**1. System & Environment**

* **OS:** Windows 10/11, macOS Monterey (or newer), or Linux (Ubuntu 20.04+, Fedora 34+)
* **RAM:** ≥ 8 GB (12 GB+ recommended for XGBoost training on large datasets)
* **Disk:** ≥ 5 GB free (for datasets, models, and logs)
* **Processor:** Quad-core CPU (or better)

**2. Python & Virtual Environment**

1. **Python** 3.8 or 3.9
2. **pip** (comes with Python)
3. **(Recommended)** Create and activate a virtual environment:
4. python3 -m venv venv
5. source venv/bin/activate # macOS/Linux
6. venv\Scripts\activate # Windows

**3. Repository Setup**

1. **Clone the repo** (all code, modules, data folders):
2. git clone https://github.com/<your-org>/YourAnalyst.git
3. cd YourAnalyst
4. **Data folder**
   * Place Rossmann.csv (or train.csv, store.csv) into data/
   * Ensure subfolders:
   * YourAnalyst/
   * ├─ data/
   * │ ├─ raw/ # original downloads
   * │ └─ cleaned/ # exported after preprocessing
   * ├─ models/ # will store .pkl/.joblib files
   * └─ src/ # all .py modules

**4. Python Dependencies**

Create a requirements.txt with at least:

# Core

pandas>=1.4

numpy>=1.22

# Preprocessing & ML

scikit-learn>=1.1

xgboost>=1.6

pyarrow>=8.0 # for Parquet support (if used)

openpyxl>=3.0 # for Excel ingestion

# Visualization

matplotlib>=3.5

seaborn>=0.11

plotly>=5.5

# Web Interface

streamlit>=1.14

# Utilities

joblib>=1.1

Then install:

pip install -r requirements.txt

**5. Running the Modules**

1. **Preprocessing / Visualization / Modeling (YourAnalyst)**
2. streamlit run src/yourAnalyst.py
   * Opens at http://localhost:8501 by default.
3. **Standalone Forecast Dashboard**
4. python src/Dashboard.py
   * Make sure your cleaned data (fc\_df.csv or similar) is in data/cleaned/.
5. **Local Model Deployment (Streamlit)**
6. streamlit run src/deploy.py
   * Loads the saved XGBoost model from models/xgboost\_model.pkl.

**6. Testing & QA**

* **Unit tests**
* pytest tests/
* **Linting (optional)**
* flake8 src/ # or pylint src/

**7. Configuration & Ports**

* Default Streamlit port: **8501**
* If you need multiple apps at once, specify:
* streamlit run yourAnalyst.py --server.port 8502

Once you’ve met all of the above, you’ll be able to:

1. Upload and clean raw Rossmann data in YourAnalyst
2. Explore and model it with XGBoost (or any other algorithm you choose)
3. Export the trained model
4. Run a separate dashboard for 90-day forecasts + inventory metrics
5. Deploy locally via Streamlit for interactive use