

Hostel Laundry Management System – Build Plan

Step 0: Tools & Setup - Code Editor: VSCode - Version Control: Git + GitHub - Hosting / Deployment: Netlify or Vercel - Optional Backend / Realtime DB: Firebase - QR Codes (optional): QRCode.js for generating QR IDs

Step 1: Plan Your Pages & Components 1. Student Dashboard - Login page - List of submitted clothes: cloth ID, optional photo, status (Pending/Washed) - Button to submit new cloth - Notifications

1. Caretaker Dashboard
2. Login page
3. List of all students' clothes: student name, cloth ID/photo, status
4. Button / QR scanner to mark clothes as washed
5. Optional: batch mark multiple clothes
6. Optional Pages
7. Home / Landing page
8. About / Info page

Step 2: Decide Frontend Stack - HTML / CSS / JS for simplicity - React + Tailwind or Bootstrap for modern UI - Firebase for backend-less real-time updates & authentication - Recommendation: React + Tailwind + Firebase for portfolio-ready project

Step 3: MVP Features 1. Student login → view submitted clothes 2. Submit new cloth → auto-generate cloth ID 3. Caretaker login → view all clothes submitted 4. Caretaker marks clothes as washed 5. Real-time status update in student dashboard

Optional: - QR code for cloth ID - Upload cloth photo - Notifications / animation

Step 4: Frontend Development Plan Day 1 – Frontend UI & Navigation - Setup project structure - Create Student Dashboard UI - Create Caretaker Dashboard UI - Style using Tailwind / CSS

Day 2 – Interactivity + Firebase Integration - Integrate Firebase: authentication - Store clothes in Firestore: student ID, cloth ID, status, optional photo - Real-time UI updates when caretaker marks clothes as washed - Add notifications / small animation - Deploy on Netlify or Vercel

Step 5: Deployment & Resume Ready - Deploy live project → shareable link - Polish UI → responsive design - GitHub README: description, tech used, features, screenshots/live demo - Resume line example: Developed a Hostel Laundry Management System with student and caretaker dashboards using React, Tailwind CSS, and Firebase. Implemented real-time status updates, QR-based cloth IDs, and notifications.

Step 6: Optional Enhancements - Batch marking for caretaker - QR code scanner for cloth marking - Dark mode for caretaker dashboard - Filtering clothes by student / status