**(1) Working Capital**

**1) Definition and Purpose**

**What is it?** The treasury/cash balance that finances: purchases (2), shipping (4), operating expenses (5).

**Key note:** You can start the system without entering an initial balance; this does not stop the workflow.

**No mixing with revenues:** Sales collections do not go here directly; they accumulate in the Revenue Balance (7). If you want new funding from profits, you use “Reinvestment from Revenues” (in 7), which increases capital.

**2) Data Structure (Fields)**

* **OpeningBalance (Optional):** Opening balance (base display currency: USD).
* **LedgerEntry (Capital entry):**
  + entry\_id, date, amount (USD), sign (+/-).
  + **Type:**
    - CapitalIn (Deposit/Reinvestment from revenues).
    - CapitalOut (Funding purchase/shipping/expenses).
    - Reclass/Reverse (Reversal/Correction).
  + **Reference:** order\_id, or purchase\_id, or shipping\_leg\_id, or expense\_id.
  + Description/Note (Optional).
  + Payment currency: USD or ETB (documentation only).
  + Exchange rate used if currency = ETB (pulled automatically from settings “10” and stored historically).
* **CapitalBalance:** Instant available balance = Opening + ∑(CapitalIn) − ∑(CapitalOut).
* **Currency:** Reports and balances are in USD. Any payment in ETB is auto-converted at the central exchange rate from (10). No manual exchange entry inside the process.

**3) Workflow (Step-by-Step)**

**A) Create/Modify Opening Balance**

* Option to enter OpeningBalance (can be left blank).
* Save — does not block workflow if zero.

**B) Fund Operating Activities (at time of payment in 2/4/5)**

* In purchases/shipping/expenses, when recording a payment, a field appears: **Funding source:**
  + From Capital → creates LedgerEntry: CapitalOut linked to that operation.
  + External Funding → does not affect capital (but operation itself is recorded as a cost to the order).
* If paid in ETB: conversion to USD automatically at current rate (from 10), and ETB + USD + Rate values are saved in the entry.

**C) Increase Capital via Profits (executed in stage 7)**

* From (7) Revenues screen → “Reinvestment from Revenues” (Admin only).
* Enter amount + transfer cost (bank/wire fee).
* System creates automatically:
  + In (7): deduction from revenue balance by amount + fees, and fees recorded as an operating expense.
  + In (1): CapitalIn with amount only (increasing capital).
* Operation is linked by a single reference to trace movement between (7) and (1).

**D) No “Partner Withdrawals” from Capital**

* Withdrawals for partners are from revenues (7) only (per your request), not from capital.

**4) Controls & Validations**

* **Prevent negative balance:** Option in settings (10).
  + If enabled: no CapitalOut allowed that lowers balance below zero (warning and block).
  + If disabled: warning only, requires manager approval (policy from 8).
* **Lock used entries:** Cannot delete a capital entry linked to an existing operation (purchase/shipping/expense). Corrections must be done via Reclass/Reverse.
* **Exchange rate:** Always pulled from (10). No manual override inside entry.
* **Amount match:** Capital entry amount must equal payment amount linked to it (considering currency conversion).
* **Edit authority:** Editing sensitive entries follows policies (8): one-time only, or with approval, or ±5% tolerance (where applicable).

**5) Accounting Integration**

**With (2) Purchases**

* Payment “from Capital” → CapitalOut linked with purchase\_id.
* Advance payment before price/weight known is allowed:
  + Recorded as Advance to Supplier and can be funded from Capital.
  + When finalized later, advance is settled with invoice (debit/credit difference with supplier) without breaking linkage.

**With (4) Shipping**

* Funding shipping or arrival costs from Capital → CapitalOut linked with shipping\_leg\_id/arrival\_cost\_id.
* Transfer commission % (if paid cash from capital) included within same funding.

**With (5) Operating Expenses**

* Supplies purchase/wages/rent from Capital → CapitalOut linked with expense\_id.
* Consumption of supplies does not create capital entry (internal transfer from supplies inventory to order cost).

**With (6) Sales**

* Sales collection does not increase capital directly; it accumulates in revenue balance (7).
* (Per your request) partner withdrawals are not from capital.

**With (7) Revenues**

* Reinvestment from revenues generates CapitalIn here, and transfer fees recorded as operating expense (5).

**With (10) Settings**

* Central exchange rate (USD/ETB).
* Option to prevent/allow negative balance.
* Entry numbering.

**6) Roles & UI**

* **Admin/Owner:** Full view/control, enable/disable negative prevention, view all entries.
* **Finance:** Full read + add adjustments/reversals with documentation.
* **Purchasing/Shipping/Warehouse/Sales:** Do not use capital screen directly; only select funding source: Capital/External from their screens.
* **Worker:** No access.

**Proposed interface (simple/clear):**

* Live balance at top.
* Filterable Ledger table (by order/supplier/period/type).
* Reverse/Correct button only for authorized roles.

**7) Reports & KPIs**

**Reports:**

* Capital Movement Statement: all entries with type + reference.
* Order funding: how much each order funded from capital vs external.
* Source of increase: Owner deposits/reinvestment from revenues (with links in 7).
* Currency breakdown: amounts paid in ETB and their USD value and exchange rate used.

**KPIs:**

* **Capital Utilization** = (Total CapitalOut for period) ÷ (Opening + CapitalIn for period).
* **Capital Turnover Days** = Avg days between CapitalOut on an order and revenue realization (linked with timeline in 9).
* **% Funding from Capital** = CapitalOut ÷ (CapitalOut + External funding).

**8) Notifications**

* Low balance: when balance drops below threshold (configurable).
* Attempt to exceed balance (when prevention enabled): blocks save and alerts.
* Large ETB payments: reference alert for exchange rate review (documentation only, no adjustment).
* Sensitive entries: any reversal/correction requires approval → Manager notified.

**9) Edge Cases & Fixes**

* **Refund from supplier (e.g., partial cancellation after advance):**
  + Recorded as CapitalIn linked with supplier/operation, closing advance balance.
* **Wrong funding source chosen (Capital vs External):**
  + Create Reverse entry to cancel wrong entry, then create correct entry, with explanation (saved in Audit Log).
* **Exchange rate change later:** No effect on past operations; new rate applies only on future entries.
* **Splitting one payment across multiple orders:** Allowed by distributing entry to multiple references (assigned values).
* **Closing period:** After reports posted, period closed; higher authority needed to reopen (optional).

**10) Operational Checklist**

* Set exchange rate in settings (10).
* Enable/disable negative balance prevention.
* Define entry numbering.
* For every payment in (2/4/5): select correct funding source.
* Link each entry with reference (order/purchase/shipping/expense).
* Use Reverse/Correct instead of delete.
* Match reinvestments from (7) with CapitalIn here.
* Review periodic reports: movement statement, order funding, ETB details.

**11) Quick Numerical Examples**

**Example 1: Purchase funded from Capital in ETB**

* Purchase 2,000 kg × 600 ETB/kg = 1,200,000 ETB.
* Exchange rate in settings = 60 ETB/USD.
* CapitalOut = 20,000 USD (also stored ETB=1,200,000 and Rate=60).
* Purchase enters warehouse, capital decreases $20,000.

**Example 2: Advance before price/weight known**

* Paid supplier $5,000 advance from capital → CapitalOut 5,000.
* Later purchase fixed at $6,200 → remaining $1,200 payable.
* Or if final total $4,700 → supplier credit $300 or refund to capital → CapitalIn 300.

**Example 3: Reinvestment from Revenues**

* In (7) revenues you have sufficient funds. You decide to transfer $100,000, with $500 transfer fee.
* (7): deducts 100,500 from revenues, records $500 as operating expense “Transfer fee.”
* (1): CapitalIn = 100,000 (capital increases by this amount).
* Now upcoming purchases/shipping can be funded.
* **(2) Purchases**
* **1) Definition and Purpose**
* Recording the purchase operations of goods from suppliers, with the basic unit of measurement: kilogram (kg).
* Each purchase operation is linked to an Order ID so we can refer back to it in the following stages (Warehouse, Shipping, Sales).
* **Purpose:** To build a clear record showing: Who is the supplier? What is the price per kilogram? What is the weight? What is the payment method? Is there an advance? Are there outstanding dues?
* **2) Supplier Data (Suppliers)**
* **SupplierID** (automatic or manual number).
* **Trade name.**
* **Notes/Definition** (e.g., medium/high quality, trust/previous experience).
* **(Optional)** Archiving supplier documents (invoice, contract).
* **3) Purchase Entry Data (Purchase Entry)**
* **SupplierID + Supplier Name.**
* **Weight in kilograms (mandatory).**
* **Price per kilogram:** in currency (USD or ETB).
  + If ETB → automatically converted into USD using the central exchange rate from (10).
* **Total = weight × price (calculated automatically).**
* **Payment method:**
  + Cash.
  + Advance → even if weight/price are not yet known.
  + Credit.
* **Amount paid at the time of operation.**
* **Remaining (calculated automatically: Total − Paid).**
* **Currency used + exchange rate** (saved as “historical exchange rate” and does not change).
* **Country of purchase.**
* **Date of transaction.**
* **Quality/Classification** (for reporting purposes).
* **4) Advances (Advance Payments)**
* It is possible to pay an advance to the supplier before he delivers the goods.
* **Status:**
* Advance without data: amount only (no weight/price yet).
* Advance with partial data: weight known but price unknown (or vice versa).
* When the goods arrive and price/weight are determined:
* The advance is settled with the invoice.
* If goods are less → shows “credit balance for supplier” or “refund” reflected in accounts.
* If goods are more than estimated → you will have an additional payable balance.
* **5) Special Cases (Return to Supplier)**
* If goods are returned after being recorded:
* The quantity is deducted from the first warehouse.
* Recorded in supplier account as a “Return” and deducted from the outstanding balance or recorded as refund.
* **6) Link with Capital (1)**
* At the time of payment: an option appears for funding source:
* From Capital → recorded as **CapitalOut.**
* External → not deducted from capital (but still a purchase cost).
* Any payment in ETB is converted into USD and saved in both currencies + exchange rate.
* **7) Workflow (Step by Step)**
* Add supplier (or select from list).
* Enter data: weight/price/payment.
* Choose funding source (Capital/External).
* System calculates the total, shows paid and remaining.
* Save operation:
  + Goods are entered directly into the first warehouse (3).
  + Financial entry is recorded (in 1).
* **8) Constraints and Validations**
* No modification to price/weight after the operation is linked to Filtering/Shipping/Sales. (Requires settlement or reversal entry).
* Total paid ≤ Total (except in case of “advance larger than expected price” → recorded as supplier credit balance).
* Currency: exchange rate is taken only from Settings (10).
* Every operation must be linked to an Order ID.
* **9) Permissions (Users & Roles)**
* **Purchasing Officer:** enters operations.
* **Admin/Finance:** can approve or modify in sensitive cases:
  + Edit price after entry.
  + Settle advances.
  + Enter returns.
* **Workers/Warehouse/Sales:** only see subsequent results (do not see supplier price details).
* **10) Purchases Reports**
* Supplier report: total purchases per supplier + paid + remaining + advances.
* Average price per kg per supplier (to compare with quality).
* Currency report: value in USD + value in ETB + exchange rate used.
* Report by period (Month/Quarter/Year).
* Report of filtering ratio per supplier (from linking purchases with filtering results in 3).
* **11) KPIs (Key Performance Indicators)**
* Average purchase price/kg (per supplier/per order).
* Percentage of purchases funded by Capital vs. External.
* Average payment period (between purchase date and full payment date).
* **12) Notifications**
* When entering a price higher than supplier’s average by X% (e.g., 15%).
* When an advance remains unsettled for more than a period (e.g., 30 days).
* When there is an unprocessed return.
* **13) Edge Cases (Special Cases)**
* Purchase without operating balance: recorded normally (External funding).
* Partial payment in USD and partial in ETB: system allows split payment (each part with its own currency and exchange rate).
* Cancel purchase before entering warehouse: line is cancelled with reversal entry for funding (if any).
* **14) Operational Checklist**
* SupplierID registered.
* Weight/price per kg/total correct.
* Funding source selected (Capital/External).
* Exchange rate used saved.
* Paid and remaining amounts clear.
* Operation linked to Order ID.
* Goods entered directly into first warehouse.
* **15) Practical Example**
* Purchase: 200 kg from Supplier X.
* Price: 600 birr/kg.
* Exchange rate in settings: 60 birr = 1 USD.
* Cost in USD: 200 × (600 ÷ 60) = 2,000 USD.
* Paid 1,200 USD from Capital, remaining 800 USD Credit.
* **System recorded:**
* CapitalOut 1,200.
* Supplier balance 800 USD.
* Entry of 200 kg into first warehouse.
* **(3) Warehouse**
* **1) Definition and Purpose**
* Managing inventory through **two warehouses**:
* **First warehouse** (country of purchase/preparation): receives goods directly after purchases (2), where condition is decided, filtering is applied, and preparation for shipping is done.
* **Final warehouse** (destination country): receives only the clean quantity after shipping + inspection (4), ready for sale (6).
* **Goal:** Accurate visibility of goods’ condition at every moment (ready to ship / awaiting filtering / non-clean / ready for sale), control of transitions between statuses, and linking everything with orders and costs.
* **2) Data Structure (Main Fields)**
* **A) Stock Line**
* order\_id (order number).
* purchase\_id (purchase line coming from stage 2).
* supplier\_id (for reports, with Supplier ID).
* warehouse (FIRST / FINAL).
* status (one of):
  + AWAITING\_DECISION (default after purchase, but we will display all purchases immediately; decision taken later by clicking).
  + READY\_TO\_SHIP (clean, no filtering needed, prepared).
  + AWAITING\_FILTER (decided it needs filtering but not yet executed).
  + NON\_CLEAN (result of filtering non-clean—local sale).
  + READY\_FOR\_SALE (entered final warehouse after inspection—clean ready for sale).
* **Weights in kg:**
* qty\_kg\_total (original purchased weight of line).
* qty\_kg\_clean (remaining/available clean weight in this line).
* qty\_kg\_nonclean (remaining/available non-clean weight).
* qty\_kg\_reserved (reserved for shipping before confirmation—optional).
* **Containers/Cartons (optional for display):**
* cartons\_count (updated at packing).
* **Cost tracking (USD):**
* unit\_cost\_clean\_usd (cost per kg of clean after filtering).
* Non-clean value = always zero (fixed rule as per your request).
* **Log / Timestamps:**
* created\_at, status\_changed\_at, filtered\_at, packed\_at.
* **B) Filter Record**
* purchase\_id (line fully filtered).
* Input: input\_kg = original purchase weight.
* Outputs: output\_clean\_kg + output\_nonclean\_kg (must ≤ input).
* Cost redistribution:
  + All purchase cost loaded onto clean only (non-clean = 0).
  + Clean unit cost rises compared to pre-filter (because clean weight < total).
* filter\_yield% = output\_clean\_kg / input\_kg.
* Linked with supplier and order for quality reports.
* **C) Packing/Prep Record**
* purchase\_id or stock\_line\_id.
* Number of empty cartons used, number of wraps/labels…
* Deduct these from supplies stock (stage 5), calculate packing cost per carton, and link to order.
* **3) Inventory Appearance after Purchase**
* Each purchase line from (2) appears automatically in first warehouse “Purchases list” (line by line).
* Next to each line: button to select status:
  + No filtering needed → Ready to Ship (moves to “Ready to Ship”).
  + Needs filtering → Awaiting Filter (moves to “Awaiting Filter”).
* Important: status selection is not forced at purchase; decision is made later in warehouse screen after physical review.
* **4) Filtering (Step by Step)**
* Executed on whole line (purchase from one supplier line).
* When pressing “Execute Filter” on line in “Awaiting Filter”:
  + System shows: input = purchase weight.
  + User enters: clean (kg) + non-clean (kg).
  + Validation: clean + non-clean ≤ input.
* On save:
  + Filter Record stored with filter\_yield%.
  + Cost redistribution: all purchase cost → clean only ⇒ clean cost rises (correct because less clean weight).
  + Non-clean = zero cost (any sale later = full profit).
* Stock movement:
  + Clean part moves to “Ready to Ship.”
  + Non-clean part moves to “Non-Clean – Local Sale.”
* Reports store filter ratios per supplier/order (for quality analysis).
* **Quick numeric filtering example:** Purchase 1,000kg for $10,000 ($10/kg). Filter result: clean 820kg + non-clean 180kg. Redistribute $10,000 over 820kg ⇒ clean cost ≈ $12.20/kg. Non-clean = $0 cost. Any non-clean sale = 100% profit.
* **5) Sections (Tabs) in First Warehouse**
* First warehouse screen divided into tabs/cards:
  + Purchases list (all from stage 2 without status yet).
  + Awaiting Filter.
  + Ready to Ship (clean & prepared).
  + Non-Clean – Local Sale.
* **Transfer behavior (per your request):**
* After status chosen, line disappears from original list and moves to correct tab.
* Top of each tab: total kg in that status.
* Filter/search tools: by supplier/date/order/quality/status.
* **6) Packing (Preparation for Shipping)**
* When moving line to Ready to Ship, worker executes:
  + Confirm weight to ship.
  + Cartons: enter carton count used (8kg/20kg per market, for packing count only—weight is in kg).
  + Labels/wraps: enter numbers.
* System automatically deducts from supplies stock (5), calculates packing cost/carton, and links cost to order.
* Then line stays “Ready to Ship” until shipping confirmation from (4).
* **7) Link with Shipping (4)**
* From Ready to Ship tab:
  + When creating shipping leg and selecting quantity (4), system marks it “Reserved for shipping” (optionally via qty\_kg\_reserved). Actual deduction happens only on shipping confirmation (end of leg in 4).
* On arrival + inspection:
  + Only clean enters final warehouse as READY\_FOR\_SALE.
  + Damaged recorded as operating loss, not added to stock.
* **8) Sales from Warehouse (Link with 6)**
* **First warehouse:** only non-clean may be sold (local sale).
  + Cost = zero → any sales amount = extra profit.
  + Default sales unit in (6) = cartons (8/20kg); if needed, enable custom option for non-clean sale by kg.
* **Final warehouse:** sale of clean (8/20kg cartons) per (6).
* **9) Controls & Validations**
* Filtering per line only: no partial filtering of same line.
* Quantity checks:
  + In filtering: clean + non-clean ≤ input.
  + In shipping: cannot reserve/ship more than available in “Ready to Ship.”
  + In sales (non-clean): cannot sell more than available in “Non-Clean.”
* Prevent transfer to final before inspection.
* Lock line when tied to confirmed shipment — later changes need sensitive policy (8) with manager approval + settlement.
* **10) Accounting Integration (Value & Cost Movement)**
* After purchase (2): line carries purchase cost/kg (USD).
* After filtering: cost redistributed to clean only ⇒ clean cost/kg rises (logical since non-clean is zero).
* Packing (cartons/labels/wraps): cost charged to order as packing/carton from (5).
* Non-clean: zero cost → any sale = pure extra profit.
* Clean in final: on sale, profit calculated on clean cost after redistribution + share of packing.
* **11) Roles & UI**
* **Warehouse:** sees only warehouse screens, changes statuses, executes filtering, enters packing numbers. Cannot see prices/costs (per policy 8).
* **Purchasing:** sees purchases list only, no warehouse status control.
* **Sales:** sees only stock available to sell (non-clean in first, clean in final).
* **Admin/Finance:** full view + permissions for adjustments/reversals.
* **Suggested interface:**
* Tabs with sections + summary bar at top showing total weight.
* Quick action buttons beside each line: choose status / execute filter / pack.
* “Reserved for shipping” badge on items linked with a shipping leg in progress.
* **12) Reports & KPIs (Warehouse-Specific)**
* **Balances by status & location:**
* First warehouse: (Awaiting Filter / Ready to Ship / Non-Clean).
* Final warehouse: (Ready for Sale).
* **Filtering ratios:** per supplier/order (filter\_yield%). **Packing cost/carton:** total supply consumption ÷ carton count. **Stock aging (optional):** time items remain in each status (Awaiting Filter / Ready to Ship).
* **KPIs:**
* Filter Yield % per supplier/order.
* Clean Unit Cost (post-filter).
* Non-Clean stock (kg) and value (always zero in accounting) + later sales revenue as extra profit metric.
* **13) Notifications**
* Item Awaiting Filter more than X days.
* Item Ready to Ship not linked to shipping for X days.
* Low supply stock (cartons/labels/wraps) during packing (from 5).
* Mismatch on inspection (4): clean + damaged ≠ expected.
* **14) Edge Cases & Fixes**
* **Change decision (Needs Filter/No Filter):**
  + Allowed: Ready to Ship → Awaiting Filter if not yet tied to shipment.
  + After confirmed shipment: needs settlement + manager approval (8).
* **Return to supplier after save:**
  + Create Supplier Return (linked to purchases), deduct weight from first warehouse, settle values with supplier (2).
* **Weight differences:**
  + If clean+non-clean < input: record difference as “Processing Loss” (optional) with approval.
* **Cancel filter mistake:**
  + Create Reverse Filter entry to restore line to pre-filter, then re-enter correct values (sensitive permission).
* **15) Operational Checklist**
* All purchase lines appear in first warehouse.
* Assign status for each line (no filter / needs filter).
* Enter filter result (clean + non-clean) per line and validate totals.
* Review cost redistribution → clean only.
* Execute packing and deduct supplies (generate packing cost/carton).
* Move clean items to Ready to Ship.
* At shipping: reserve, then deduct on confirmation (4).
* On arrival: enter inspection (clean/damaged), move clean to final (4).
* Sell non-clean from first (6) when needed.
* **16) End-to-End Example (Real Numbers)**
* **Purchase (2):** T-005 from supplier S-12: 1,000kg × $10/kg = $10,000. Line enters first warehouse (Purchases list).
* **Status:** Decided Needs Filter → moves to Awaiting Filter.
* **Filtering:** Input = 1,000kg. Outputs: clean = 820kg, non-clean = 180kg. Redistribute cost: $10,000 ÷ 820kg ⇒ clean cost ≈ $12.20/kg. Moves: 820kg → Ready to Ship, 180kg → Non-Clean.
* **Packing:** Prepared 600kg clean: used 30 cartons (20kg), 30 labels, 10 wraps. Deducted from supplies (5), calculated packing cost/carton, linked to order. Line in Ready to Ship now shows 600kg available for shipping reservation.
* **Shipping (4):**
* Created shipping leg 600kg → reserved. Confirmed shipment → deducted 600kg from first warehouse. On arrival inspection: 595kg clean + 5kg damaged. 595kg enters final warehouse “Ready for Sale.”
* **Sales (6):** Sell 100 cartons × 8kg = 800kg? (No—only 595kg available clean). Instead, sell 25 cartons × 20kg = 500kg. Sales only from final warehouse for clean. Sell non-clean (180kg) locally from first warehouse (by carton/kg per settings) ⇒ pure extra profit.
* Perfect ✅ — here’s your section **(4) Shipping** translated into clear, precise English, preserving every field, step, rule, and example.
* **(4) Shipping**
* **1) Definition and Goal**
* Manage the transfer of goods from the **first warehouse (country of purchase/prep)** to the **final warehouse (destination country)** via **one or multiple legs**: Country 1 → Country 2 → Country 3 → (rarely) Country 4.
* Goal: Track **weights (Net/Carton/Gross/Chargeable)**, **costs per leg**, **arrival expenses**, and the **final inspection**, linking all of that to the order, inventory, and accounts.
* **2) Data Model (Entities & Fields)**
* **A) Shipment / Order**
* order\_id (order/shipment number).
* Overall status: PLANNED → IN\_TRANSIT → ARRIVED\_PENDING\_INSPECTION → INSPECTED\_CLEARED.
* Warehouse link: from **FIRST** to **FINAL**.
* Document archive (optional): airway bill/carrier invoice… (for recall from reports/timeline).
* **B) Shipping Leg**
* leg\_id, order\_id.
* **From/To**: departure country, arrival country.
* **Dates**: departure date (planned/actual), arrival date (planned/actual).
* **Shipping company/carrier**.
* **Weights**:
  + net\_kg (net)
  + carton\_kg (carton/packing weight)
  + gross\_kg = net\_kg + carton\_kg
  + chargeable\_kg (billing weight if different from gross)
* **Carton count**.
* **Pricing** (one calculated from the other):
  + rate\_per\_kg (shipping rate per kg)
  + leg\_base\_cost (base shipping amount)
* **Transfer commission on the shipping amount** (optional %), calculated as base\_cost × % and added to leg cost.
* **Payment method**: cash / advance / credit.
* **Funding source**: from working capital / external.
* **Currency**: USD or ETB (auto-convert to USD using the central rate from settings (10); rate stored historically).
* Leg status: DRAFT → CONFIRMED\_DEPARTURE → ARRIVED.
* **Pricing rule:** If chargeable\_kg exists, the system uses it as the basis (Rate × Chargeable); otherwise it uses gross\_kg.
* **C) Arrival Costs**
* Linked to order\_id (usually after the **last leg** arrives).
* Components (customizable):
  + Broker / customs clearance
  + Cargo handling
  + Delivery / domestic transport
  + Storage (if any)
  + Other / miscellaneous expense
* For each line: amount, currency (USD/ETB), **funding source** (capital/external), date, note.
* Added **as part of the order’s total shipping cost**.
* **D) Arrival Inspection**
* Linked to order\_id (**shipment level**, not supplier level).
* Input:
  + **Clean quantity (kg)**
  + **Damaged quantity (kg)**
* Required condition: **clean + damaged = expected to arrive** (actual shipment Net).
* Inspection results:
  + **Clean**: recorded into the final warehouse as READY\_FOR\_SALE.
  + **Damaged**: recorded as an **operating loss** (not added to inventory).
* **3) Shipping UI — Three Sections (as requested)**
* **(1) Add Shipping Leg**
* Add a leg and fill the **core data** (as above).
* **Two-way auto-calculation**:
  + If you enter **Rate/kg** + **Chargeable or Gross** → system calculates **leg total**.
  + If you enter **leg total** + **Chargeable or Gross** → system calculates **Rate/kg**.
* **Confirm departure** from the origin country:
  + Immediately **deducts shipped net quantity** from the **first warehouse** (“Ready to Ship” section).
  + Sets leg status to CONFIRMED\_DEPARTURE, and order status to IN\_TRANSIT.
* You can **add an additional leg** to the same order (“+ Additional Leg”).
* Leg arrives → status ARRIVED (but order remains ARRIVED\_PENDING\_INSPECTION until inspection is done).
* **(2) Additional Arrival Expenses**
* From this screen, pick a **shipment that has arrived** (or the **final leg** in the chain) and enter expense lines (Broker/Cargo/Delivery… etc.).
* **Currency** USD/ETB with automatic conversion to USD.
* **Funding source**: capital/external.
* All these lines are **aggregated** into the **order’s total shipping cost**.
* **(3) Inspection**
* After the **last leg** arrives: enter **clean + damaged** (kg).
* On save:
  + **Clean**: enters the **final warehouse** as READY\_FOR\_SALE.
  + **Damaged**: recorded as **operating loss** (not inventory).
  + Order status changes to INSPECTED\_CLEARED.
* **Note:** The inspection report **does not mention the supplier** (per your request) because it is at **shipment level**.
* **4) Calculation Rules (Formulas)**
* **A) Cost of Each Leg**
* **Weight basis for calculation**:
  + If chargeable\_kg > 0 → basis is **chargeable**.
  + Else → basis is **gross\_kg**.
* **Computation**:
  + If rate\_per\_kg is entered:
    - leg\_base\_cost = rate\_per\_kg × basis\_kg
  + If leg\_base\_cost is entered:
    - rate\_per\_kg = leg\_base\_cost ÷ basis\_kg
* **Add transfer commission (optional)**:
  + transfer\_fee = leg\_base\_cost × (commission %)
  + leg\_total\_cost = leg\_base\_cost + transfer\_fee
* **B) Order Total Shipping Cost**
* total\_shipping\_cost = Σ(leg\_total\_cost for all legs) + Σ(arrival\_costs)
* **C) Shipping Cost per kg**
* Before inspection:
  + shipping\_cost\_per\_kg\_planned = total\_shipping\_cost ÷ total\_basis\_kg
  + where total\_basis\_kg = sum of basis weights (Chargeable or Gross) across legs.
* **D) Landed Cost on Clean Quantity**
* After inspection, to show **final average cost/kg** and **computed 8-kg carton cost** as requested:
* landed\_cost\_total = purchase\_cost\_allocated\_to\_shipped\_clean // from (3) after redistributing purchase cost to clean only
* + total\_shipping\_cost // (all legs + arrival costs)
* + packing\_cost\_allocated\_to\_shipped // from (5) packing/carton cost
* landed\_clean\_kg = quantity\_clean\_after\_inspection // clean only
* avg\_landed\_cost\_per\_kg = landed\_cost\_total ÷ landed\_clean\_kg
* carton8\_landed\_cost = avg\_landed\_cost\_per\_kg × 8
* // (optional) carton20\_landed\_cost = avg\_landed\_cost\_per\_kg × 20
* **Damaged = operating loss**:
  + Can be shown in a separate report:
    - operational\_loss = avg\_landed\_cost\_per\_kg × damaged\_kg (or by components: (clean purchase/kg + shipping/kg + packing/kg) × damaged).
* **Important:** We keep the same methodology from (3): **purchase cost is redistributed to “clean” only** after filtering. Here in shipping we add **shipping + arrival costs + packing**, then divide by **clean after inspection** to get **final average cost/kg** and **8-kg carton cost**.
* **5) Controls & Validations**
* **Confirming shipment** from origin instantly deducts quantity from the first warehouse (no selling from that source quantity after confirmation).
* **No sale** from the shipment at destination before inspection.
* **Weight consistency**:
  + gross\_kg = net\_kg + carton\_kg.
  + clean\_after\_inspection + damaged = expected\_arrival\_net\_kg.
* **Chargeable**, if entered, should be ≥ Gross (usually); if lower → warning.
* **No duplication**: the same quantity cannot be tied to two legs at the same time.
* **Arrival discrepancies**: if received quantity is less/more than expected, a **settlement** is required with manager approval (policies 8).
* **Currency & FX**: ETB auto-converted to USD (from settings 10), with historical rate saved.
* **6) Accounting Integration**
* **Funding for legs and expenses**:
  + If “from working capital” → **CapitalOut** in (1).
  + “External” → does not affect capital, but is included in the **order’s shipping cost**.
* **Transfer commission %**: recorded as part of the **leg cost**.
* **Arrival costs**: recorded as shipping expenses for the order and included in total\_shipping\_cost.
* **Inspection/damaged**: recorded as a separate **operating loss** (not inventory).
* **7) Roles & UI**
* **Operations/Shipping**: create/edit legs, confirm shipment, enter arrival, record arrival costs, perform inspection.
* **Warehouse**: sees reservations/deductions from the first warehouse, and receipt of clean in the final warehouse.
* **Finance**: sees costs, capital funding, exchange rates, and handles settlements.
* **Admin**: approvals for sensitive settlements (large variances, write-offs, editing a leg after confirmation).
* **Main interfaces:**
* “Shipping” screen with three tabs: **[Shipping Legs] – [Arrival Costs] – [Inspection]**.
* On the right, an order summary card: **total legs/total cost/total Chargeable/last updated**.
* Quick actions: **+ Leg**, **+ Arrival Expense**, **Perform Inspection**.
* **8) Reports & KPIs**
* **Reports**
* **Leg details**: from/to, weights, Rate/kg, pricing basis, leg cost, transfer commission.
* **Arrival costs**: line details, amounts, and funding sources.
* **Shipment summary per order**: total shipping cost (all legs + arrival).
* **Chargeable vs Gross**: weight difference and impact on cost.
* **Shipping cost/kg** and **cost per 8-kg carton**.
* **Transit time**: from departure to arrival per leg (Timeline).
* **KPIs**
* **Average shipping cost/kg**.
* **Shipping as % of total landed cost**.
* **Average transit days** per leg/carrier.
* **Chargeable vs Gross variance** (how much % uplift charged).
* **9) Notifications**
* **Large Chargeable variance** vs Gross.
* Shipment **arrived but not inspected** within X hours/days.
* **Arrival costs missing** after arrival.
* Editing/cancelling a leg after confirmation → requires manager approval (8).
* **10) Edge Cases & Fixes**
* **Intermediate leg** arrived but last leg not yet: order remains IN\_TRANSIT; no receipt to final warehouse before inspection.
* **Arrival weight differences**: system opens a “Settlement” with three options:
  + Accept received weight and record the difference as shipping loss.
  + Request carrier correction (set to Pending).
  + Write-off/correction with manager approval.
* **Advance payment to carrier** before final invoice: recorded as **Advance** on shipping (funded or not from capital), settled when the leg’s final cost is entered.
* **Cancel a leg** before confirmation: deletion allowed. After confirmation: **reverse entry** + manager approval.
* **Change carrier or intermediate destination** while in transit: modify leg while keeping a **log** and manager signature.
* **11) Operational Checklist**
* Create/link shipment to the correct order number.
* Add a shipping leg and fill **Net/Carton/Gross/Chargeable** accurately.
* Verify **Rate ↔ Total** (auto-calc).
* Choose **funding source** (capital/external) and payment currency.
* **Confirm departure** from origin (deduct from first warehouse).
* At final arrival: enter **arrival costs** (if any).
* Perform **inspection** (clean + damaged = expected).
* Move clean to final warehouse; book damaged as loss.
* Review **shipment summary**: average cost/kg, **8-kg carton cost**.
* Lock the shipment (optional) after inspection and reports are complete.
* **12) Full Numeric Example**
* **Inputs after filtering in (3):**
* **820 kg clean** available to ship (purchase cost redistributed to clean, now e.g., **$12.20/kg**).
* You decide to ship **600 kg** in one shipment.
* **(1) Single shipping leg**
* Net = 600 kg
* Carton = 20 kg → Gross = 620 kg
* Chargeable = **700 kg** (carrier billed 700)
* You entered rate\_per\_kg = \$1.8 →
  + leg\_base\_cost = 1.8 × 700 = \$1,260
* Transfer commission 2% (optional):
  + transfer\_fee = 1,260 × 0.02 = \$25.2
* **Leg cost = $1,285.2**
* **Confirm departure** → deduct 600 kg from the first warehouse.
* **(2) Arrival costs**
* Broker = $90, Delivery = $60 → **$150**
* **Total order shipping cost = 1,285.2 + 150 = $1,435.2**
* **(3) Arrival inspection**
* Received: **595 kg clean** + **5 kg damaged** (equals the expected Net 600).
* **Clean (595 kg)** enters final warehouse **Ready for Sale**.
* **Damaged (5 kg)** is recorded as an **operating loss**.
* **(4) Final average cost**
* From (3): **clean purchase cost before shipping ≈ $12.20/kg**.
* Purchase cost for the shipped portion (600 kg) = 600 × 12.20 = \$7,320.
* **Total shipping & expenses = $1,435.2**.
* (If you have packing/carton from (5), add it here; assume 0 to simplify.)
* landed\_cost\_total = 7,320 + 1,435.2 = \$8,755.2
* avg\_landed\_cost\_per\_kg = 8,755.2 ÷ 595 ≈ \$14.72/kg
* **Computed 8-kg carton cost** = 14.72 × 8 ≈ \$117.76 per carton
* **Damaged loss** ≈ 14.72 × 5 ≈ \$73.6 (shown as an operating loss line).
* This is how the “final totals” view will show **average cost/kg** and **8-kg carton price** exactly as you requested.
* All set — your **Shipping** stage is now fully specified with **multi-leg flows**, immediate deduction on confirmation, **separate arrival costs**, **shipment-level inspection**, working-capital accounting, clear reports, and **KPIs** highlighting Chargeable impact and landed pricing.
* **(5) Operating Expenses**
* **1) Definition and Goal**
* Operating expenses are everything that is **not** a merchandise purchase (Stage 2) and **not** a direct shipping cost (Stage 4), and specifically include:
* Labor (loading, filtering, cartonizing/boxing… etc.).
* Rents (warehouses, temporary spaces).
* Operating/packaging supplies (cartons, tape, labels… with the ability to add new fixed items).
* **Goal:**
* Prove the actual cost of the order accurately (Costing) by linking every expense to the order whenever possible.
* Manage supplies inventory (purchase → balance → consumption at packing).
* Surface indicators such as packaging cost per carton, labor cost per kg, and the ratio of expenses to landed cost.
* **2) Data Model (Entities & Fields)**
* **A) Expense Entry**
* **expense\_id**
* **type:** LABOR / RENT / SUPPLY\_PURCHASE / SUPPLY\_CONSUMPTION / OTHER
* **description:** free text (example: “Filtering workers T-007”, “Addis warehouse rent March 2025”).
* **amount + currency** (USD or ETB) → Automatic conversion to USD at the central FX rate from Settings (10), with historical FX stored.
* **date**
* **funding source:** FROM\_CAPITAL / EXTERNAL (From capital deducts in (1); external does not reduce the balance.)
* **link reference:**
  + order\_id (an order) — when possible,
  + or **order\_ids[]** to allocate the expense across multiple orders (percentages or fixed amounts).
* **attachments (optional):** photo of invoice/receipt.
* **B) Labor**
* worker\_id, worker name, work type (filtering/loading/cartonizing/…).
* Reference quantity (optional): hours/daily wage/per output.
* Payment amount, date, currency, funding source.
* Link to one order or distribute across multiple orders by percentages.
* **C) Rent**
* Lessor/entity, period (from/to), location (warehouse/city).
* Amount, currency, funding source.
* Allocation method to orders (details below).
* **D) Supplies**
* **Catalog of fixed items:**
* For each item: item\_code, name (cartons/tape/labels/…), unit of measure (piece/pack/large carton), conversion factor if any (example: 1 pack = 50 labels).
* **Supply Purchase (SUPPLY\_PURCHASE):**
* Item, quantity purchased, unit price, total cost, currency, funding source.
* **Effect:** increase supplies inventory balance.
* **Supply Consumption (SUPPLY\_CONSUMPTION):**
* Not usually entered manually; generated **automatically** during packing in the warehouse (Stage 3) when the worker enters the numbers of cartons/tape/labels used.
* **Effect:** deduct from supplies inventory + charge packaging cost to the order.
* **Costing method for supplies (adopted):** Purchase → Supplies inventory; upon consumption → move cost from supplies inventory to the order’s cost. (Supports accurate “packaging cost per carton”.) (You can switch to treating the purchase as a direct expense instead of inventory from Settings for those who prefer it; default is inventory to control cost per carton.)
* **3) Workflow (Step-by-Step)**
* **A) Labor**
* Record/select the worker (Worker ID, name, work type).
* Enter payment: amount, currency, date, funding source (from capital/external).
* Link the payment:
  + To a specific order (e.g., filtering T-008), **or**
  + Distribute it over several orders (by % shares, by kg quantities, or by hours per order).
* Save → reflected in the cost of the order(s), and accounting impact is recorded (capital deduction if chosen).
* **B) Rent**
* Enter the rent contract/payment: entity, period, location, amount, currency, funding source.
* Allocate rent to orders (choose one):
  + **Weighted allocation** (by kg shipped/prepared during the period).
  + **Time-based allocation** (by days of warehouse use per order if documented).
  + **Manual allocation** (full control).
* Save → system generates an allocation entry automatically: each order is charged its share of rent.
* **C) Supplies**
* **Supplies purchase:** enter item, quantity, unit price, currency, funding source → increases supplies inventory balance.
* **Consumption (automatic):** from Warehouse (3) at the “Packing for Shipping” step:
  + Worker enters: number of cartons/tape/labels used.
  + System deducts from supplies (FIFO or Weighted Average) and charges the cost to the order.
  + Automatically computes packaging cost per carton and stores it for display in reports and Stage (4) within the landed cost.
* **4) Controls & Validations**
* **Currencies:** any ETB entry is auto-converted to USD at the central exchange rate (10), with the historical FX saved.
* **Funding source:** must be specified for every expense (from capital/external).
* **Supplies:**
  + No consumption beyond balance (prevent negatives).
  + Units of measure and conversion factors are mandatory and precise (example: 1 pack = 50 labels).
  + If consumption exceeds available balance for an item, show “Supplies stock shortage” alert and block save unless manager approval (Policy 8).
* **No deletion** of an expense posted to an order’s cost (has costing impact): requires documented reversal/settlement.
* **Rent allocation:** total allocation = 100% or equals the full amount.
* **Abnormal supplies purchase price** vs last average price for the item → warning (e.g., ±20%).
* **Period closing:** no entries dated before the closing date unless with manager permission + reason.
* **5) Accounting Integration**
* **Working Capital (1):**
  + If funded from capital → create **CapitalOut** linked to expense\_id (or split across multiple orders).
  + If external → does not reduce the capital balance, but still recorded as an expense linked to the order(s).
* **Warehouse (3):**
  + Consumption is generated automatically from packing (cartons/tape/labels) and deducts from supplies balance.
* **Shipping (4):**
  + Packaging cost per carton enters the final **landed cost** when calculating average cost/kg and 8-kg carton price (as detailed in 4).
* **Sales (6):**
  + Expenses (especially packaging and labor) affect order profitability shown in Sales/P&L reports.
* **Reports (9):**
  + Operating expenses appear separately from purchases and shipping in the order and period summaries.
* **Settings (10):**
  + Manage currency and exchange rate.
  + Enable/disable “store supplies as inventory” vs “recognize as immediate expense” (default: inventory).
* **6) Roles & UI**
* **Finance:**
  + Enter/edit all expense types.
  + Perform settlements/reversals.
  + Configure rent allocation policies.
* **Warehouse:**
  + Does not enter expenses; only triggers **automatic consumption** during packing and answers “How many cartons/tape/labels did you use?”
* **Purchasing/Shipping/Sales:**
  + Limited visibility as needed (without detailed prices if a hide policy is set in 8).
* **Admin:**
  + Approvals for: deletions/reversals after posting, consumption with stock shortage, back-dated entries after close.
* **Proposed UI**
* Tabs: **[Labor] [Rent] [Supplies] [Aggregated]**.
* Supplies balance board: shows current balance per item + reorder point.
* Quick forms to add labor payments, link to orders, and allocate.
* Rent allocation wizard with three methods (weighted/time/manual).
* **7) Reports & KPIs**
* **Detailed Reports**
* Expense breakdown by type (Labor/Rent/Supplies/Other) and period.
* Expenses per order: how much labor/rent/packaging each order carried.
* Supplies balance (opening, purchases, consumption, closing).
* Packaging cost per carton (derived from total consumed supplies cost ÷ number of cartons).
* Supplies purchase price trend per item.
* “Expense per kg” comparison across orders (Labor/Packaging per kg).
* **Core KPIs**
* Packaging cost per carton (target vs actual).
* Labor cost per kg per order.
* Operating expenses as a % of final landed cost.
* Allocation accuracy (how many orders unallocated/with manual allocation).
* Compliance with spend limits (vs budgets if enabled).
* **8) Notifications**
* Supplies stock below reorder point.
* Supplies purchase price deviates from average by ±X%.
* General expenses not allocated to orders by period end.
* Late labor payments (if payroll scheduling is enabled).
* Rent due soon (reminder before due date).
* Attempt to consume more than stock (block + request manager approval).
* Entry after period close (request reopen/approval).
* **9) Edge Cases & Fixes**
* **Supplies return to supplier:** Deduct from “supplies inventory” and refund value (**CapitalIn** if originally from capital), and remove the effect from order cost if it had been consumed and then unused items were returned.
* **Correct consumption mistake:** Create a **reverse consumption** movement to restore balance, then enter the correct consumption (logged in Audit Log).
* **Change rent allocation method after approval:** Post a **Reverse Allocation**, then re-allocate using the new method (every step documented).
* **Supplies write-off (waste/expiry):** A “Supplies Write-off” movement recorded as a general expense (not tied to an order) with reason.
* **Supplies purchase in two currencies:** Allow splitting the invoice: part USD and part ETB, each portion with its own historical FX.
* **Back-dated entry after close:** Requires manager permission and justification, and appears in the “Post-close adjustments” report.
* **10) Operational Checklist**
* Specify the **funding source** for every expense (capital/external).
* Automatically document the **FX rate** for every ETB entry.
* Link expenses to orders (or allocate them accurately by percentages).
* Supplies purchase → increases supplies inventory.
* Supplies consumption **only via packing** (Stage 3) and by the **actual quantities used**.
* Set **reorder points** for each supplies item.
* Review **packaging cost/carton** and **labor/kg** in reports.
* Prevent over-consumption (or document approvals).
* Close the period after review; no changes without approval.
* **11) Realistic Numeric Examples**
* **Example (A): Filtering labor and allocation**
* You paid **900 USD** to three filtering workers over a week.
* They worked on two orders: **T-010 (600 kg clean)** and **T-011 (400 kg clean)**.
* You chose **weighted allocation**: 60% to T-010 and 40% to T-011.
* Posting: **540 USD** to T-010 and **360 USD** to T-011.
* Funding source: **from capital** → a **CapitalOut 900** is created in (1).
* **Example (B): Monthly rent and its allocation**
* Addis warehouse rent (March 2025) = **1,200 USD**, **external** (doesn’t reduce capital).
* During March, three orders passed with weight shares: **50%**, **30%**, **20%**.
* System generates an allocation entry: **600 / 360 / 240 USD** across the three orders.
* **Example (C): Supplies purchase + consumption**
* You purchased **1,000 empty cartons × 0.6 USD = 600 USD** (from capital → **CapitalOut 600**).
* Consumption at packing (Stage 3) for order **T-012**:
  + Used **40 cartons + 40 labels + 10 wraps**.
  + System deducted quantities from supplies inventory and charged their cost to **T-012**.
  + Reports show: **packaging cost per carton = (total consumed supplies cost ÷ number of cartons consumed)**.
* **Example (D): Supplies purchase in Birr**
* You purchased **tape 200 packs × 150 ETB = 30,000 ETB**.
* Central FX (Settings 10) that day = **60 ETB/USD** → **500 USD**.
* Entry is recorded in USD with historical ETB + Rate saved.
* Later consumption deducts from balance and charges cost in USD.
* **12) Where do we enter “how many cartons/tape/labels”?**
* In the **warehouse (Stage 3)** at **“Packing for Shipping.”** A simple form appears to the worker:
* “How many cartons did you use? How many labels? How many tape rolls?”
* Save → it automatically deducts from supplies inventory, charges the cost to the order, and updates **packaging cost per carton**.
* **Stage (5) Summary**
* A tight accounting model: we **store supplies as inventory** and **convert them to cost at consumption**.
* Labor and rents can be linked directly or smartly **allocated** across orders.
* Currencies: base **USD**, and **ETB** converts centrally with historical tracking.
* Reports/KPIs show **packaging cost per carton**, **labor per kg**, and **expenses as a share of landed cost**.
* Notifications keep discipline (supplies shortage, abnormal prices, unallocated expenses…).
* **(6) Customers & Sales**
* **1) Definition and Goal**
* Manage the sale of **clean goods** from the **final warehouse**, and the sale of **non-clean goods** from the **first warehouse** (locally).
* Support invoicing from **one order by default**, with the ability to **split an invoice across two or more orders** when needed, showing the quantity taken from each order.
* Calculate revenue, accounts receivable, and profitability per order/invoice, and allow **customer returns** that put goods back into the same warehouse and the same order.
* **2) Data Model (Entities & Fields)**
* **A) Customer**
* **customer\_id** (auto or manual numbering).
* **name** (required).
* **type:** Wholesale or Retail.
* **note** (optional).
* As requested: no contact details or city are collected.
* **B) Invoice / Sale**
* **invoice\_id** (per numbering settings in stage 10).
* **customer\_id**.
* **source\_warehouse:**
  + **FIRST** for non-clean (local) sales.
  + **FINAL** for clean sales (after shipping & inspection).
* **Order selection:**
  + One **order\_id** by default.
  + “+ Add Order” option to split the invoice across multiple orders.
* **Sales unit:** 8-kg or 20-kg carton (set in sales settings, stage 10).
* **Pricing:**
  + price\_per\_carton ↔ total\_amount (enter one, the system calculates the other based on cartons × 8 or 20 kg).
* **Currency:** input in USD or ETB (auto-convert to USD using the central FX from settings 10; historical FX saved).
* **paid\_now**, **remaining (A/R)**.
* **invoice\_date**.
* **attachments** (optional: manual voucher, customer signature).
* **C) Invoice Lines**
* For each line:
* **order\_id** (the order from which the quantity is deducted).
* **unit** (8/20 kg).
* **cartons\_qty** (number of cartons).
* **Auto weight calc:** kg = cartons\_qty × (8 or 20).
* **price\_per\_carton** (may be unified for the invoice or per line—recommended unified unless differentiation is required).
* **source:** FIRST (non-clean) / FINAL (clean).
* **Reference cost:**
  + **Clean:** kg cost after filtering + share of shipping/arrival + share of packaging (from stages 3/4/5) → converted to **Cost/Carton** for margin.
  + **Non-clean:** cost = **zero** (per your fixed rule).
* **D) Receipts**
* **receipt\_id**, **invoice\_id**, **amount**, **currency** (USD/ETB), **historical FX**, **date**, **note**.
* Supports **multiple receipts** for the same invoice (installments).
* **E) Customer Return (Sales Return)**
* **return\_id**, **invoice\_id** (or directly to **order\_id** if it’s a later independent return).
* **cartons\_qty**, **unit** (8/20), **source** (must return to the **same warehouse** the goods came from: FIRST for non-clean, FINAL for clean).
* **Auto calculation of A/R/refund values:**
  + Either **cash refund** (negative against revenue / payout from revenues),
  + Or **offset** against the customer’s later invoices.
* **3) Pricing and Currency**
* Reporting defaults to **USD** pricing.
* If price/payment is entered in **ETB**:
  + Automatic conversion using the central FX from settings (10), **historically fixed** on invoice/receipt.
* **Two-way auto-calc:**
  + Enter price\_per\_carton + cartons\_qty → system computes total\_amount.
  + Or enter total\_amount + cartons\_qty → system computes price\_per\_carton.
* **4) Workflow (Step-by-Step)**
* **A) Create a Customer**
* Enter name, type (Wholesale/Retail), note (if needed).
* Save → customer\_id becomes available for invoices.
* **B) Create a Sales Invoice**
* Select the customer.
* Choose **source**:
  + **FIRST** → sell **non-clean** (local).
  + **FINAL** → sell **clean** (arrived & inspected).
* Select the **order** (one by default).
* If quantity is insufficient: use **“+ Add Order”** and specify quantities from each order (clear per-line allocation).
* Choose unit (8/20 kg).
* Enter number of cartons.
* Enter price/carton (or the total).
* Enter amount paid now (if any) and currency.
* **Save**:
  + System deducts quantities from the specified warehouse/order per line.
  + Creates **A/R** for the remaining amount.
  + Calculates per-line profitability (visible internally to authorized roles only).
* **C) Record a Later Receipt (Collection)**
* Open the invoice or the customer account.
* Enter amount, currency, date.
* Amount is auto-converted to USD and deducted from A/R.
* You can split a receipt across multiple invoices for the same customer (collection allocation).
* **D) Customer Return**
* Choose the invoice or the related order.
* Enter cartons and unit.
* System returns the goods to the correct warehouse (**FIRST** for non-clean, **FINAL** for clean).
* Calculates return value:
  + Reduces revenue and decreases A/R (if still outstanding) or generates a refund voucher.
* Accounting links are posted automatically.
* **5) Controls & Validations**
* **Source & warehouse:** No selling clean from **FIRST**, and no selling non-clean from **FINAL**.
* **Balance:** No sales above available order/warehouse balance.
* **Invoice splitting:** Default is a single order. When splitting, you must specify a clear quantity per order (no ambiguous totals).
* **ETB invoices:** FX rate only from settings (10); stored historically—no retroactive edits.
* **Returns:** Must return to the **same order** and **same warehouse** the goods were issued from.
* **Deletion/Editing:** No deletion of a sales line that has a return/receipt; use **reverse/correct** under the sensitive policy (8).
* **Consistency check:** Sum of invoice lines = invoice total; sum of receipts ≤ total.
* **6) Accounting & Costing**
* **Revenue** = payments (with remaining balance as A/R).
* **Accounts Receivable (A/R)** = amount due from the customer.
* **Cost:**
  + **Clean (FINAL):** line cost =
    - clean kg cost after filtering (from 3)
      * share of shipping & arrival (from 4)
      * share of packaging (from 5) → converted to **Cost/Carton × number of cartons**.
  + **Non-clean (FIRST):** cost = **zero** → full sales amount = **extra profit**.
* **Return:** reduces revenue, restores inventory, and adjusts A/R.
* **Currency:** all **reporting** numbers in USD; each transaction keeps its own historical FX.
* **7) Roles & UI**
* **Sales:** create/edit invoices, record receipts, process returns; sees only their inventory/reports.
* **Warehouse:** does not open invoices, but sees quantity deductions and stock effects.
* **Finance:** sees pricing, profitability, collections, aging, and performs settlements.
* **Admin:** sets sensitive policies (change price after save, reverse lines, reopen periods…).
* **Proposed UI**
* Invoice form on the **left**, and a **stock/order balance board** on the **right** (showing 8/20 carton balances and kg).
* “**+ Add Order**” button adds a new line with order selection and quantity.
* Instant calculations: invoice total, paid, remaining.
* “**Record Receipt**” button below the invoice.
* “**Return**” button on the invoice screen or customer account.
* **8) Reports & KPIs**
* **Customer/Sales Reports**
* Sales detail: by date/customer/order/unit (8/20).
* Customer profile:
  + total purchased,
  + **Top Buyer**,
  + customer with the largest debt,
  + best customer by profitability (highest aggregated margin),
  + most frequent buyer.
* **Aging Report**: 30/60/90 days with customer balances.
* **Sales by order**: how much sold, how much remains, profit margin (based on the detailed cost above).
* **Non-clean report**: quantities sold/remaining and their revenues (all extra profit).
* **KPIs (on the Dashboard and in reports)**
* Average selling price per kg.
* Profit margin % by period/by orders.
* Collection rate (collected ÷ sales).
* Share of non-clean sales out of total revenue.
* Average **Days Sales Outstanding (DSO)**.
* **9) Notifications**
* Selected order doesn’t have enough quantity (pre-save).
* Invoice hits the customer’s **credit limit** (if enabled).
* Overdue receivables: alert at 30/60/90 days.
* Selling price deviates from period/customer average (±X%).
* Return without sufficient balance in the same order/warehouse (block + warning).
* **10) Edge Cases & Fixes**
* **Mixing orders:**
  + Default is to prevent (single order), but when needed use “+ Add Order” and specify exact quantities (full transparency).
* **Price change after save:**
  + Under sensitive policy (8): once only, or with manager approval, or within ±5% cap.
  + System logs the change and recomputes margin.
* **Cancel invoice/line after collection:**
  + No direct deletion; reverse with proper adjustments to collections/A/R (manager approval).
* **Multi-currency receipts for the same invoice:**
  + Allowed; each receipt keeps its own currency & historical FX, and USD totals are shown in summaries.
* **Partial return after a period:**
  + Accepted as long as it goes back to the same order/warehouse; revenue/A/R are corrected.
* **11) Operational Checklist**
* Pick the correct **source** (FIRST for non-clean, FINAL for clean).
* Select the order (single by default; split when needed with explicit per-order quantities).
* Set the unit (8/20) and enter number of cartons.
* Enter price/carton or the total (auto-calc).
* Record **paid now** and FX if currency is ETB.
* Save → deduct stock and create A/R.
* For later collections: add receipts and allocate to invoice(s).
* For returns: same order & warehouse; adjust revenue/A/R.
* Review profitability and **Aging** reports regularly.
* **12) Numeric Examples**
* **Example (1): Clean sale from the final warehouse**
* Order **T-020** has **READY\_FOR\_SALE = 595 kg** (after stage-4 inspection).
* Average landed cost **14.72 USD/kg** (from 4/5/3).
* Wholesale customer: sold **25 cartons × 20 kg = 500 kg**.
* Selling price/carton = **145 USD** → total = **25 × 145 = 3,625 USD**.
* Paid now **2,000 USD**, remaining **1,625 USD** A/R.
* Approximate line margin = revenue − cost:
  + Cost/kg **14.72** → cost of **500 kg ≈ 7,360 USD** (Note: this cost already aggregates purchase, shipping, and packaging computed earlier; if you sell less than the whole shipment, the system allocates cost by quantity sold.)
* Margin appears in the order & period reports (authorized roles only).
* **Accounting note:** the on-screen margin uses the stored **Cost/Carton** at sale time, not a fresh recomputation for the entire shipment each time. This example illustrates the method only.
* **Example (2): Non-clean sale from the first warehouse**
* Non-clean balance: **180 kg** (cost = **zero**).
* You sold **22 cartons × 8 kg = 176 kg** at **20 USD/carton** → total **440 USD**.
* Profit for this line = **440 USD** (full extra profit).
* Remaining **4 kg** non-clean.
* **Example (3): Split invoice across two orders**
* Need **40 cartons × 8 kg = 320 kg** clean.
* **T-021** has **280 kg**, **T-022** has **100 kg**.
* Add two lines on the same invoice:
  + From **T-021**: **35 cartons** (280 kg).
  + From **T-022**: **5 cartons** (40 kg out of 100).
* The invoice shows a clear distribution per line/order.
* **Example (4): Return**
* Customer returns **5 cartons × 8 kg = 40 kg** from a previous (clean) invoice.
* System puts **40 kg** back into the **final warehouse** for the **same T-021**.
* Deduct the value of 5 cartons from revenue and **reduce A/R** or create a **refund voucher**.
* **Summary**
* **Source controls the product type:** **FINAL** for clean, **FIRST** for non-clean.
* Default invoice uses **one order**, with the option to split when needed by specifying quantities per order.
* **Price/carton ↔ total** auto-computed; currencies handled centrally with historical FX.
* **Non-clean = zero cost** → any sale is extra profit.
* **Customer returns** restore stock to the same order/warehouse and correct revenue/A/R.
* Strong reports: **Aging**, **Top Buyers**, best profitability, non-clean share of revenue, and accurate margins per order and period.
* **Core Pillars (fixed as agreed)**
* **Withdrawals come only from Revenues** (not from Working Capital).
* **Reinvesting Revenues:** deduct from the Revenue balance **and** record transfer fees as an **Operating Expense (5)** **and** increase **Working Capital (1)** by the transferred amount.
* The **Revenues screen** always shows two numbers:
  + **Accounting revenue** = total sales (net of returns) as in reports.
  + **Aggregated withdrawable revenue balance** = actual cash collections (after returns/refunds) − (withdrawals + transfers + fees). This is the balance used for withdrawals/transfers to capital.
* **1) Definition and Goal**
* The purpose of this stage is to manage the **cash inflow** from sales (customer collections) and then organize its use in two ways:
* **Partner Withdrawals** — distributions from the accumulated cash balance.
* **Reinvesting** part of this balance into **Working Capital** to finance new purchase/shipping cycles.
* **Goal:** Full transparency between “how much we sold” (accounting) and “how much we collected” (cash), and what was used (withdrawal/transfer/fees), with tight accounting linkage.
* **2) Data Model (Entities & Fields)**
* **A) Revenue Ledger**
* **rev\_entry\_id**
* **date**
* **type:**
  + **CustomerReceipt** (customer collection: cash/bank transfer)
  + **CustomerRefund** (refund to customer on return/correction)
  + **Withdrawal** (partner withdrawal)
  + **ReinvestOut** (amount transferred to Working Capital)
  + **TransferFee** (fee related to reinvest or any transfer out of revenues)
  + **Reclass/Reverse** (reversal/correction)
* **amount (USD)**, with original currency if ETB, and **historical FX rate** (from Settings 10).
* **reference:** invoice\_id / receipt\_id / return\_id / withdrawal\_id / reinvest\_id
* **note**
* **order\_id(s)** (optional: link the receipt to its source orders for per-order analysis)
* The **withdrawable balance** is computed in real time from this ledger.
* **B) Withdrawal Record**
* **withdrawal\_id**, **partner (name)**, **amount**, **currency**, **historical FX** (if ETB), **date**, **note**.
* Creates a **Withdrawal** entry in the Revenue Ledger.
* **C) Reinvestment**
* **reinvest\_id**
* **amount (USD)** to transfer into Working Capital
* **transfer/sending cost** (bank/wire/spread)
* **actual fee currency** (USD/ETB) + **historical FX**
* **date**, **note**, **counterparty** (bank/internal entry)
* **Allocation policy across orders (optional):**
  + **Aggregate** (deduct from total balance without per-order ratios)
  + **Pro-rata** across a set of orders by their share of collections
  + **Specified** (choose exact orders)
* **Generates automatically:**
* In **Revenue Ledger**: **ReinvestOut** for the amount, and **TransferFee** for the fees.
* In **Working Capital (1)**: **CapitalIn** for the amount (increases capital).
* In **Operating Expenses (5)**: a **transfer fee expense**.
* **D) Currency & FX Framework**
* Base **USD**; supporting **ETB**.
* Every entry is saved with its **original currency** and auto-converted to USD using the **active central rate at entry time** (stored historically).
* No retroactive FX edits after save; any later differences are posted as a separate settlement.
* **3) Workflow (Step-by-Step)**
* **A) Revenue Accumulation**
* When a **sales invoice (6)** is issued, it **does not** increase the withdrawable revenue balance; it’s accounting revenue only.
* When recording a **customer receipt**:
  + System creates a **CustomerReceipt** in the Revenue Ledger (with payment currency + historical FX).
  + **Withdrawable balance increases**.
* When issuing a **customer refund** (return/adjustment):
  + **CustomerRefund** (negative), reducing the balance.
* **The screen always shows:**
* **Accounting revenue** for the period/total (sales net of returns) — for reference.
* **Withdrawable balance** (cash-like) — used for withdrawals/transfers.
* **B) Partner Withdrawals**
* Authorized user selects **+ Withdrawal**.
* Enter: partner, amount, currency, date, note.
* **Validation:** cannot exceed the withdrawable balance.
* Save → a **Withdrawal** ledger entry is created and shown in order/period summaries as a separate “Withdrawals” line.
* **Accounting note:** withdrawals are **not operating expenses** and **do not reduce order profit**; they are distributions of accumulated cash. They still appear in management and liquidity summaries.
* **C) Reinvest Revenues into Working Capital**
* **Admin only** opens **+ Reinvestment**.
* Enter:
  + Amount to transfer to capital (USD).
  + Transfer/sending cost (bank/wire/FX spread).
  + Fee currency/FX if ETB.
  + Note and (optional) order allocation policy.
* **Validation:** amount + fees ≤ withdrawable balance.
* **On save, the system executes automatically:**
* **Revenue Ledger:**
  + **ReinvestOut = − amount**
  + **TransferFee = − fees**
* **Working Capital (1):**
  + **CapitalIn = + amount** (operational balance increases immediately)
* **Operating Expenses (5):**
  + Records **transfer fees** as an operating (bank/transfer) expense, linked to the period (and to orders if desired).
* Now new purchases/shipping (2/4) can be funded from Working Capital as usual.
* **4) Controls & Validations**
* No **withdrawal/transfer** may exceed the **withdrawable balance**.
* **Central FX rate (from 10)** required for any ETB entry; stored historically and **not editable** retroactively.
* **Period lock:** after closing, no new entries within that period unless manager-approved with **Audit Log**.
* No deletion of a revenue entry that has been used in subsequent actions (withdrawal/transfer) — use **reverse/correct** only.
* **Duplication prevention:** cannot post the same receipt twice for the same bank transfer (check reference number/Hash).
* **Large emergency withdrawal:** requires second approval (sensitive policy from 8).
* **Customer refund** cannot exceed that customer’s collections or A/R — must reconcile precisely with invoices/returns (6).
* **5) Accounting Integration**
* **With (6) Sales**
* **CustomerReceipt** generated from customer payments.
* **CustomerRefund** generated from customer returns or refunds.
* **Aging** and **DSO** reports are anchored to these entries.
* **With (1) Working Capital**
* **ReinvestOut** in Revenues ↔ **CapitalIn** in Working Capital (one matching reference).
* **Withdrawals never** pass through Working Capital.
* **With (5) Operating Expenses**
* **Transfer fees** are recorded as operating expenses (bank/wire) and shown separately from purchases and shipping.
* **With (9) Reports**
* Withdrawals and reinvestment lines appear separately in order/period summaries.
* Order **P&L** remains **unaffected by withdrawals**; withdrawals are liquidity/distribution items, not costs.
* **With (10) Settings**
* Currency and central FX.
* Approval policies (large withdrawal/transfer/reversal).
* **6) Roles & UI**
* **Admin/Owner:** see everything; execute reinvestments; set allocation/approval policies.
* **Finance:** record collections and refunds; execute withdrawals after approval; manage settlements.
* **Sales:** record customer collections and link to invoices; no access to withdrawals/reinvestment (unless Admin enables limited view).
* **Warehouse/Purchasing/Worker:** no access.
* **Proposed UI:**
* Top of page: large cards for
  + **Accounting Revenue (period/total)**
  + **Withdrawable Revenue Balance**
  + **Withdrawals this month**
  + **Reinvestment this month**
* Tabs: **[Collections] [Withdrawals] [Reinvestment] [Refunds] [Summary]**.
* **+ Withdrawal**, **+ Reinvestment** buttons with step-by-step wizards and balance validation.
* **7) Reports & KPIs**
* **Reports**
* **Revenue Ledger statement:** all entries with types and references.
* **Revenue summary:** Collections − (Refunds + Withdrawals + Transfers + Fees) = **Withdrawable balance**.
* **By orders:** source of collections per order (if linked), and each order’s contribution to the balance.
* **Use of revenues:** Withdrawals vs Reinvestment (absolute and %).
* **Collections vs Sales:** period sales vs collections.
* **KPIs**
* **Collection rate** = (period collections ÷ period sales).
* **DSO (Days Sales Outstanding)** = average days to convert sales into cash.
* **Reinvestment ratio** = (ReinvestOut ÷ period collections).
* **Revenue-to-Capital ratio** = what % of capital originates from reinvested revenues.
* **Transfer/collection cost ratio** = (fees ÷ transfers).
* **8) Notifications**
* Low **withdrawable balance** below a threshold (configurable).
* Withdrawal request exceeding balance (block + alert).
* Large withdrawal requiring manager approval (second sign-off).
* Old collections not allocated to invoices (unapplied cash).
* High transfer fees vs usual (±X%).
* FX rate variance across close days — review alert (documentation only).
* **9) Edge Cases & Fixes**
* **Receipt misapplied to wrong customer/invoice:** Post a reversal, then re-assign correctly, with Audit Log.
* **Excess refund to customer:** Block, or record as a **customer credit** to offset future invoices.
* **Reinvestment from revenues to capital in ETB:** Amount is converted to USD at current central FX; any extra spread/difference recorded under **TransferFee**.
* **Cancel a reinvestment after execution:** Double reversal:
  + In **Revenue Ledger:** +ReinvestOut, +TransferFee (reversing the negatives).
  + In **Working Capital:** **CapitalOut** equal to the prior **CapitalIn**.
  + In **Expenses:** reverse the fee (if posted). Requires manager approval.
* **Allocating reinvestment deduction across orders:** If enabled, create reference-only allocation records for analysis — **no effect** on order P&L.
* **10) Operational Checklist**
* Record **all collections** and link them to invoices/customers (currency and historical FX saved).
* Ensure the **withdrawable balance** displays correctly *(Collections − Refunds − Withdrawals − Transfers − Fees)*.
* **Withdrawals:** document partner/amount/date and never exceed balance.
* **Reinvestment:** enter amount + transfer fee and post the triple impact: (Revenue Ledger, Working Capital, Expenses).
* Prevent direct deletion of any used entry — use **reverse/correct**.
* Review reports: collections vs sales, DSO, reinvestment ratio.
* Configure alerts (low balance, high fees, large withdrawal).
* **Close the period** after review (reopening requires permission and justification).
* **11) Realistic Numeric Examples**
* **Example (1): Accumulation and Transfer**
* Collections this month: **120,000 USD** (net after **3,000 USD** refunds).
* **Available balance before use = 117,000 USD**.
* You decide to reinvest **100,000 USD**, with **500 USD** transfer fees: **Revenue Ledger:**
  + ReinvestOut = **−100,000**
  + TransferFee = **−500**
  + New balance = **117,000 − 100,000 − 500 = 16,500 USD**
* **Working Capital (1):**
  + CapitalIn = **+100,000** (capital increased)
* **Operating Expenses (5):**
  + Transfer fee = **500** (operating expense)
* **Example (2): Partner Withdrawal**
* Withdrawable balance now = **16,500 USD**.
* Partner withdrew **5,000 USD**:
  + **Withdrawal = −5,000**
  + New balance = **11,500 USD**.
  + “Withdrawals” shows as a separate line in period/order summaries.
* **Example (3): Collection in ETB**
* You received **3,000,000 ETB**; central FX today **= 60 ETB/USD** → **50,000 USD**.
* Revenue Ledger: **CustomerReceipt = +50,000** (with ETB=3,000,000, Rate=60 stored).
* It enters the **withdrawable balance** immediately.
* **Example (4): Customer Refund (Return)**
* A customer returned goods worth **2,400 USD**:
  + **CustomerRefund = −2,400**
  + Withdrawable balance decreases by the same amount.
  + Stock returns to the correct warehouse per its type (see stage 6).
* **Summary**
* Stage **(7)** cleanly separates **accounting revenue** from the **cash-like withdrawable balance**. **Withdrawals** do not touch Working Capital nor P&L; they are distributions. **Reinvestment** reduces available revenues and increases Working Capital, with transfer fees recorded as an operating expense. Everything is recorded in the original currency with historical FX, with approvals and **Audit Log** controls. Reports and KPIs show **collections, withdrawals, transfers, fees, DSO, and reinvestment ratios** — at both period and per-order levels.
* **1) Definition and Goal**
* Build a **flexible RBAC system** (Role-Based Access Control) that defines **what a user can see** and **what they can do**, with the ability to **combine multiple roles** for the same user and bind them to **specific warehouse(s)**.
* Reduce risk via **sensitive-change policies** (one-time/approval/price cap ±%/block), with a comprehensive **Audit Log**.
* **2) RBAC Model + Principles**
* **Core roles:**
  1. **Owner/Admin**
  2. **Finance**
  3. **Purchasing**
  4. **Warehouse**
  5. **Shipping**
  6. **Sales**
  7. **Worker** (execution/filtering/packing staff)
* **Role combination:** a user can hold more than one role (e.g., *Purchasing + Warehouse* as you requested).
* **Warehouse Scope:** a user can be linked to one or more warehouses; everything they see/do intersects with **only their assigned warehouses** (your request applied).
* **Permission merge rule:** union permissions across roles **then** intersect with the **warehouse scope**.
  1. On conflict (allow/deny): **deny wins**.
* **Sensitive financial visibility** (optional — enabled per user by Owner/Admin): show/hide **costs, margins, working capital**. (This was “proposal #4” and made **optional** for Owner/Admin as requested.)
* **User notifications** (In-app/Email) configurable. (This was “#5” and remains enabled.)
* **Users see only their tasks and related reports** (role-guided navigation).
* **3) User Profile**
* UserID, name, status (active/frozen).
* **Roles[]:** list of assigned roles (combinable).
* **Warehouses[]:** assigned warehouses (one or more).
* **Visibility Options (optional):**
  1. Show/Hide: supplier prices, landed costs, profit margins, working-capital balance.
* **Sensitive Change Allowances:**
  1. **“One-time”** for a specific edit (e.g., change purchase price after save **once** per transaction — your proposal).
  2. **Price change cap ±%** (e.g., ±5%) before approval is required.
* **Notification Preferences:** alert types, channels.
* **Language/Time Zone** (system default).
* **4) Permission Matrix by Stages (R = Read, C/U/D = Create/Update/Delete, A = Approve)**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Stage** | **Owner/Admin** | **Finance** | **Purchasing** | **Warehouse** | **Shipping** | **Sales** | **Worker** |
| (1) Working Capital | R/C/U/D/A | R/C/U/D | R (balance view **optional**) | – | – | – | – |
| (2) Purchases | R/C/U/D/A | R/C/U/D | **R/C/U/D** (within policies) | R (balances only) | – | – | – |
| (3) Warehouse | R/C/U/D/A | R | R (purchases list only) | **R/C/U/D** (statuses/filter/pack) | R | R (sellable balances) | C (enter packing consumption) |
| (4) Shipping | R/C/U/D/A | R/C/U/D | R | R (reservations/deductions visible) | **R/C/U/D** | R (view) | – |
| (5) Operating Expenses | R/C/U/D/A | **R/C/U/D** | – | C (auto consumption only) | – | – | – |
| (6) Sales | R/C/U/D/A | R/C/U/D | – | – | – | **R/C/U/D** | – |
| (7) Revenues/Reinvestment | **R/C/U/D/A** | R/C/U/D (withdrawal after approval) | – | – | – | R (balance only) | – |
| (8) Users & Permissions | **R/C/U/D/A** | R | – | – | – | – | – |
| (9) Reports & Summaries | **R** | **R** | R (purchases only) | R (warehouse only) | R (shipping only) | R (sales/customers) | R (own scope) |
| (10) Settings | **R/C/U/D/A** | R | – | – | – | – | – |

**Important notes:**

* **Revenues → Capital funding** (7) is **Admin only**.
* Regular users in purchases/expenses fund **from Working Capital only** (per your request).
* Showing costs/margins to non-admin users is **optional** per user (Owner/Admin decision).
* Warehouse/Purchasing can be **restricted to a single warehouse** (scoped).
* **5) “Sensitive Change” Policies**
* **Policy types (set per action):**
* **Fully blocked.**
* **One-time** per document/transaction (a token auto-expires after execution or short duration).
* **Price cap ±%** (e.g., ±5%) without approval; above that → **requires approval**.
* **Always pre-approval** (no final save before manager sign-off).
* **Catalog of sensitive actions (tight examples):**
* **(2) Purchases:** edit *price/weight/currency* after linking to filtering/shipping/sales; change **funding source** after save.
* **(3) Warehouse:** reverse filtering; change status for an item with confirmed shipping; edit quantities after deduction.
* **(4) Shipping:** edit weights/rate for a **confirmed** leg; cancel a confirmed leg; enter inspection that ≠ expected.
* **(5) Expenses:** reverse supplies consumption after deduction; change rent allocation method post-posting.
* **(6) Sales:** change price after invoice issuance; change source warehouse/order; delete a line after collection.
* **(7) Revenues:** cancel reinvestment after posting; large withdrawal above threshold.
* **Your requested “one-time” for purchases (practical example):**
* For each purchase\_id: Purchasing lead may **change price once** within 24 hours of save (window configurable), with **before/after Log**. Any further edit → **Manager approval**.
* **6) Approvals**
* **Triggers:** automatically when exceeding caps/policies, or manually (Submit for Approval).
* **Approvers:** Owner/Admin by default; alternates can be added.
* **Flow:** request → approver notified → approve/reject with reason → apply/revert.
* **Timeout:** configurable (e.g., 24–48 hours).
* **Effect:** prevents further use of the item until decided (partial lock).
* **7) Notifications *(enabled — “#5”)***
* **Per-user configurable** (types + channel). Examples:
* **Purchasing:** supplier price deviation, unsettled advances.
* **Warehouse:** items awaiting filter/ready to ship for X days, supplies stockout.
* **Shipping:** high Chargeable variance, shipment arrived not inspected.
* **Finance:** consumption over balance, entry after close, high transfer fees.
* **Sales:** A/R 30/60/90 days, customer credit limit.
* **Admin:** all approval requests, sensitive changes, large withdrawal.
* Channels: **In-app / Email**. Daily/weekly summaries available.
* **8) What Each Role Sees (Guided UI)**
* **Purchasing:** suppliers screen + purchase entry + purchase-only reports (no full costs if hidden).
* **Warehouse:** first/final warehouse screens, statuses, filtering, packing; supplies balance board (read-only).
* **Shipping:** shipping legs, arrival costs, inspection, shipping cost summary.
* **Finance:** expenses, collections, withdrawals, settlements, financial reports, ledgers.
* **Sales:** customers, invoices, receipts, returns, sales/Aging reports.
* **Worker:** execution-only forms (filter/pack/readings), no sensitive figures.
* **Owner/Admin:** everything + settings + user management + KPI dashboards.
* **Per your request:** users see **only assigned tasks** and their reports; other menus are hidden.
* **9) Audit Log**
* For every important change: **who? when? what? before/after? reason?**
* Covers: create/update/delete/reverse/approve, and “one-time” policies.
* Reports: “Most editing users,” “Approved/Rejected approval requests,” “Post-close edits.”
* **10) User-Level Reports & KPIs**
* **User activity:** number of entries, error/reversal rate, task completion time.
* **Warehouse/Worker performance:** number of filtering/packing ops, productivity/hour.
* **Purchasing KPIs:** adherence to price caps, advance settlement, avg invoice close time.
* **Sales KPIs:** collection rate per user, customer DSO under their management.
* **Compliance:** adherence to sensitive/approval policies.
* **11) Real Scenarios (Examples)**
* **A) User “Purchasing + Warehouse” scoped to Addis warehouse only**
* Sees purchases + **only** the first-warehouse (Addis) screen.
* Can set “Needs Filter/Ready to Ship,” execute filtering and packing.
* May change purchase price **once** per transaction within 24 hours; afterward **Admin approval** required.
* **B) Sales without cost visibility**
* Sees: price/carton, sellable stock, Aging.
* **Does not** see landed cost/margin unless Owner enables visibility.
* **C) Finance reviews negative supplies consumption**
* Warehouse tried to deduct supplies beyond balance → blocked + approval request → Finance approves after adding a supplies purchase.
* **D) Shipping changes Chargeable after confirmation**
* Sensitive action → creates approval request → Admin reviews and approves with reason → full Log captured.
* **12) Checklists**
* **Onboarding (add user):**
* Assign appropriate **roles** (can be combined).
* Set allowed **warehouses**.
* Configure **visibility options** (show/hide prices/costs).
* Define **sensitive-change policies** (one-time/±%/approval).
* Enable appropriate **notifications**.
* Test that the user only sees what they need.
* **Offboarding (exit):**
* Disable account (Freeze).
* Reassign pending tasks.
* Archive their permissions and signatures in Audit.
* **13) Edge Cases & Fixes**
* **Role conflicts:** deny wins; if *Sales* grants price visibility but user-level setting hides it → **hiding applies**.
* **Empty warehouse scope:** user sees nothing until at least one warehouse is assigned.
* **Reopen a closed period:** Admin only with reason; logged in Audit.
* **Changing permissions historically:** does not alter past records, but limits what they see/do going forward.
* **Summary**
* Implemented a **precise and flexible permission system**: role combination, warehouse scoping, optional sensitive-data visibility per user (Owner/Admin control), sensitive-change policies (one-time/±%/approval), configurable approvals & notifications, role-guided UIs, and a full Audit Log.
* **Your special constraints are enforced:**
  + **Revenues → Capital funding** is **Admin only**.
  + Purchasing/Expenses users fund **from Working Capital** only.
  + Ability to **combine Purchasing + Warehouse** for one user.
  + **Warehouse scoping** for Purchasing/Warehouse (and others as needed).
  + **Show/Hide costs** is **optional** per user.
  + **Notifications** active for every role.
* **User Profile Template**
* **1) Basic Information**
* **User ID** (auto- or manual generation)
* **Full Name**
* **Username**
* **Password** (secure management / reset by Admin)
* **Status**: ✅ Active / ❌ Frozen
* **2) Roles**
* (Selected from a multi-select list — multiple roles can be combined)
* ☐ Owner/Admin
* ☐ Finance
* ☐ Purchasing
* ☐ Warehouse
* ☐ Shipping
* ☐ Sales
* ☐ Worker
* **3) Warehouse Scope**
* (Select one or more)
* ☐ Addis Ababa Warehouse
* ☐ Riyadh Warehouse
* ☐ Destination (Final) Warehouse
* ☐ … (add as defined in Settings 10)
* **4) Visibility Options**
* ☐ Show supplier prices
* ☐ Show landed cost
* ☐ Show profit margin %
* ☐ Show working-capital balance
* These are tied to **Owner/Admin** decision: if not enabled for the user → the data stays hidden.
* **5) Sensitive Change Policy**
* For each action type, Admin sets:
* **Purchases (buy/price/currency):** ☐ Block ☐ One-time ☐ ±% (set %) ☐ Manager approval
* **Warehouse (quantities/filtering/packing):** ☐ Block ☐ One-time ☐ ±% (set %) ☐ Manager approval
* **Shipping (weights/Chargeable):** ☐ Block ☐ One-time ☐ ±% (set %) ☐ Manager approval
* **Sales (price/order/invoice line):** ☐ Block ☐ One-time ☐ ±% (set %) ☐ Manager approval
* **Revenues/Withdrawals/Reinvestment:**
  + **Withdrawal:** ☐ Allowed ☐ Allowed after approval ☐ Forbidden
  + **Reinvestment:** ☐ Admin only (default and unchangeable)
* **6) Notifications**
* (Selected by the user or Admin: types and channel)
* **Types:**
  + ☐ Purchasing (price changes, unsettled advances)
  + ☐ Warehouse (awaiting filtering, supplies stockout)
  + ☐ Shipping (high Chargeable, arrival not inspected)
  + ☐ Sales (A/R 30/60/90, customer limit exceeded)
  + ☐ Finance (high fees, entry after period close)
  + ☐ Administrative (approval requests, large withdrawal)
* **Channel:**
  + ☐ In-app
  + ☐ Email
  + ☐ Both
* **Frequency:**
  + ☐ Instant
  + ☐ Daily summary
  + ☐ Weekly summary
* **7) Other**
* **Language:** Arabic / English
* **Time Zone**
* **Last Login Date** (for review only)
* 📝 **User Profile Sample**
* **User ID:** U-102
* **Name:** Ahmed Ali
* **Username:** ahmed.a
* **Status:** Active
* **Roles:** Purchasing ✅ + Warehouse ✅ **Warehouse Scope:** Addis Ababa ✅
* **Visibility Options:**
* Supplier prices: ✅
* Landed cost: ❌
* Profit margin: ❌
* Working-capital balance: ❌
* **Sensitive Change Policies:**
* Purchases: price edit **one-time** allowed
* Warehouse: quantity edit requires **Manager approval**
* Shipping: **fully blocked**
* Sales: **not authorized**
* Revenues: Withdrawal ❌, Reinvestment ❌
* **Notifications:**
* Purchasing ✅
* Warehouse ✅
* Channel: In-app only **Time Zone:** Africa/Addis\_Ababa
* With this template, every user has a **clear profile**: roles, warehouse scope, permissions, their specific sensitive-change policies, notifications, and what is shown or hidden for them.
* **(9): Reports & Summaries**
* **1) Definition & Objective**
* This stage is the **analysis & management interface** of the entire system.
* **Purpose:** Convert data (purchases, warehouses, shipping, sales, expenses, revenues, withdrawals, etc.) into **accurate financial & operational reports** and **strategic summaries**.
* Helps answer:
* How much did each order cost me?
* How much did I profit or lose?
* Which supplier is best?
* Which customer is most loyal and profitable?
* Where is the damaged stock? What is its percentage?
* How does liquidity flow (from capital to revenues and back)?
* **2) Levels of Reports**
* **A) Stage Reports**
* Each stage has its own report (from Stage 1 to 8):
* (1) Capital: opening balance + increase + usage + reinvestment.
* (2) Purchases: quantities in kg, supplier prices, payments (cash/credit/advance), historical FX rate.
* (3) Warehouse: balances filtered/unfiltered/pending, filtering ratio per supplier.
* (4) Shipping: leg costs, net/gross/chargeable weight, avg shipping cost/kg.
* (5) Expenses: labor, rent, supplies (cartons/tape), bank fees, etc.
* (6) Sales: sales by customer, by order, by unit (8 or 20 kg), receivables.
* (7) Revenues & Withdrawals: receipts, refunds, available balance, withdrawals, reinvestments.
* (8) Users: user activity, approval requests, sensitive changes.
* **B) Order Summary Reports**
* Purchase cost
* Shipping cost (legs + arrival costs)
* Operating expenses
* Damaged stock
* Net saleable stock
* Actual sales (per order)
* Total generated revenue
* Withdrawals for the period
* Final result: Profit / Loss
* **C) Period Summaries**
* Monthly / Quarterly / Annual (per Settings 10)
* Aggregate multiple orders
* Compare periods (e.g., Q1 vs Q2 profit)
* **D) Global Summary**
* Covers the entire system (all orders + all periods).
* Measures: capital, revenues, withdrawals, reinvestments, profit margins, damaged ratios.
* **3) Report Content in Detail**
* **A) Purchase Reports**
* By supplier: total qty, avg price/kg, filtering need ratio.
* Supplier comparison: lowest price, lowest filtering %.
* Payables aging (paid / remaining / due).
* **B) Warehouse Reports**
* Stock by status: filtered / unfiltered / pending.
* Warehouse stock by country (Addis, Riyadh, etc.).
* Filtering % by supplier (clean vs unclean).
* Value of stock ready for shipping.
* Value of unclean stock (zero cost but shows as extra profit when sold).
* **C) Shipping Reports**
* Leg details: net/gross/chargeable weight, rate/kg, total cost.
* Arrival costs grouped (Broker, Cargo, Delivery).
* Avg transit time.
* Compare shipping cost/kg across orders.
* Inspection report: clean qty + damaged qty.
* **D) Operating Expense Reports**
* Total expenses by period.
* Breakdown by type (rent, labor, supplies, bank fees).
* Monthly comparisons.
* Supplies tracking (cartons/tape/etc.): purchase − usage = remaining.
* **E) Sales Reports**
* Total sales (USD).
* By unit: 8 kg vs 20 kg.
* By customer:
  + Total purchase
  + Top buyer
  + Largest debtor
  + Most profitable customer
* By order: how much sold vs remaining.
* Sales of unclean goods as extra profit.
* **F) Revenue & Withdrawal Reports**
* Revenue ledger: receipts, refunds, withdrawals, reinvestments, fees.
* Available revenue balance.
* Reinvestment % of total revenue.
* Withdrawals by partner.
* Comparison: withdrawals vs reinvestment.
* **G) User Reports**
* Activity per user (number of actions).
* Who edited most / needed approvals.
* Policy compliance: edits within ±%, approvals requested.
* **4) KPIs**
* **Operational**
* Avg cost per kg (purchase + shipping + prep ÷ clean qty).
* Avg selling price/kg.
* Damage % rate.
* Profit margin %.
* Collection rate (receipts ÷ sales).
* DSO (Days Sales Outstanding).
* % of sales from unclean goods.
* **Financial**
* Withdrawals ÷ revenues %.
* Reinvestment %.
* Current capital balance.
* Available revenue balance.
* **5) Report Alerts**
* Loss-making orders (red highlight).
* Customer overdue >90 days.
* Operating expenses above historical average.
* Transfer fees higher than usual.
* Damage ratio above threshold.
* **6) Report Access Rights**
* **Admin:** all reports.
* **Finance:** financial, revenues, expenses.
* **Purchasing:** supplier & purchase reports.
* **Warehouse:** stock balances & filtering.
* **Shipping:** shipping & inspection reports.
* **Sales:** customer & sales reports.
* **Worker:** only task reports (filtering, packing).
* **7) Integration with Other Stages**
* Relies on all: (1) Capital, (2) Purchases, (3) Warehouse, (4) Shipping, (5) Expenses, (6) Sales, (7) Revenues/Withdrawals, (8) Users.
* Each stage feeds its numbers into reports.
* Accounting link: every report value is traceable to its origin (invoice, purchase, shipping, etc.).
* **8) Display Interface**
* **Dashboard:** quick KPIs (cost/kg, selling price, margin, damage %, revenue balance).
* **Report list:** by stage (purchases, warehouse, shipping, etc.).
* **Order summary:** integrated report per order.
* **Period summary:** integrated report per month/quarter/year.
* **Export:** PDF / Excel.
* **9) Special Cases & Corrections**
* FX variance: shows in expenses/revenues as separate line.
* Mixed order: sales report shows distribution from each order.
* Excess withdrawal: revenue balance negative → instant alert.
* Reversal: any report shows reversed/corrected entries (full transparency).
* **10) Practical Examples**
* **Example (1): Order Summary**
* Purchase: $7,320
* Shipping: $1,435
* Expenses: $300
* Damaged: $73
* Sales: $3,625
* Net: −$5,503 (loss)
* **Example (2): Customer Report**
* Customer: Mohammed
* Total purchase: $12,000
* Paid: $7,000
* Remaining: $5,000
* Delay: 65 days → shown in Aging.
* **Example (3): KPI**
* Avg cost/kg: $14.72
* Avg selling price: $18.2
* Margin: 19%
* Damage: 1.5%
* Reinvestment ratio: 80%
* **Conclusion**
* Stage (9) provides a **comprehensive picture** of the system through:
* Stage reports,
* Order summaries,
* Period summaries,
* Global summary.
* Supports **accounting traceability, financial & operational analysis, proactive alerts, and strategic decisions**.
* All reports can be **filtered & exported**, with **access rights** defining who sees what.
* **1) Purpose & Access Scope**
* Settings are the **central brain** that governs the behavior of the entire system (warehouses, sales, filtering, reports, security, numbering, etc.).
* **Access:** Owner/Admin only (Finance can see some **read-only** screens like historical FX, with no edits).
* **2) Settings Sections & Fields (Forms)**
* **A) General System Settings (System)**
* **Fields:**
* **Base currency:** fixed = **USD**.
* **Secondary currency:** fixed = **ETB**.
* **Default time zone**.
* **Default UI language:** Arabic / English.
* **System start date** (for archiving and default reports).
* **Enable/disable “non-clean” sales** (affects interfaces in (6) Sales and (3) Warehouse).
* **Policies:**
* Changing language/time zone does not affect data, only presentation.
* Disabling “non-clean” sales immediately hides local-sale options and **blocks any invoice from FIRST**.
* **B) Currency & FX Settings**
* **Fields:**
* **Current central FX rate:** **ETB per 1 USD** (decimal).
* **Activation date** (default: now).
* **Note/Reference** (optional).
* **Historical FX log** (table: date, rate, who modified).
* **Strict rules (per your request):**
* **No FX entry inside any transaction** (purchase/sale/expense/shipping/collection). Conversion to USD is **automatic** using the **central rate active at save time**, and it is **written** into the transaction line as fx\_rate\_used and never changes later.
* **Changing the rate here does not reprice past transactions**. Any later variance is handled as separate settlements if needed.
* **Alerts:**
* On rate change: confirm dialog + Admin notification + **Audit** entry.
* If changed by more than X% from the last rate (configurable): warning “Unusual rate jump”.
* **C) Warehouse Settings (Warehouses)**
* **Fields (per warehouse):**
* **Warehouse Code** (unique identifier).
* **Name** (e.g., Addis Ababa Warehouse).
* **Type:**
  + **FIRST** (post-purchase/filtering/packing).
  + **FINAL** (destination-country warehouse after shipping & inspection).
* **Country/City** (for reporting).
* **Status:** active / disabled.
* **Default?** (default warehouse for operations that need a default).
* **User binding:**
* Link specific users to this warehouse (integrates with (8) Users). A user only sees their assigned warehouses.
* **Rules:**
* You cannot **delete** an active warehouse with balances; you may **disable** it after zeroing balances.
* A warehouse cannot be both types at once.
* **D) Sales Settings**
* **Fields:**
* **Default unit:** carton **8 kg** or **20 kg**. *(As requested: these two only.)*
* **Allow non-clean sales?** (mirror of the global switch; shown here for clarity—kept in sync).
* **Rules:**
* Changing the default unit affects **new invoices only**.
* If non-clean sales are disabled: selecting **FIRST** as source in invoices is blocked.
* **E) Filtering Settings**
* **Fields:**
* **Critical damage threshold for alert** (e.g., 2%): exceeding it at inspection (4) triggers an alert.
* **Allow selling non-clean locally:** on/off (synchronized with System/Sales).
* **Rules:**
* **Non-clean cost = zero** (fixed and **not** editable from settings — system policy).
* **F) Reports & KPIs Settings**
* **Fields:**
* **Default reporting currency:**
  + **USD only**, or
  + **USD + ETB** (dual display).
* **Default period:** monthly / quarterly / annual.
* **Dashboard KPI toggles** (on/off per indicator):
  + Avg cost per kg
  + Avg selling price/kg
  + Profit margin %
  + Damage %
  + Collection rate
  + (Optional) % of non-clean sales
* **Rules:**
* Changing the default currency does not change balances, only presentation.
* **G) Security Settings**
* **Fields:**
* **Session timeout**.
* **Account lock** after N failed attempts.
* **Sensitive-change policies** (enable approvals framework — integrates with (8)):
  + “One-time” for specific edits.
  + ±% cap before requiring approval.
  + Actions that **always require approval** (define here; e.g., editing a confirmed shipping leg).
* **Notifications:** enable/disable In-app and/or Email at system level.
* **Rules:**
* Policies set here become **defaults** for all users; can be customized per user in their profile (8).
* **H) Numbering Settings**
* **Fields (per document type):**
* **Document:** Purchases (PO), Sales (INV), Shipping (SHP), Expense (EXP)…
* **Prefix:** e.g., INV, with optional country/warehouse code: ADD-INV.
* **Pattern details:**
  + **Year** (YYYY or YY) optional
  + **Separator** (e.g., -)
  + **Sequence length** (3 → 001, 4 → 0001)
* **Preview:** shows a real example number before saving.
* **Reset:** yearly reset / continuous.
* **Rules:**
* **Uniqueness** check across type + prefix + year + sequence.
* You may not change a pattern in use to one that would collide with existing numbers.
* With **yearly reset**, numbering starts from 001 at the new year.
* **3) Business Rules & Integration**
* **Currencies:** any ETB transaction is auto-converted to USD at the central rate active at save time and stores fx\_rate\_used inside the record. **No manual FX field inside transactions** (per your request).
* **Warehouses:** choosing type (FIRST/FINAL) governs sellability, shipping, and filtering as detailed in (3)(4)(6).
* **Sales:** units restricted to 8/20 kg. The non-clean sales toggle controls the source in sales invoices (FIRST).
* **Filtering:** the critical damage threshold triggers an alert in (4) Inspection.
* **Reports/KPIs:** changes to default period/currency reflect immediately in all **new** report views (presentation only).
* **Security:** “One-time / ±% / Approval” policies apply automatically to sensitive actions across stages (2–7).
* **Numbering:** each document type has its own counter; pre-save validation prevents number conflicts.
* **4) Change Governance**
* **Draft → Publish** (recommended): edit a set of settings in **Draft**, then **Publish** them at once.
* **Audit Log:** who changed? when? old → new? description.
* **Activation scheduling** for some changes (e.g., FX): now / midnight.
* **Settings snapshots:** save the current configuration (restore when needed).
* **5) Access**
* **Owner/Admin:** can edit everything.
* **Finance (read-only):** can see “historical FX” and “numbering patterns” (no edits).
* All other roles: **no access** to settings.
* **6) Settings-Driven Notifications**
* **FX rate change:** notify Admin + add Audit entry.
* **Unusual FX jump:** warn before save.
* **Enable/disable non-clean sales:** notify Sales/Warehouse.
* **Numbering sequence nearing end** (e.g., 3 digits approaching 999): alert to increase length.
* **Disable warehouse with balances:** block + corrective message.
* **7) Edge Cases & Corrections**
* **Retroactive FX change:** **forbidden**. To correct impact, use **settlement entries** in (5)/(7).
* **Moving documents to a new prefix** after changing numbering: old remain as-is; new start with the new prefix.
* **Attempt to delete an active warehouse:** system blocks; shows a report of balances preventing deletion.
* **Disable non-clean sales** while draft invoices exist from FIRST: block issuing until source changes to FINAL or drafts are canceled.
* **Numbering conflict:** save is blocked; show conflicting examples for review.
* **8) Checklists**
* **Initial system launch:**
* Set time zone and language.
* Add warehouses and define each warehouse type.
* Assign users to warehouses (Stage 8).
* Choose default sales unit (8 or 20 kg).
* Enter the **current central FX rate**.
* Enable/disable non-clean sales (per your policy).
* Enable desired Dashboard KPIs.
* Define numbering pattern per document type + preview examples.
* Enable security policies (timeout/lock/approvals).
* Take a **Snapshot** and store it.
* **For every important change:**
* Review impact (who/which screens are affected?).
* Write a clear change note.
* Publish change + review Audit.
* Inform relevant teams (Sales/Warehouse/Finance) if needed.
* **9) Short Operational Examples**
* **Example 1 — Raise FX from 60 to 65 ETB/USD:**
* Admin → Settings > Currency.
* Edit value to 65, add note “Bank update…”, save.
* All ETB transactions **after this moment** convert at 65 and store fx\_rate\_used = 65.
* Earlier transactions keep their historical rate (no repricing).
* **Example 2 — Add a new final warehouse (Jeddah):**
* Create Warehouse:
  + Code: **JED-FINAL**
  + Type: **FINAL**
  + City: **Jeddah**
  + Status: **Active**
* Link Shipping (and Sales if needed).
* Appears in (4) as a possible shipping destination, and in (6) as a source for clean sales after inspection.
* **Example 3 — Change default sales unit to 20 kg:**
* Settings > Sales: choose **20 kg** as default.
* New invoices will default to 20 kg (users can still select 8 manually).
* Stock and old invoices unaffected.
* **Example 4 — Toggle non-clean sales:**
* If you disable:
  + Sales screen won’t allow choosing FIRST.
  + “Non-clean” tab remains in Warehouse for viewing only, with **no “Sell”** button.
* Re-enabling restores the button.
* **Example 5 — Configure sales numbering:**
* Document: **Invoice**
* Prefix: INV
* Year: YYYY
* Separator: -
* Length: 4
* Reset: **Yearly**
* Preview: INV-2025-0001
* Save → starts from **0001** for invoices in 2025.
* **10) Full Integration with Stages (1 → 9)**
* **(1) Working Capital:** central FX governs ETB↔USD conversions; “Reinvestment” posts fees as **Expense (5)**.
* **(2) Purchases:** no FX inside transactions; auto-convert at central rate with historical store; purchase units fixed in kg.
* **(3) Warehouse:** warehouse type (FIRST/FINAL) defined here controls what is sold, where filtering happens, and where stock enters after inspection.
* **(4) Shipping:** leg/arrival cost currencies converted centrally; inspection uses damage alert threshold from Filtering settings.
* **(5) Expenses:** expense items (bank/transfer fees) are recorded and shown in reports; consumption tied to packing.
* **(6) Sales:** units 8/20 kg; allowing non-clean sales controlled here; source warehouse must comply.
* **(7) Revenues/Withdrawals:** transfer to capital is **Admin only**; transfer fees recorded as expense and deducted from available revenue.
* **(8) Users:** warehouse binding, sensitive-change policies, notifications; cost/margin visibility **optional** per user by Owner decision.
* **(9) Reports:** display currency, default period, and KPIs come from here; any change reflects immediately in new views.
* **Stage (10) Summary**
* **Centralized, disciplined** control for everything governing operations and finance.
* A single **central FX rate** ensures consistency, with **historical locking** in every ETB transaction (no manual entry).
* Flexibility in **warehouses, sales, filtering, reports, security, numbering** with change governance (Draft/Publish, Audit, alerts, snapshots).
* Full integration with the other stages, so any change here reflects **immediately and safely** across daily operations.