

Functional Requirements Document (FRD)

1. Overview

The system is a multi-tenant, phase-based product tracking and processing platform.

It allows an admin to design workflows (phases), create phase head/user profiles, dynamically define product forms, and manage items as they move forward or backward in the workflow. Bulk operations, returns, real-time dashboards, and historical tracking are included.

2. Key Features

1. Admin Management

- Create/update/delete phases in a workflow
- Assign multiple users to a phase
- Manage user credentials & roles
- Create dynamic item form (predefined field types: small text, large text, image, dropdown)
- View global dashboard of all phases

2. Item Lifecycle

- Create item at “Kora” (start point)
- Assign global tracking ID (6-digit, zero-padded, human-readable, e.g., 000123)
- Add/remove items in single/bulk
- Move items forward (sequential)
- Return items backward (non-sequential)
- Maintain full history (timestamps, phase name, user, action)

3. Bulk Operations

- CSV/manual ID input for bulk operations
- Partial successes allowed
- Success/failure report in CSV/JSON format
- Return requests single/bulk
- Accept/reject return requests

4. Returns

- Raise return request from current phase
- Approval required by receiving phase
- Log return request with status (Pending, Accepted, Rejected)
- Non-sequential backward move

5. Real-Time Dashboard

- WebSocket: For bi-directional events (e.g., item arrival, return approvals)
- SSE: For one-way dashboard status updates
- Status of all items and phases visible in real time

6. File Handling

- Max 10 images per item, 5 MB each
- Store on Cloud (AWS S3 or VPS)

- Optional image compression/resizing
- Secure pre-signed URL uploads

7. Multi-Tenant

- Each organization has its own phases, users, and items
- Shared infrastructure with tenant-based isolation in DB

3. Entities & Data Model

Core Entities

1. Tenant

- _id
- name
- createdAt

2. User

- _id
- tenantId
- username
- passwordHash
- role (ADMIN, PHASE_HEAD, OPERATOR)
- phaseId (nullable for Admin)
- createdAt

3. Phase

- _id
- tenantId
- name
- sequenceOrder
- users[] (linked to User IDs)
- createdAt

4. ItemFormTemplate

- _id
- tenantId
- fields[]:
 - name
 - type (small_text, large_text, image, dropdown)
 - options[] (for dropdown)
 - required
- createdAt

5. Item

- _id
- tenantId

- trackingId (6-digit padded, unique globally)
- formData (key-value pairs)
- currentPhaseId
- history[]:
 - phaseId
 - userId
 - action (CREATE, MOVE_FORWARD, RETURN, ADD_BULK, REMOVE_BULK)
 - timestamp
- images[] (URLs)
- status (IN_PROGRESS, RETURNED, COMPLETED)

6. ReturnRequest

- _id
- tenantId
- itemIds[]
- fromPhaseId
- toPhaseId
- status (PENDING, ACCEPTED, REJECTED)
- requestedBy
- approvedBy (nullable)
- createdAt

4. API Endpoints

Auth

- POST /auth/login — login, JWT token
- POST /auth/register — admin creates phase head/operator

Admin

- POST /admin/phase — create phase
- PUT /admin/phase/:id — update phase
- DELETE /admin/phase/:id — delete phase
- POST /admin/form-template — create/update form template
- GET /admin/dashboard — full system status

Item Management

- POST /items — create item at Kora
- POST /items/bulk — bulk create
- PUT /items/:id/move-forward — move to next phase
- PUT /items/bulk/move-forward — bulk move
- PUT /items/:id/return — single return request
- PUT /items/bulk/return — bulk return request
- GET /items/:id/history — full history

Returns

- POST /returns — raise return request
- PUT /returns/:id/accept — accept return
- PUT /returns/:id/reject — reject return

File Handling

- POST /files/presign — get pre-signed URL
- POST /files/upload — upload image

5. Real-Time Updates

- **WebSocket channels:**
 - /phase/{phaseId} — subscribe to phase events
 - /dashboard — subscribe to global events
- **Events:**
 - ITEM_MOVED
 - ITEM_RETURN_REQUEST
 - ITEM_RETURN_ACCEPTED
 - BULK_OPERATION_RESULT

6. Bulk Processing Flow

- Upload CSV or paste IDs → Backend validates IDs
- Perform operations in batches (avoid blocking)
- Log partial successes
- Generate report (CSV/JSON) → downloadable link

7. Security

- JWT authentication
- Role-based access control middleware
- Tenant-based data filtering
- Pre-signed S3 URLs for secure uploads

8. Operational Considerations

- **Scaling:** Horizontal scaling for WebSocket servers
- **Backups:** Daily DB backups, S3 lifecycle rules
- **Monitoring:** API metrics, error logs
- **Data Retention:** Configurable per tenant