

Technical Documentation

Document ID: DOC^TD-17882

Title: Technical Architecture and Implementation Overview

Author: Avirup Saha

Date: June 15, 2025

1. Purpose

This document describes the technical architecture, data flow, component interactions, and system behavior of the Clinical Trial Packaging and Labeling Execution application. It serves as a reference for developers, QA engineers, and system integrators.

2. Technology Stack

Layer	Technology
UI Framework	React.js (Functional Components + Hooks)
Styling	CSS with modular class management
State Mgmt	Context API + localStorage
Routing	React Router DOM
Notifications	react-toastify
Backend Data	Static JSON (/DB_records/*.json)
Deployment	Browser-based app (static hosting compatible)

3. Folder Structure (Core)

```
/src
├─ /components
│   ├─ OperatorLogin.js
│   ├─ JobAllocationProcess.js
│   ├─ RoomAllocationProcess.js
│   ├─ EquipmentRegistration.js
│   ├─ Packaging.js
│   ├─ Reconciliation.js
│   └─ OtherInfo.js
├─ /data      (served as static JSON)
├─ /context   (Roomcontext.js, Jobcontext.js)
└─ /routes    (App-level navigation)
```

4. Component-Level Description

- OperatorLogin: Authenticates user, displays active session, prevents duplicate logins.
- JobAllocationProcess: Dropdown-based job allocation. Saves printertype to localStorage.
- RoomAllocationProcess: Allocates room and stores room ID.
- EquipmentRegistration: Validates and stores equipment types per job config.
- Checklist: Ensures pre-packaging steps are acknowledged.
- PackagingProcess: Tracks scanned barcodes and visual progress.
- Reconciliation: Calculates reconciliation % and enforces deviation checks.
- OtherInfo: Summary of current session info (room, job, operator, printer).
- SopDocument / SopDocumentCredits: Downloads SOPs and credits authorship.

5. Data Source Format

All data is loaded via fetch() from /DB_records/:

- Operator.json - ID, name, scanCode, password
- Barcode.json - Serial number to label
- SU.json - Input Sus
- Supervisor.json - User with supervisor role.

- Job.json - Job ID, order quantity, printer type
 - Equipment.json - Scan code, equipment name/type
 - Room.json - Room ID, location-status
 - Checklists.json - Array of yes/no checklist items
-

6. Key Logic Areas

- Printer Type Matching: During equipment scan, checks job's printertype against scanned equipment's equipment_type.
 - Packaging Completion Check: Stops packaging UI if already completed.
 - Reconciliation Enforcement: If result < 100%, requires supervisor reason and login.
 - State Persistence: All session values persisted in localStorage.
-

7. Limitations

- No real-time API integration.
 - Session state reset only via browser storage clear or manual logout.
 - JSON files act as static DB mockups.
-

8. Future Enhancements

- API-driven CRUD operations
 - Audit log tracking
 - Role-based access control
 - Report generation (CSV/PDF)
-

9. Contributed by

- Avirup Saha 15-June-2025