Product Requirements

Plock PRD, AI features, integrations (Fortnox, PostNord, DHL, Shopify), user personas

AI Features (Priority Order)

AI Features (Priority Order)

Overview

Plock is **AI-native** — not AI-bolted-on. Every data model is designed with AI querying in mind. The AI layer runs continuously; warehouse workers are alerted, not asked to check.

Target user for AI: A warehouse worker with dirty gloves asking *var finns SKU-X* on a cracked phone screen — not an analyst building pivot tables.

Language: Swedish first. Short, direct responses. Action-oriented. Mobile-friendly.

Priority 1 — Plock AI Chat (Launch)

Type: LLM natural language interface **Model:** Claude API (claude-sonnet-4-5) **Language:** Swedish

What it does

- Answer warehouse questions in natural language Swedish
- Execute actions with single confirmation (move stock, reserve, assign pick)
- Access real-time inventory, orders, performance data
- Multi-turn conversation with warehouse context injected

Example queries

- *Hur manga enheter av SKU-1234 har vi?* — live stock count + trend
- *Vilka ordrar ar forsenade idag?* — list with root cause
- *Flytta 50 enheter fran A-01-01 till B-02-03* — confirmation + action

Architecture

- System prompt includes: warehouse schema snapshot, org context, user role, current time

- Function calling for write operations (move, reserve, assign)
- Organization data never sent raw to Claude — only sanitised context
- Response target: less than 3 seconds

Priority 2 — Smart Picking (Launch)

Type: Route optimisation **Engine:** OR-Tools (Google) — VRP/TSP solver **Reduction target:** 30-70% travel time

What it does

- Automatically generates optimal pick routes when orders are released
- Groups orders into picking waves based on zone, carrier cutoff, priority
- Outputs sequential bin list on mobile scanner
- No configuration required — runs on every order release

How it works

1. Order released — ML microservice receives order + bin locations
2. OR-Tools VRP solver computes shortest path across warehouse zones
3. Optimised pick list returned to mobile app in under 2 seconds
4. Worker follows sequence — no backtracking

ROI

- Industry data: 30-70% of warehouse time is walking (not picking)
- Conservative 20% efficiency gain = 8,800 SEK/month per worker saved

---

Priority 3 — Anomaly Detection (Launch)

Type: Statistical detection on inventory transactions **Algorithms:** Z-score + isolation forest (scikit-learn) **Trigger:** Real-time, on every transaction

What it detects

- Sudden stock drops (theft, mispick, data error)
- Unusual pick rates (demand spike or data entry error)
- Inventory discrepancy after goods receipt
- Carrier label printed but no scan out (lost shipment risk)

Alert flow

1. Transaction recorded — Python microservice analyses against rolling baseline
2. Z-score above 3 sigma or isolation forest flags as outlier — alert created
3. Warehouse manager receives push notification + AI chat message
4. Manager confirms or dismisses — feedback improves model

Priority 4 — Demand Forecasting (Month 6)

Type: Time-series forecasting **Models:** XGBoost + Prophet (Facebook) **Input:** Order history (minimum 90 days)

What it does

- Predicts demand per SKU for next 7/14/30 days
- Generates reorder recommendations with suggested quantity and timing
- Adjusts for seasonality (Swedish calendar events, e-commerce peaks)
- Accessible in AI Chat: *Vad ska vi bestalla hem nasta vecka?*

Priority 5 — Smart Slotting (Month 6)

Type: ML-based placement recommendations **Input:** Pick frequency, order affinity, SKU dimensions

What it does

- Recommends optimal bin locations for each SKU based on pick frequency
- Groups frequently co-picked SKUs near each other (affinity slotting)
- Identifies slow-movers in prime locations and suggests moves
- Presented as actionable task list, not a forced migration

Priority 6 — Labor Planning (Month 12)

Type: Shift optimisation **Constraints:** Swedish labour law (Arbetstidslagen)

What it does

- Predicts staffing needs based on demand forecasts
- Generates compliant shift schedules (max hours, breaks, rest periods)
- Suggests overtime or temp worker needs 1-2 weeks in advance
- Integrates with order volume forecasts from Demand Forecasting module

---

AI Cost Model (per customer)

Feature	Est. cost/month (Growth plan)
Plock AI Chat (1,000 queries/month)	~45 SEK (Claude Sonnet pricing)
Smart Picking (10,000 optimisations)	~5 SEK (OR-Tools compute)
Anomaly Detection (100K transactions)	~10 SEK (Python microservice)
Total AI cost per Growth customer	**~60 SEK/month**
Growth plan revenue	**2,990 SEK/month**
AI cost as % of revenue	**~2%**

Key Integrations

Key Integrations

Integration Priority Table

Integration	Type	Priority	Launch Date
Fortnox	REST v3, OAuth2	**P0**	**Launch**
PostNord	REST, OAuth2	**P0**	**Launch**
DHL Sweden	REST	P1	Month 2
Instabee / Budbee	REST JSON	P1	Month 3
Shopify	REST + GraphQL	P1	Month 3
WooCommerce	REST	P2	Month 6
Visma	REST	P2	Month 6

P0 — Fortnox (Launch Critical)

Why P0: 598,000 Swedish companies use Fortnox. Fortnox Marketplace is the primary distribution channel. Deep integration = instant value for every new customer.

API: Fortnox REST API v3 (api.fortnox.se) **Auth:** OAuth2 (authorisation code flow)

Sync scope

Data	Direction	Frequency
Products (artiklar)	Fortnox to Plock	On demand + webhook
Suppliers (leverantorer)	Fortnox to Plock	On demand
Purchase orders	Bidirectional	Real-time
Sales orders	Fortnox to Plock	Real-time (webhook)
Stock levels	Plock to Fortnox	On every transaction

Data	Direction	Frequency
Invoices	Plock to Fortnox	On order completion

Fortnox Marketplace listing

- Plock listed as official Fortnox Marketplace integration
- Customers connect via OAuth flow directly from Fortnox UI
- Integration reviewed and certified by Fortnox team

P0 — PostNord (Launch Critical)

Why P0: PostNord is the dominant carrier for Swedish e-commerce (primary for ~70% of SMBs).

API: PostNord REST API (developer.postnord.com) **Auth:** OAuth2

Features

- Generate shipping labels (PDF + ZPL for Zebra printers)
- Book pickup
- Track shipments (webhook updates)
- Print manifests at end of day
- Support for: PostNord MyPack, Varubrev, DHL Parcel (via PostNord reseller)

P1 — DHL Sweden (Month 2)

API: DHL Express REST (developer.dhl.com) **Features:** Label generation, booking, tracking, rate quotes **Auth:** API key

---

P1 — Instabee / Budbee (Month 3)

API: Instabee REST JSON (developer.budbee.com) **Features:** Label generation, booking, real-time tracking, consumer notifications **Note:** Instabee acquired Budbee — same API, two brands

---

P1 — Shopify (Month 3)

API: Shopify REST Admin API + GraphQL **Auth:** OAuth2 (Shopify Partner app)

Sync scope

- Orders to Plock outbound queue
- Inventory levels bidirectional (Plock as source of truth)
- Product catalogue to Plock (read-only)
- Fulfilment events to Shopify order tracking

P2 — WooCommerce (Month 6)

API: WooCommerce REST API **Auth:** API key **Sync scope:** Orders, products, inventory levels (similar to Shopify)

P2 — Visma (Month 6)

API: Visma eAccounting REST API **Auth:** OAuth2 **Sync scope:** Similar to Fortnox — products, orders, invoices, stock levels

---

Integration Architecture

All integrations run through the Integration Layer in the Kotlin backend:

File structure:

- integrations/fortnox/ — FortnoxClient.kt, FortnoxSyncService.kt, FortnoxWebhookHandler.kt
- integrations/carriers/ — PostNordClient.kt, DhlClient.kt, InstabeeClient.kt
- integrations/ecommerce/ — ShopifyClient.kt, WooCommerceClient.kt

Queue: Redis + Kotlin coroutines for async integration jobs (label generation, carrier calls, sync tasks)

Error handling: Retry with exponential backoff, dead-letter queue, alert on persistent failure