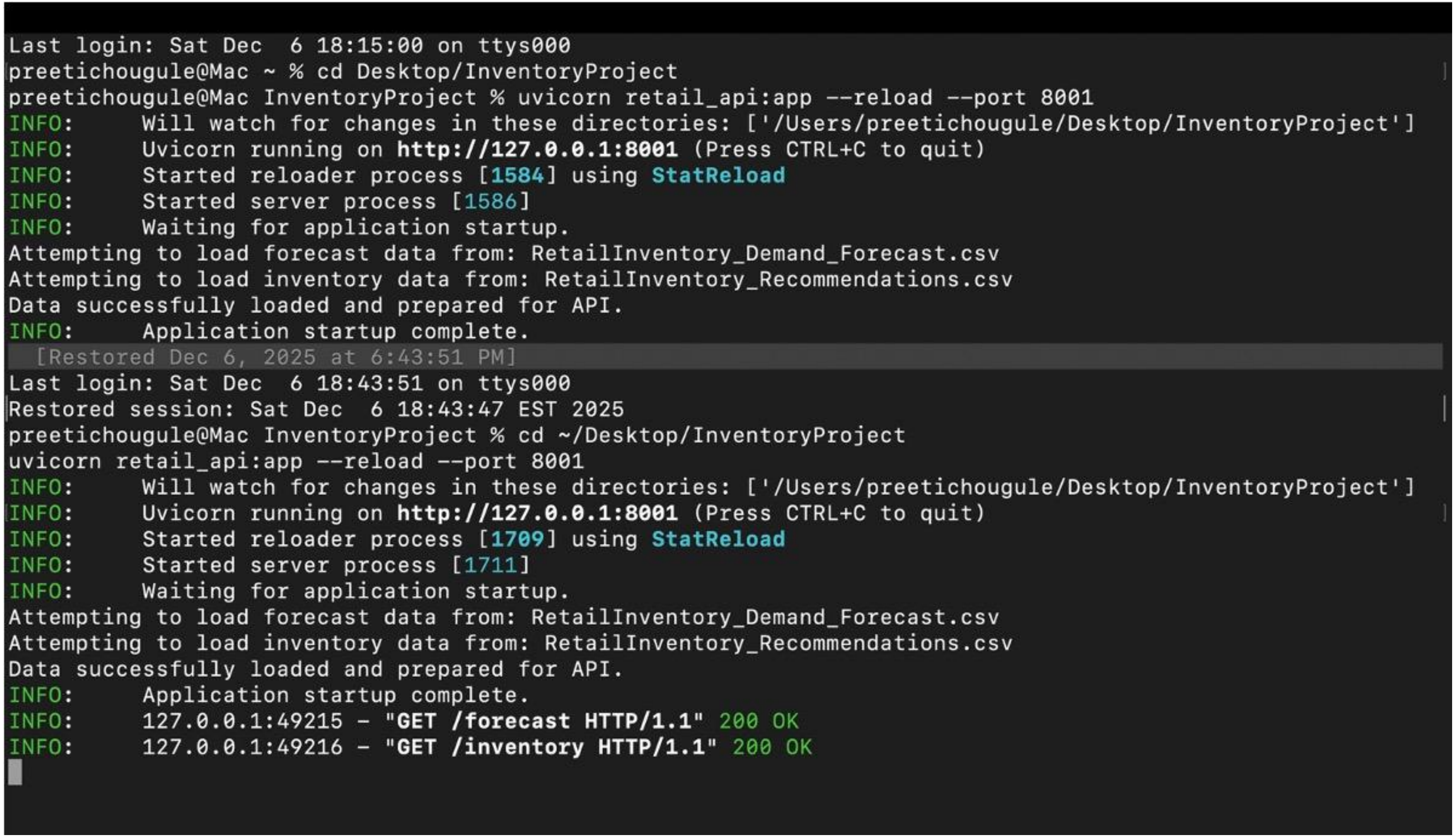
API Backend Running - FastAPI

- The process begins with the ML pipeline.
- The script loads historical inventory & sales data.
- It trains forecasting models (ARIMA / Random Forest / LSTM).
- It generates:

  - **RetailInventory_Demand_Forecast.csv**

  - **RetailInventory_Recommendations.csv**
- These files are used for prediction and restocking logic.

API Backend Running - FastAPI



```
Last login: Sat Dec  6 18:15:00 on ttys000
preetichougule@Mac ~ % cd Desktop/InventoryProject
preetichougule@Mac InventoryProject % uvicorn retail_api:app --reload --port 8001
INFO:     Will watch for changes in these directories: ['/Users/preetichougule/Desktop/InventoryProject']
INFO:     Uvicorn running on http://127.0.0.1:8001 (Press CTRL+C to quit)
INFO:     Started reloader process [1584] using StatReload
INFO:     Started server process [1586]
INFO:     Waiting for application startup.
Attempting to load forecast data from: RetailInventory_Demand_Forecast.csv
Attempting to load inventory data from: RetailInventory_Recommendations.csv
Data successfully loaded and prepared for API.
INFO:     Application startup complete.
  [Restored Dec 6, 2025 at 6:43:51 PM]
Last login: Sat Dec  6 18:43:51 on ttys000
Restored session: Sat Dec  6 18:43:47 EST 2025
preetichougule@Mac InventoryProject % cd ~/Desktop/InventoryProject
uvicorn retail_api:app --reload --port 8001
INFO:     Will watch for changes in these directories: ['/Users/preetichougule/Desktop/InventoryProject']
INFO:     Uvicorn running on http://127.0.0.1:8001 (Press CTRL+C to quit)
INFO:     Started reloader process [1709] using StatReload
INFO:     Started server process [1711]
INFO:     Waiting for application startup.
Attempting to load forecast data from: RetailInventory_Demand_Forecast.csv
Attempting to load inventory data from: RetailInventory_Recommendations.csv
Data successfully loaded and prepared for API.
INFO:     Application startup complete.
INFO:     127.0.0.1:49215 - "GET /forecast HTTP/1.1" 200 OK
INFO:     127.0.0.1:49216 - "GET /inventory HTTP/1.1" 200 OK
```

Dashboard Server Running - Streamlit UI Launch

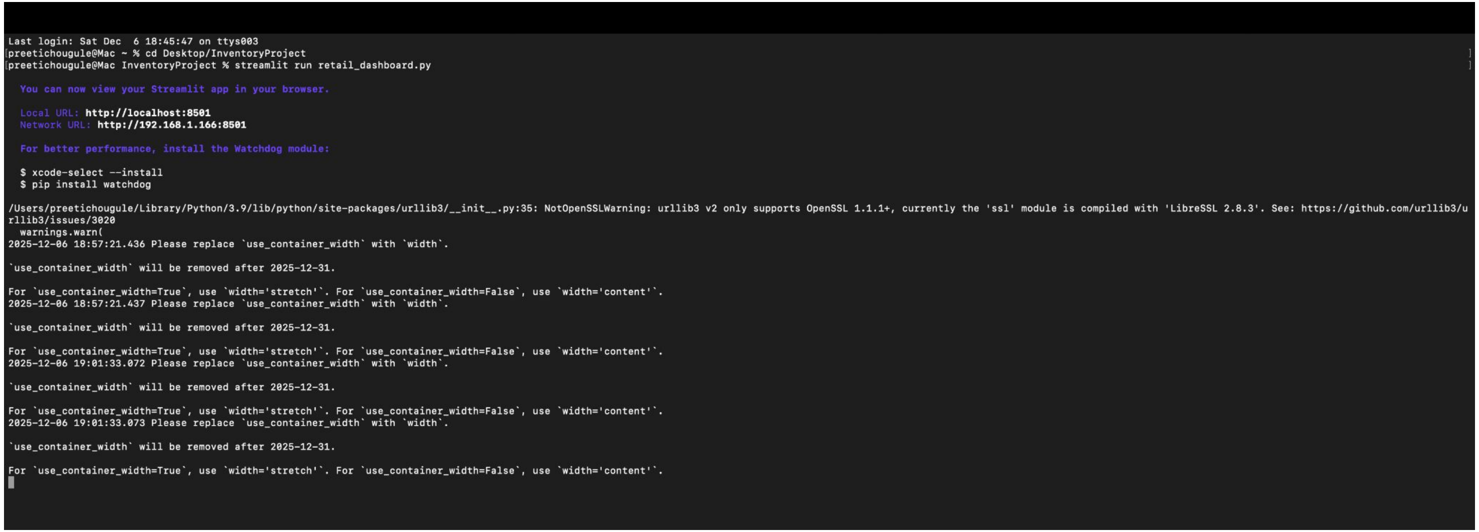
- Once the API is active, the dashboard is launched using Streamlit.
- The dashboard automatically connects to the API and retrieves:

  - Forecasts

  - Inventory levels

  - Restocking recommendations
- This provides a user-friendly interface for interacting with the ML outputs.

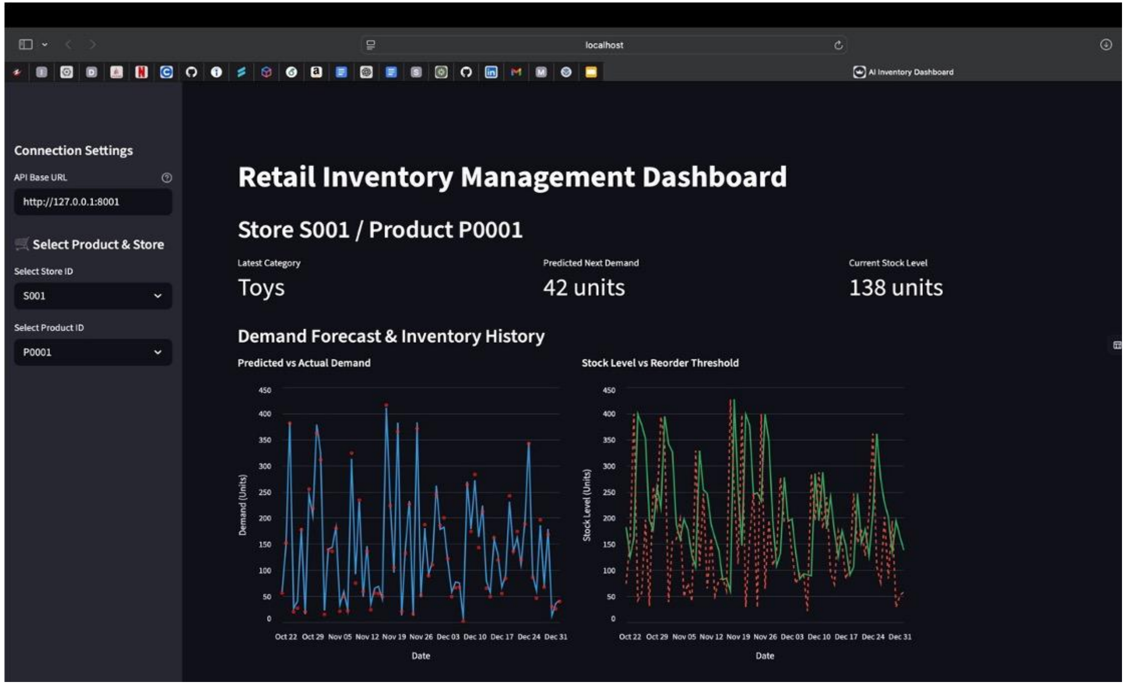
Dashboard Server Running - Streamlit UI Launch



```
Last login: Sat Dec  6 18:45:47 on ttys003
preetichougule@Mac ~ % cd Desktop/InventoryProject
preetichougule@Mac InventoryProject % streamlit run retail_dashboard.py

  You can now view your Streamlit app in your browser.

  Local URL: http://localhost:8501
  Network URL: http://192.168.1.166:8501

  For better performance, install the Watchdog module:

  $ xcode-select --install
  $ pip install watchdog

/Users/preetichougule/Library/Python/3.9/lib/python/site-packages/urllib3/__init__.py:35: NotOpenSSLWarning: urllib3 v2 only supports OpenSSL 1.1.1+, currently the 'ssl' module is compiled with 'LibreSSL 2.8.3'. See: https://github.com/urllib3/u
rllib3/issues/3020
  warnings.warn(
2025-12-06 18:57:21.436 Please replace `use_container_width` with `width`.

`use_container_width` will be removed after 2025-12-31.

For `use_container_width=True`, use `width='stretch'`. For `use_container_width=False`, use `width='content'`.
2025-12-06 18:57:21.437 Please replace `use_container_width` with `width`.

`use_container_width` will be removed after 2025-12-31.

For `use_container_width=True`, use `width='stretch'`. For `use_container_width=False`, use `width='content'`.
2025-12-06 19:01:33.072 Please replace `use_container_width` with `width`.

`use_container_width` will be removed after 2025-12-31.

For `use_container_width=True`, use `width='stretch'`. For `use_container_width=False`, use `width='content'`.
2025-12-06 19:01:33.073 Please replace `use_container_width` with `width`.

`use_container_width` will be removed after 2025-12-31.

For `use_container_width=True`, use `width='stretch'`. For `use_container_width=False`, use `width='content'`.
```

Dashboard Home View

- Users start by selecting:
  - Store ID
  - Product ID
- The dashboard displays:
  - Category
  - Current stock level
  - Predicted demand
  - Inventory trend over time
- This interface is the main control panel for decision-making.



Connection Settings

API Base URL: http://127.0.0.1:8001

🛒 Select Product & Store

Select Store ID: S001

Select Product ID: P0001

Retail Inventory Management Dashboard

Store S001 / Product P0001

Latest Category: Toys

Predicted Next Demand: 42 units

Current Stock Level: 138 units

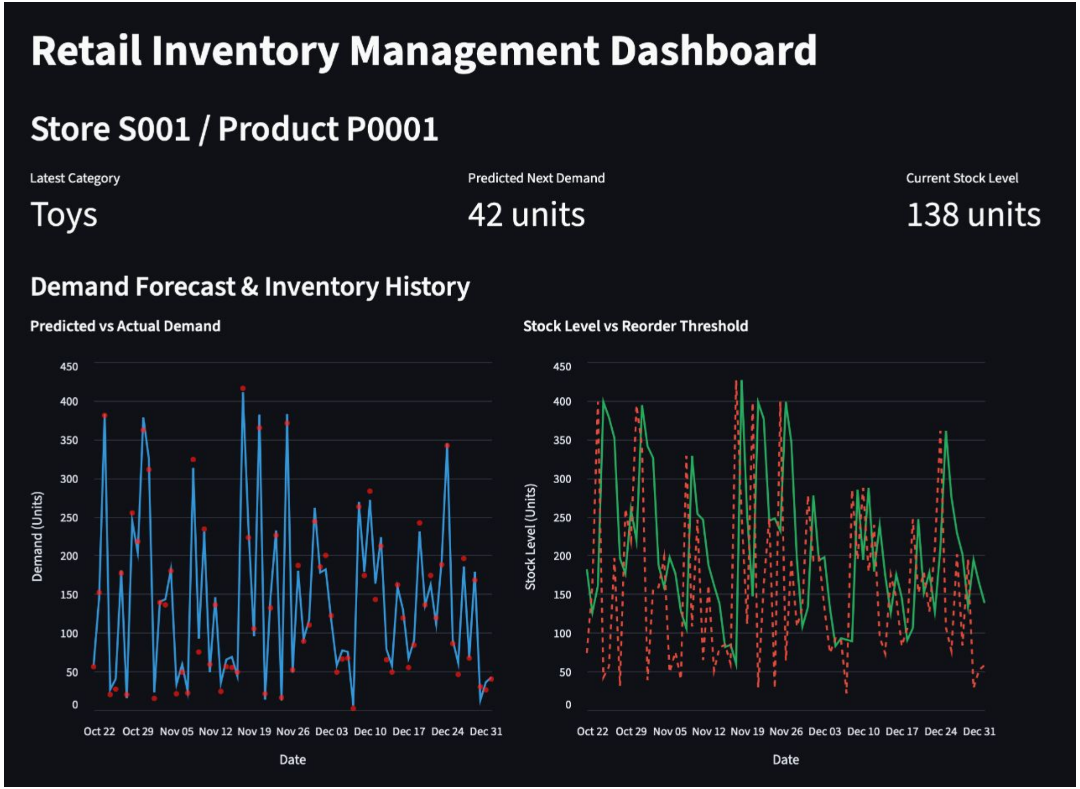
Demand Forecast & Inventory History

Forecast Visualization - AI Prediction

- The forecasting model predicts future demand trends.
- Two charts are shown:

  1. Predicted vs. Actual Demand – Model accuracy visualization

  2. Stock Level vs. Reorder Threshold – Shows when stock dips below safety levels.
- These predictive charts guide the retailer on when demand will rise or fall.



Retail Inventory Management Dashboard

Store S001 / Product P0001

Latest Category	Predicted Next Demand	Current Stock Level
Toys	42 units	138 units

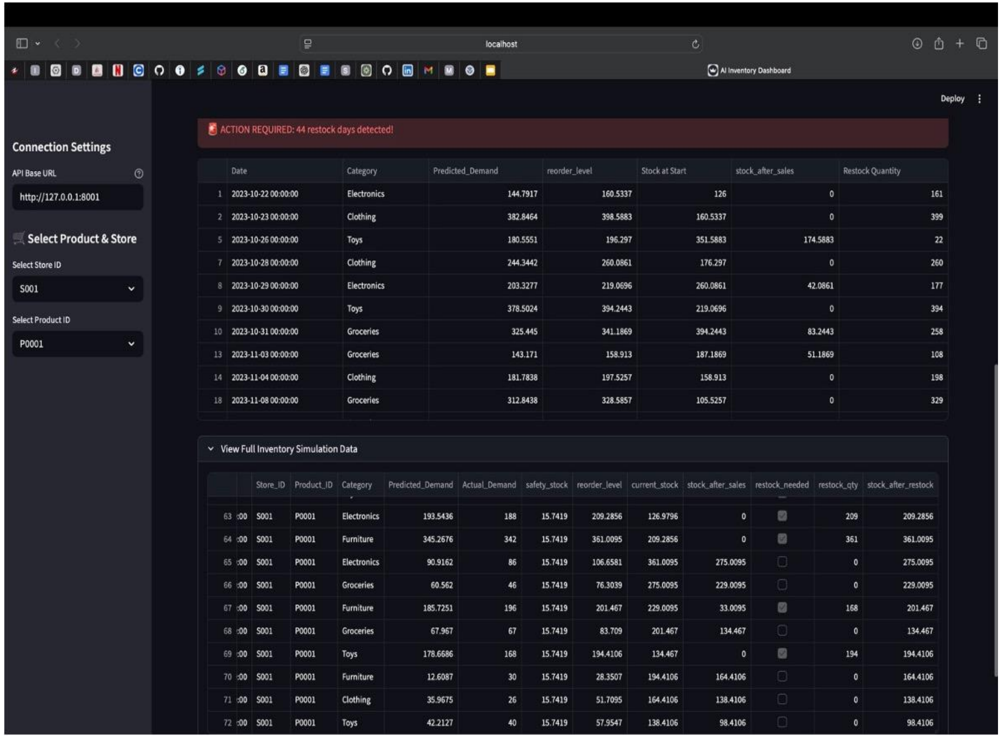
Demand Forecast & Inventory History

Predicted vs Actual Demand

Stock Level vs Reorder Threshold

Restock Alerts & Recommendations

- The dashboard analyzes forecasted demand and current stock.
- It identifies dates on which the inventory will reach low levels.
- It recommends:
  - How much quantity to reorder
  - When the stock will run out
  - What safety stock should be maintained
- This enables automated, data-driven restocking.



Connection Settings

API Base URL: http://127.0.0.1:8001

Select Product & Store

Select Store ID: S001

Select Product ID: P0001

ACTION REQUIRED: 44 restock days detected!

	Date	Category	Predicted_Demand	reorder_level	Stock at Start	stock_after_sales	Restock Quantity
1	2023-10-22 00:00:00	Electronics	144.7917	160.5337	126	0	161
2	2023-10-23 00:00:00	Clothing	382.8464	398.5883	160.5337	0	399
5	2023-10-26 00:00:00	Toys	180.5551	196.297	351.5883	174.5883	22
7	2023-10-28 00:00:00	Clothing	244.3442	260.0861	176.297	0	260
8	2023-10-29 00:00:00	Electronics	203.3277	219.0696	260.0861	42.0861	177
9	2023-10-30 00:00:00	Toys	378.5024	394.2443	219.0696	0	394
10	2023-10-31 00:00:00	Groceries	325.445	341.1869	394.2443	83.2443	258
13	2023-11-03 00:00:00	Groceries	143.171	158.913	187.1869	51.1869	108
14	2023-11-04 00:00:00	Clothing	181.7838	197.5257	158.913	0	198
18	2023-11-08 00:00:00	Groceries	312.8438	328.5857	105.5257	0	329

View Full Inventory Simulation Data

		Store_ID	Product_ID	Category	Predicted_Demand	Actual_Demand	safety_stock	reorder_level	current_stock	stock_after_sales	restock_needed	restock_qty	stock_after_restock
63	:00	S001	P0001	Electronics	193.5436	188	15.7419	209.2856	126.9796	0	☑	209	209.2856
64	:00	S001	P0001	Furniture	345.2676	342	15.7419	361.0095	209.2856	0	☑	361	361.0095
65	:00	S001	P0001	Electronics	90.9162	86	15.7419	106.6581	361.0095	275.0095	☐	0	275.0095
66	:00	S001	P0001	Groceries	60.562	46	15.7419	76.3039	275.0095	229.0095	☐	0	229.0095
67	:00	S001	P0001	Furniture	185.7251	196	15.7419	201.467	229.0095	33.0095	☑	168	201.467
68	:00	S001	P0001	Groceries	67.967	67	15.7419	83.709	201.467	134.467	☐	0	134.467
69	:00	S001	P0001	Toys	178.6686	168	15.7419	194.4106	134.467	0	☑	194	194.4106
70	:00	S001	P0001	Furniture	12.6087	30	15.7419	28.3507	194.4106	164.4106	☐	0	164.4106
71	:00	S001	P0001	Clothing	35.9675	26	15.7419	51.7095	164.4106	138.4106	☐	0	138.4106
72	:00	S001	P0001	Toys	42.2127	40	15.7419	57.9547	138.4106	98.4106	☐	0	98.4106

Tech Stack

- Python, Pandas, NumPy
- Scikit-learn, ARIMA/LSTM forecasting models
- FastAPI for backend
- Streamlit UI dashboard for recommendations

Thank You!!