# **Everest Inventory Intelligence Module - Complete Design Specification**

## **Executive Summary**

A comprehensive inventory replenishment and supply planning module within the Everest ERP suite, designed for SaaS companies managing physical goods across multiple global warehouses. The solution transforms manual Excel-based planning into an AI-powered, exception-driven system that reduces stockouts by 50% and inventory costs by 20%.

## **Target Customer Profile**

* **Company Type**: Non-public US-based SaaS company with physical product sales
* **Scale**: $50M+ annual inventory, 200+ SKUs
* **Operations**: Multi-warehouse (US, UK, Germany), multi-currency
* **Current State**: Manual Excel processes, 15% stockout rate, $8M excess inventory
* **Pain Points**: 4+ hours weekly planning, no exception visibility, reactive ordering

## **Core Architecture**

### **Data Model Structure**

The system uses 11 interconnected tables:

1. **Items\_Master**: SKU catalog with costs, prices, ABC classification
2. **Warehouses**: Locations with currencies, lead time adjustments
3. **Inventory\_On\_Hand**: Real-time stock positions by SKU/location
4. **Demand\_History**: Historical sales data for pattern analysis
5. **Demand\_Statistics**: Calculated averages, standard deviations, trends
6. **Demand\_Forecast**: ML-generated predictions with confidence levels
7. **Reorder\_Rules**: Service levels, review periods, safety stock parameters
8. **Suppliers**: Vendor information, MOQs, lead times
9. **Item\_Suppliers**: SKU-supplier relationships by location
10. **Open\_Purchase\_Orders**: In-transit inventory tracking
11. **Currency\_Exchange\_Rates**: Real-time FX for global calculations

### **System Layers**

#### **1. Master Data Layer**

* Stable foundational data (items, warehouses, suppliers)
* Updates trigger cascade through system

#### **2. Current State Layer**

* Real-time inventory positions (🟢 updates)
* Open POs and in-transit visibility
* Historical demand patterns (🔵 daily batch)

#### **3. Intelligence Layer**

* Statistical calculations (averages, variations)
* ML ensemble forecasting (🟡 hourly updates)
* Dynamic reorder rules (⚪ weekly review)

#### **4. Decision Layer**

* Replenishment recommendations
* Exception detection and alerts
* Network optimization opportunities

#### **5. Action Layer**

* Purchase order generation
* Inter-warehouse transfer suggestions
* Automated supplier communication

## **Key Calculations Engine**

### **Inventory Position (IP)**

IP = On-hand + In-transit - Allocated - Backorders

### **Safety Stock (Periodic Review Model)**

Safety Stock = z × σ × √(P+L)

Where:

- z = service level factor (1.65 for 95%)

- σ = demand standard deviation

- P+L = Review Period + Lead Time

### **Target Level & Order Quantity**

Target Level (T) = Demand during (P+L) + Safety Stock

Order Quantity (Q) = T - IP (adjusted for MOQ/multiples)

### **Newsvendor Optimization (Seasonal Items)**

Critical Ratio = Cu/(Cu + Co)

Optimal stock balances understocking vs overstocking costs

## **ML/AI Components**

### **Forecast Ensemble Architecture**

* **Model Zoo**: Prophet, LightGBM, N-HiTS, ADIDA, TimeGPT
* **Dynamic Weighting**: Based on recent performance (MAE, MAPE)
* **Continuous Learning**: Daily accuracy checks trigger retraining
* **External Signals**: Weather, events, social sentiment, economic indicators

### **Natural Language Interface**

* **Conversational Queries**: "Why are we ordering 500 units of SKU001?"
* **Intelligent Explanations**: Shows calculations and reasoning
* **Action Execution**: Approve/modify orders through chat
* **Proactive Alerts**: AI detects anomalies and suggests actions

### **Learning Loop Features**

* Daily forecast vs actual comparison
* Automatic model reweighting
* Feature importance recalculation
* Drift detection and retraining triggers

## **User Interface Design**

### **1. Executive Dashboard**

* Global inventory value by location
* Critical alerts summary
* Order recommendations value
* Service level trending

### **2. Replenishment Workbench**

* Filterable grid: SKU | Location | Stock | Coverage | Recommended Action
* Bulk actions: Approve, modify, export
* Exception highlighting (red/yellow/green)
* One-click PO generation

### **3. AI Assistant Chat**

* Bottom-right embedded interface
* Natural language queries
* Rich responses with tables/charts
* Action buttons in chat

### **4. Exception Management**

* Card-based alert system
* Categorized by severity
* Actionable recommendations
* Snooze/escalate options

## **Implementation Approach**

### **Phase 1 (Weeks 1-4): Foundation**

* Basic dashboard deployment
* Automated calculations
* Initial time savings: 3 hours/week

### **Phase 2 (Weeks 5-8): Intelligence**

* ML forecasting activation
* Exception detection
* Error reduction: 50%

### **Phase 3 (Weeks 9-12): Optimization**

* AI assistant launch
* Network optimization
* Full ROI realization

## **Expected Outcomes**

### **Quantitative Benefits**

* **Stockout Reduction**: 50% (from 15% to 7.5%)
* **Inventory Reduction**: 20% ($1.6M working capital freed)
* **Planning Time**: 80% reduction (4 hours to 45 minutes)
* **Forecast Accuracy**: 15-20% MAPE improvement

### **Financial Impact**

* **Investment**: $150,000
* **Annual Savings**: $600,000
* **Payback Period**: 3 months
* **5-Year NPV**: $2.1M

## **Technical Integration**

### **Inbound Data Sources**

* ERP systems (item master, costs)
* WMS (real-time inventory)
* CRM (sales pipeline)
* External APIs (weather, trends)

### **Outbound Connections**

* Procurement systems (PO creation)
* Financial planning (inventory projections)
* BI tools (analytics/reporting)
* Communication (Slack/email alerts)

## **Differentiation Points**

1. **Multi-Location Intelligence**: Unified view with currency handling
2. **Adaptive ML Ensemble**: Self-improving forecasts
3. **Conversational Interface**: Natural language for all users
4. **Exception-Driven**: Focus attention where needed
5. **Network Optimization**: Cross-warehouse opportunities

## **Deliverables Created**

1. **Notion Playbook**: Interactive one-page solution overview
2. **Excel Demo**: Working calculations with sample data
3. **Slide Deck**: 10-slide presentation narrative
4. **Architecture Diagrams**: Data flow and system design
5. **UI Mockups**: Dashboard and key screens
6. **ROI Calculator**: Customizable business case tool

## **Key Technical Details for Implementation**

* **Frontend**: React-based UI with embedded chat widget
* **ML Pipeline**: Python (Darts, Prophet, LightGBM) with Airflow orchestration
* **Real-time Updates**: Kafka for event streaming
* **Vector Database**: Pinecone for similarity search
* **LLM Integration**: GPT-3.5/Llama 2 for natural language
* **Deployment**: Kubernetes with auto-scaling

This specification provides a complete picture of the Everest Inventory Intelligence module, combining operational excellence with cutting-edge AI/ML capabilities to transform inventory management from reactive to proactive.