Flag Panel – Detailed Technical Documentation

# Objective

Provide a centralized interface for supervisors to manage audit discrepancy flags raised during warehouse audits. This includes reviewing, resolving, or rejecting discrepancy types like LOST, EXCESS, DAMAGED, BATCH DISCREPANCY, and SKU MISMATCH.

# Core Flows

* 1. Flag Filtering:
* - User can filter the flag list by flag type, status, crate ID or WT ID, and date.
* - Positive: Correct filter returns relevant flag entries.
* - Negative: Invalid combinations (e.g., no matching data) result in an empty list.
* 2. Resolve LOST Flag:
* - Modal opens with flag details (SKU, quantity, box ID).
* - On approval, it triggers 'Lost GON' creation via `createGon` & `submitGonItcV2`.
* - Positive: Valid submission resolves flag and updates status.
* - Negative: Backend failure or user cancels submission keeps status unchanged.
* 3. Resolve EXCESS Flag:
* - Modal allows checking for recovery via `/audit/recovery-gon`.
* - If found, it initiates recovery GON + fresh inward (if partial).
* - If not, only inward flow is triggered.
* - Positive: GONs processed or recovery initiated based on scenario.
* - Negative: Recovery fails, user must manually resolve by inwarding entire qty.
* 4. Resolve DAMAGED Flag:
* - User must enter total damaged quantity and split it across specific reasons.
* - Submit triggers GON with summed damage reasons.
* - Positive: Correct breakdown enables approve, resolves flag.
* - Negative: Total mismatch disables approval, or missing reasons trigger alerts.
* 5. Resolve BATCH DISCREPANCY:
* - Modal shows expected vs. found batch data.
* - User must input new batch ID, MFG, and EXP dates (in dd-mm-yyyy format).
* - Positive: Inputs submitted to create/update batch as per resolution type.
* - Negative: Missing fields trigger alert, invalid dates rejected by SAP.
* 6. Resolve SKU MISMATCH:
* - Requires user to enter actual SKU (mandatory) and optional batch ID.
* - Approval updates SKU mapping.
* - Positive: Flag resolved with updated SKU.
* - Negative: Missing SKU disables submission.
* 7. Reject Flag:
* - Rejection modal prompts for mandatory reason.
* - Positive: Reason entered, flag status changes to REJECTED.
* - Negative: Empty reason field prevents submission (alert shown).

# Error Handling (UI + API)

* • Missing Rejection Reason → 'Please provide a reason for rejection.' alert shown.
* • Batch resolution missing MFG/EXP → Alert prevents submission.
* • Inconsistent damage quantity sum → Approval disabled until corrected.
* • Missing mandatory SKU input in SKU mismatch → Alert shown.
* • Backend failure on GON creation or flag API → Alert shown, modal remains open.

# API Mappings

* • GET /audit/flags → Fetch flags for display table.
* • POST /audit/flags/resolve → Submit resolution payload depending on flag type.
* • POST /audit/flags/reject → Submit rejection reason.
* • POST /audit/recovery-gon → Check if excess can be matched with 'Lost' flag.

# Validation Matrix

* Flag Type: LOST
* - Must include boxCode and qty.
* - Box ID mandatory.
* Flag Type: EXCESS
* - qty must be > 0.
* - isRecoveryCandidate boolean determines flow (recovery vs. fresh inward).
* Flag Type: DAMAGED
* - Total damage qty must match sum of individual reasons.
* - Each reason must include valid quantity and type.
* Flag Type: BATCH\_DISCREPANCY
* - New batch ID, manufacturing date, expiry date are mandatory.
* - Dates must be in dd-mm-yyyy format (validated against SAP).
* Flag Type: SKU\_MISMATCH
* - Actual SKU mandatory.
* - Batch ID optional.
* Rejection Case
* - Reason mandatory.