# Stock Tracking System

A comprehensive time-series stock tracking system built with Node.js, Express, and MongoDB. This system provides real-time stock level tracking, historical data analysis, trend detection, and restocking pattern recognition for retail and inventory environments.

# Features

- Real-time Stock Tracking: Record stock changes with automatic level calculation.

- Historical Data Analysis: Aggregate data by daily, weekly, monthly, and quarterly periods.

- Trend Analysis: Moving average calculations and trend detection.

- Pattern Recognition: Detect restocking cycles and predict next restock dates.

- RESTful API: Complete API endpoints for all functionality.

- Comprehensive Testing: Unit tests for all features.

# Tech Stack

- Backend: Node.js + Express

- Database: MongoDB with Mongoose ODM

- Frontend: HTML/CSS/JS (dashboard in `public/`)

- Testing: Jest + Supertest

# API Endpoints

## Stock Management

- `POST /api/stocks/update`

Record a stock change.

- `GET /api/stocks/current?productId=123`

Get current stock level for a product.

## Historical Data

- `GET /api/stocks/history?productId=123&from=YYYY-MM-DD&to=YYYY-MM-DD&period=daily`

Get historical data with aggregation.

## Analysis

- `GET /api/stocks/trend?productId=123&period=weekly&window=7`

Get trend analysis (moving average, trend direction, confidence).

- `GET /api/stocks/restock-pattern?productId=123`

Analyze restocking patterns and predict next restock.

## Data Management

- `POST /api/stocks/seed`

Seed database with sample data.

- `DELETE /api/stocks/clear`

Clear all data from database.

- `GET /api/stocks/stats`

Get database statistics.

## Real-Time Updates

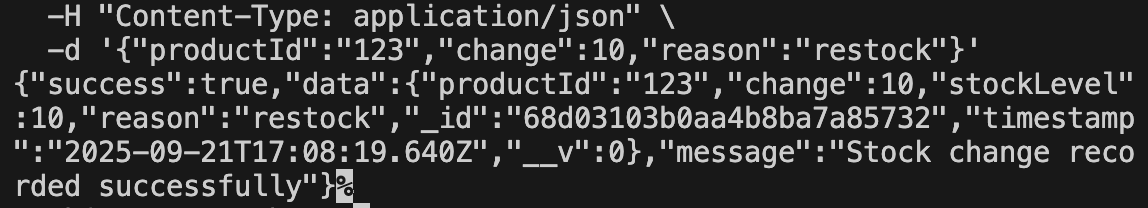
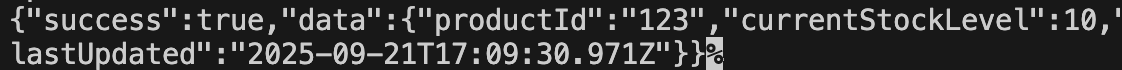
- `GET /api/stocks/events`

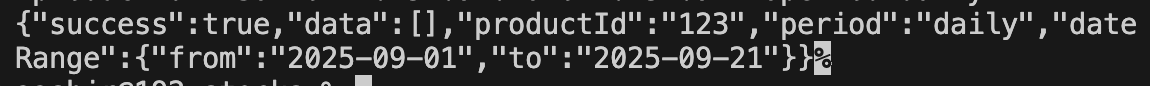
Server-Sent Events for real-time stock updates.

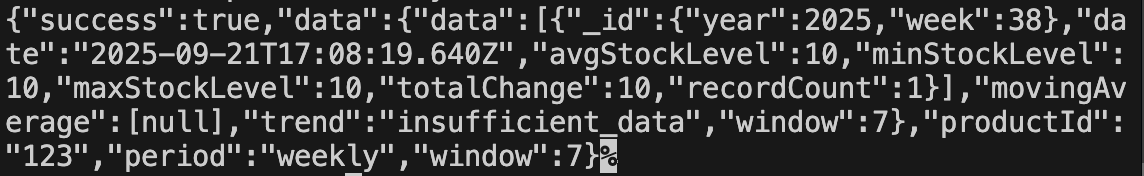
# Result Screenshots

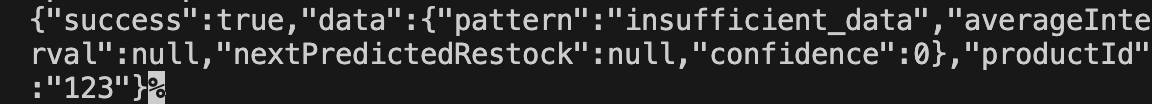
1. Record a stock change  
curl -X POST http://localhost:3000/api/stocks/update \

-H "Content-Type: application/json" \

-d '{"productId":"123","change":10,"reason":"restock"}'  
  
  
  
  
  
  
2. Get current stock level  
curl <http://localhost:3000/api/stocks/current?productId=123>  


3. Get historical data  
curl <http://localhost:3000/api/stocks/history?productId=123&from=2025-09-01&to=2025-09-21&period=daily>  


4. Get trend analysis  
  
curl “<http://localhost:3000/api/stocks/trend?productId=123&period=weekly&window=7>”  
  


5. Analyze restocking patterns  
  
curl <http://localhost:3000/api/stocks/restock-pattern?productId=123>  
  


6. Seed database with sample data  
curl -X POST <http://localhost:3000/api/stocks/seed>  
  
