2. Automated Reordering System (Python)

python

Copy

import pandas as pd

from datetime import datetime, timedelta

def calculate\_reorder(product\_sales, current\_stock, lead\_time=3, safety\_stock=0.25):

# Calculate average weekly sales

weekly\_sales = product\_sales.resample('W').sum()

avg\_weekly = weekly\_sales.mean()

# Calculate reorder point

reorder\_point = (avg\_weekly \* (lead\_time/7)) \* (1 + safety\_stock)

# Calculate order quantity

order\_qty = max(0, reorder\_point - current\_stock)

return order\_qty

3. Machine Learning for Demand Forecasting (Python/R)

python

Copy

# Using Facebook Prophet for time series forecasting

from prophet import Prophet

def forecast\_demand(sales\_data, periods=12):

# Prepare data

df = sales\_data.reset\_index()

df.columns = ['ds', 'y']

# Create and fit model

model = Prophet(seasonality\_mode='multiplicative')

model.fit(df)

# Make forecast

future = model.make\_future\_dataframe(periods=periods, freq='W')

forecast = model.predict(future)

return forecast[['ds', 'yhat', 'yhat\_lower', 'yhat\_upper']]

4. Expiry Management System

r

Copy

# R code for expiry alerts

library(dplyr)

library(lubridate)

check\_expiry <- function(inventory\_data, alert\_days = 90) {

inventory\_data %>%

mutate(

days\_to\_expiry = as.numeric(expiry\_date - Sys.Date()),

expiry\_status = case\_when(

days\_to\_expiry <= 0 ~ "Expired",

days\_to\_expiry <= alert\_days ~ "Expiring Soon",

TRUE ~ "OK"

)

) %>%

filter(expiry\_status != "OK") %>%

arrange(days\_to\_expiry)

}

Implementation Roadmap

1. **Phase 1 (0-2 weeks)**
   * Set up MySQL database with current inventory
   * Import historical sales data
   * Implement basic reorder calculations
2. **Phase 2 (2-4 weeks)**
   * Develop automated reporting (weekly stock levels)
   * Implement expiry tracking system
   * Create dashboard for inventory visibility
3. **Phase 3 (4-8 weeks)**
   * Build ML forecasting models
   * Implement predictive ordering
   * Set up alert system for stockouts/expiries
4. **Ongoing Optimization**
   * Refine models with new sales data
   * Adjust safety stock levels based on service level targets
   * Expand to multi-location inventory management

Key Performance Indicators (KPIs) to Monitor

1. Inventory Turnover Ratio
2. Days of Inventory On Hand
3. Stockout Rate
4. Expired Stock Percentage
5. Order Accuracy Rate

Would you like me to elaborate on any specific aspect of this system or provide sample code for a particular component? I can also help design the dashboard interface if that would be helpful for your team.