

Reconciliation Chatbot - Product Specification Document

Purpose

To develop a **semi-automated chatbot** that performs reconciliation between key actors (Donors, Service Partners, Delivery Teams, and Beneficiaries) across all asset movements. The chatbot will identify mismatches, notify relevant users, generate reports, and optionally act as a standalone tool for external integrations.

Overall Workflow

Asset Lifecycle Reconciliation:

1. **Donor Upload → Partner Handoff**
2. **Partner Repair/Dispatch → Delivery Partner**
3. **Delivery Partner → Beneficiary Receipt**

Chatbot will reconcile each handoff by:

- Comparing expected vs actual data (quantity, condition, location, etc.)
 - Generating alerts or summaries
 - Allowing admin to approve/resolve issues
 - Collecting beneficiary feedback
-

Core Functionalities

1. Reconciliation Automation Mode

- Mode: **Semi-automated**
- Bot suggests reconciliation actions

- Admin approves or overrides

2. Access Control

- Internal Admin/QA only (for now)
- Future scope to include Donor, Partner, Beneficiary access

3. Chatbot Behavior

- Dual role:
 - **Dashboard** (for viewing records, logs)
 - **Action Assistant** (to trigger reconciliation, rescheduling, notify)

4. Real-time Alerts

- Bot alerts instantly when a mismatch is detected:
 - Example: "Partner A uploaded 25 devices, Partner B received 22. Discrepancy = 3. Approve report?"



Reconciliation Logic by Workflow

A. Donor Upload → Partner Pickup

- Fields to compare:
 - Quantity
 - Device ID / Serial
 - Device type/condition
 - Scheduled pickup date
- Actions:

- Flag duplicates
- Alert discrepancies
- Generate summary report

B. Partner → Delivery Partner Handoff

- Fields to compare:
 - Quantity
 - Dispatch vs Delivery status
 - Repair status vs final delivery log
 - Time/location differences
- Actions:
 - Reschedule handoff
 - Notify admin & stakeholders

C. Delivery → Beneficiary Verification

- Fields to confirm:
 - Quantity received vs sent
 - Device condition (working/not working)
 - Confirmation by beneficiary (signature/image)
 - Actions:
 - Collect feedback (via bot)
 - Trigger alert for missing or faulty delivery
-

Reporting Features

- Generate:
 - Daily Reconciliation Logs
 - Weekly ESG Impact Reports
 - Mismatch/Exception Reports
 - Formats:
 - PDF
 - Excel
 - API JSON (optional)
-

Standalone Bot Usage

- Operate independently of main platform
 - Embed as a:
 - Web widget on partner/donor sites
 - WhatsApp/Slack/MS Teams bot
 - Allow:
 - Reconciliation commands
 - Report generation
 - Feedback collection
-

Tech Team Deliverables

1. Chatbot Interface

- Web component with conversational interface
- Message-based prompts for mismatches

2. Data Matching Engine

- Logic layer to perform field-based reconciliation
- Logging all matched/unmatched entries

3. Alert System

- Admin notifications via in-app chat or email

4. Feedback Collection

- Post-delivery chat with beneficiary
- Store responses in backend linked to asset

5. Report Builder

- Generate auto-summary based on reconciliation status

6. Admin Panel (Optional Phase 2)

- To review logs, approve actions, and configure workflows

Immediate Next Steps

- Finalize chatbot data schemas
 - Build test flows for each reconciliation step
 - Review platform integration vs standalone needs
 - Set up pilot with mock donor → partner → beneficiary test cycle
-

Prepared For: Development Team

Prepared By: [Product Manager / Consultant Name]