

Consolidate the task to create the detailed comprehensive PRD & other required docume...

Caldron Flex Master Requirements Pack

Consolidated PRD + BRD/MRD/FRD/NFR + Analytics, IA, Risks, Traceability

Version: 2.1 (finalized for build planning)

Date: August 20, 2025

Applies to: Caldron Flex All-in-One Printing Business Management System

Assumptions: Currency NPR; Timezone Asia/Kathmandu; Language EN/Nepali;

Initial capacity ~30 concurrent users, ~6 orders/day; Core brain on Ever-Gauzy;

e-commerce on Bagisto; App hosted on VPS (4 vCPU, 6 GB RAM, 100 GB SSD);

cPanel used as bulk file storage (1.5 TB). No online payments initially; WhatsApp via self-hosted proxy with email fallback.

0. One-look summary (what's in this pack)

- BRD: Why we're doing this, success, scope, constraints
- MRD: Market/users/competitors/differentiation
- FRD: Exactly what the system does (features, logic, rules)
- NFR: Performance, security, availability, i18n, a11y
- Reporting Requirements: Every report's purpose, columns, filters, owners, frequency
- Use cases & User stories: With acceptance criteria
- Traceability Matrix: BR → FR/NFR → design → tests
- Technical Feasibility: Stack, constraints, costs/benefits, go/no-go calls
- Information Architecture: Navigation, flows, data model overview
- Gap Analysis: Current vs target, priorities
- Risk Assessment: Technical, business, security, compliance with mitigations
- Implementation & Ops: Environments, CI/CD, storage, backup/restore, monitoring
- Integration Plan: Bagisto, WhatsApp proxy, email, storage, AR/VR, future PS API

- Test Strategy: Functional, integration, performance, security, UAT
- Stakeholder Questionnaires & Forms: For clients and clients' clients (end-users)
- AI Research Model Prompt: To find open-source/paid building blocks and scripts

1. Business Requirements Document (BRD)

1.1 Business goals

- Centralize and automate the entire custom printing lifecycle to reduce manual effort and errors.
- Increase throughput without hiring (target up to 3× current volume).
- Improve client experience with transparent proofs, revisions, and approval workflow.
- Establish a scalable foundation with bilingual UX (EN/Nepali) and auditability.

1.2 Success metrics

- 50–60% reduction in manual processing steps per order
- 80% fewer manual notifications (via automation)
- 90–95% first-submission approval rate (measured after 3 months)
- Ability to absorb 3× order volume with same staff
- SLA: <2 s P95 for typical page loads; <60 s to render previews for 90% of uploads <50 MB

1.3 Scope (initial)

- End-to-end order → design → client review/annotation → approval → production → invoice → payment tracking (cash/cheque) → ready for collection
- CRM for organizations, individuals, guests; organizational roles/permissions
- PIM with dynamic and manual pricing; seasonal promo codes (can be Phase 2)

- Inventory (mixed: raw + finished), purchasing (no barcode initially)
- File management up to 500 MB with TIFF→JPG preview and watermark, version annotations (not full versioning)
- Notifications via WhatsApp proxy + email templates
- Bagisto storefront for standardized items; Ever-Gauzy as the brain
- Bilingual UI: EN/Nepali

Out of scope (initial): online payments, shipping/logistics carriers, deep accounting, automated QC checkpoints, barcode, official WhatsApp Business API (for now).

1.4 Constraints

- VPS for app (4 vCPU/6 GB/100 GB). cPanel shared hosting for bulk storage (1.5 TB)
- No Docker on cPanel; long-running daemons best hosted on VPS
- WhatsApp via proxy (unofficial), must fallback to email

1.5 Key decisions (locked)

- Revision limit is a guideline (soft, not enforced)
- Rush orders: explicit "urgent" flag affects queue priority; no extra fee
- Credit: track balances; no credit limit enforcement
- Medal variants: Gold/Silver/Bronze same price individually; bundle also supported
- Seasonal pricing via discount codes with date ranges (Phase 2 ok)

2. Market Requirements Document (MRD)

2.1 Target segments

- Primary: Local SMBs and organizations ordering custom print (60%)
- Secondary: Individuals (30%)
- Tertiary: Walk-in/guests (10%)

2.2 Problems to solve

- Disorganized proofing and approvals → rework and delays
- Fragmented data across spreadsheets, chats, and paper
- Price confusion for custom items
- Poor visibility into order status; too much manual follow-up

2.3 Competitor landscape (representative)

- Print shop SaaS: Printavo, Ordant, InkSoft, OnPrintShop (end-to-end workflow)
- Generic ERPs/CRMs: Odoo, Zoho, ERPNext (broad but not tailored for print proofing)
- E-commerce: Shopify/WooCommerce (needs heavy plugins for print workflows)
- Open-source stacks: Ever-Gauzy (ERP/PM/CRM), Bagisto (e-commerce)

2.4 Differentiation

- Tailored print proofing with in-app annotations and bilingual UX
- Hybrid storage (VPS + cPanel 1.5 TB) optimized for large design files
- Open-source core with extensibility (API, role granularity, custom pricing engine)
- WhatsApp proxy + email for local comms preference; offline payments and credit flows

2.5 Go-to-market (internal use first)

- Internal rollout; later, offer as service to partner print shops if desired
- Emphasize fast approvals, fewer errors, transparent history

3. Functional Requirements Document (FRD)

3.1 User and roles

- System Admin: full system control
- Staff Admin: full operational access, can manage helper permissions, can see purchase prices

- Staff Helper: can do everything operationally except viewing purchase prices
- Org Admin: manage org, members, create/approve projects
- Org Member: create tasks, comment, approval if granted by Org Admin
- Individual: manage their own orders; Guest: minimal (name, phone)

3.2 Orders & projects

- Project groups multiple tasks/items; tasks move through statuses:
New → Claimed → Design In Progress → Awaiting Client Review → Changes Requested → Approved → In Production → Ready for Collection → Closed
- Urgent flag surfaces tasks to top with badge/color; sorting by deadline then urgent
- Auto-transitions based on actions (e.g., upload proof → Awaiting Client Review)

3.3 Proofing & files

- Accepted: TIFF, JPG, PNG, PDF, SVG, PSD; max 500 MB/file
- Auto convert TIFF/PSD to low-res watermarked JPG preview
- Annotation on preview: pins, rectangles, text comments; threaded replies
- Store original + annotation sets (not full file versions), audit timestamps
- Up to 5 correction rounds recommended (soft guidance)
- Digital sign-off with timestamp/name/IP; prevents further changes unless staff reopens

3.4 Products & pricing

- Fixed pricing items with attributes (e.g., Rs/sqft, size tiers, materials)
- Manual quote flow for complex items (Holding boards, custom)
- Staff override price with reason logging
- Seasonal promo codes with date ranges (Phase 2)
- Medal set bundle and individual medals (same price per color)

- Tax configurable (GST/VAT %), per-invoice
- Approximated price display allowed for certain items pre-quote

3.5 Inventory & purchasing

- Mixed tracking:
 - Finished-goods only for flex/banner where raw-to-custom conversion is complex
 - Raw + finished tracking for frames, medals, stamps, etc.
- Low-stock alerts; suppliers and purchase orders
- No automatic cost-based pricing; no barcodes initially

3.6 Invoicing & payments

- Auto-generate invoice at Ready for Collection
- Record payments: Cash/Cheque; partial payments supported; show outstanding
- Credit tracking for orgs; no credit limit enforcement
- No refunds post-approval; cancellation rules documented

3.7 Communications

- WhatsApp proxy integration for order updates, approvals, reminders
- Email fallback; all messages templated and bilingual
- In-app notifications for staff
- Message log per order (audit)

3.8 Complaints/tickets (our applied workflow)

- Any client can file a ticket linked to project/order
- Severity: Low/Medium/High (High pauses production automatically if not shipped)
- SLA timers and escalation (Helper → Admin after 24h on Medium/High)

- Resolution states: Open → In Review → Actioned → Client Confirmed → Closed
- Root-cause and resolution notes required before close; reportable metrics

3.9 E-commerce (Bagisto)

- Products synced from Gauzy master PIM to Bagisto catalog
- Orders from Bagisto sync back to Gauzy for fulfillment
- AR/VR product viewer on Bagisto product pages (uses GLB/GLTF/USDZ assets)
- Customer accounts unify via single identity model (email/phone mapped)

3.10 Administration

- Dynamic roles/permissions with granular toggles
- Tax rates, invoice sequences, currencies, languages
- Message templates (WhatsApp/email), file retention rules, backup settings

3.11 API

- REST API for core entities (orders, products, files, annotations, invoices)
- Webhooks for state changes (order status, approvals, payments)
- Auth via JWT with role-based claims; IP allowlists optional

4. Non-Functional Requirements (NFR)

- Performance: P95 page load <2 s for typical pages; preview generation start <5 s, completion <60 s for 90% files <50 MB; supports 30 concurrent users; designed to scale to 150 with VPS tier up
- Availability: 99.5% monthly; scheduled maintenance windows
- Security: RBAC; salted hashing; HTTPS only; per-file signed URLs; ClamAV scan on upload; rate limiting; audit log for all critical actions
- Data: daily DB backup; file backup policies (see Ops); retention 7 years for invoices, 2 years for raw uploads by default

- Localization: Full i18n for EN/Nepali (static and templates); Devanagari font embed for invoices/PDF
- Accessibility: WCAG 2.1 AA target for primary flows
- Observability: Request/DB/queue metrics; error tracking; structured logs

5. Reporting Requirements Specification

5.1 Daily Ops Dashboard

- Purpose: Snapshot of work in progress and bottlenecks
- Audience: Staff Admin, System Admin
- Frequency: Daily; live dashboard
- Content: Orders by status; urgent orders; due today/overdue; queue time; average correction rounds; tickets open
- Filters: date range; staff; product; client
- Parameters: org vs individual segmentation
- Owner: Staff Admin

5.2 Sales and Revenue (Monthly)

- Purpose: Revenue trends, product mix
- Audience: Management
- Frequency: Monthly
- Content: Invoices issued/paid; revenue by product/category; average order value; discount usage; tax collected
- Filters: date range; product; client type
- Owner: System Admin

5.3 Staff Productivity (Weekly)

- Purpose: Workload and throughput
- Audience: Staff Admin
- Frequency: Weekly

- Content: Tasks claimed/completed per staff; average cycle time per status; rework count; SLA breaches
- Filters: staff; date range
- Owner: Staff Admin

5.4 Inventory Status (Weekly)

- Purpose: Stockout risk
- Audience: Purchasing
- Frequency: Weekly
- Content: Low stock items; consumption trends; lead time projections
- Filters: category; supplier
- Owner: Staff Admin

5.5 Credit and Collections (Bi-weekly)

- Purpose: Outstanding balances
- Audience: Finance/Owner
- Frequency: Bi-weekly
- Content: Aging buckets; org credit balances; partial payments; promises to pay
- Filters: client; date range
- Owner: System Admin

5.6 Complaints/Tickets (Monthly)

- Purpose: Quality and CX
- Audience: Management
- Frequency: Monthly
- Content: Tickets by severity; time to first response; time to resolution; root-cause categories
- Filters: product; client; staff
- Owner: Staff Admin

5.7 Communication Log Export (On demand)

- Purpose: Audit
- Audience: Compliance/Owner
- Frequency: On demand
- Content: WhatsApp/email messages with timestamps and delivery status
- Filters: order; date range
- Owner: System Admin

6. Use cases and user stories (selected)

UC-01 Place Order (Guest)

- As a guest, I submit an order with name + phone, upload reference file, pick deadline, mark urgent if needed.
- Acceptance: System creates guest account, sends confirmation via WhatsApp/email, order visible in New queue.

UC-02 Staff Claims Task

- As a staff helper/admin, I view New queue sorted by deadline/urgent and claim a task.
- Acceptance: Task status becomes Design In Progress; claim is logged.

UC-03 Upload Proof and Notify

- Staff uploads source file; system auto generates preview and watermarks; notifies client.
- Acceptance: Status → Awaiting Client Review; preview loads in client portal; message sent via primary channel; email fallback if WhatsApp fails.

UC-04 Client Annotates and Requests Changes

- Client opens preview, adds annotations/comments, submits.
- Acceptance: Status → Changes Requested; annotations stored; staff notified.

UC-05 Client Approval (Digital Sign-off)

- Client approves final proof.
- Acceptance: Status → Approved; sign-off captured; further edits locked unless reopened by staff.

UC-06 Production and Ready for Collection

- Staff moves to In Production then Ready for Collection; invoice auto-generated and sent.
- Acceptance: Invoice shows partial/total due; tax applied if configured.

UC-07 Payment Recording

- Staff records cash/cheque; partial allowed.
- Acceptance: Balance updates; receipt generated; audit log updated.

UC-08 Complaint Ticket

- Client raises ticket with severity; if High and order not closed, system pauses production.
- Acceptance: SLA timers start; escalation if overdue; resolution notes mandatory.

UC-09 Inventory Low-stock Alert

- When item below threshold, alert appears in dashboard; optional email to purchasing.
- Acceptance: Alert persists until PO created or stock updated.

7. Traceability matrix (sample)

- BR-1 Reduce manual steps → FR-Queue, FR-AutoStatus, FR-Templates → Tests: workflow auto-transition; notif firing
- BR-2 Faster approvals → FR-Annotations, FR-Previews → Tests: file conversion time, annotation UX

- BR-3 Offline payments tracking → FR-Invoices, FR-PartialPayments → Tests: partial payment math, audit logs
- BR-4 Scalability → NFR-Perf/Avail/Security → Tests: load test @30 users; soak test; auth hardening
- BR-5 Bilingual UX → NFR-i18n → Tests: language toggle, template localization

8. Technical Feasibility Study

8.1 Stack choices (reasoning)

- Core: Ever-Gauzy (NestJS + Angular) for ERP/PM/CRM extensibility and RBAC
- Store: Bagisto (Laravel) for catalog/cart/AR viewer plugin ecosystem
- Queue: Redis + BullMQ on VPS for previews, notifications
- Image/preview: libvips/sharp for fast TIFF/PSD→JPG; ImageMagick fallback
- Uploads: Uppy + tus server (chunked/resumable) to VPS; background move to storage
- Storage abstraction: supports Local FS (VPS), S3 (MinIO/R2), and SFTP (cPanel) so we can use cPanel as bulk store

8.2 Hosting realities

- VPS runs app servers (Gauzy API/Angular UI, Bagisto PHP, Redis, ClamAV, tus server, WhatsApp proxy service)
- cPanel used as bulk file storage via SFTP (no long-running daemons feasible reliably)
- Optional: Cloudflare CDN for preview assets; signed URLs via app

8.3 WhatsApp proxy feasibility

- Use Node libraries (e.g., Baileys/WPPConnect/Venom) on VPS; keep single connection; add watchdog and autosign-in
- High risk vs official API; mitigations: queuing, retry with backoff; email fallback; in-app notifications

8.4 AR/VR feasibility

- Bagisto supports model-viewer/three.js plugins; we host GLB/GLTF/USDZ assets; for Phase 1 attach 3D assets manually per product; later automate generation if feasible

Conclusion: Feasible within given resources; careful with storage and WhatsApp reliability.

9. Information Architecture

9.1 Top-level navigation (app.caldronflex.com.np)

- Dashboard
- Orders/Projects (All, My Queue, Urgent, By Status)
- Proofing (Uploads, Previews, Annotations)
- Products & Pricing
- Inventory & Purchasing
- Clients (Organizations, Individuals, Guests)
- Invoices & Payments
- Tickets/Complaints
- Reports
- Admin (Roles, Templates, Tax, Integrations, Backups)

9.2 E-commerce (store.caldronflex.com.np)

- Home, Catalog, Product Detail (with AR/VR), Cart/Checkout, My Orders

9.3 Key flows

- Order submission flow (guest/registered) → confirmation
- Staff claim → design upload → preview generation → client review → approval
- Invoice creation → payment capture
- Ticket open → triage → resolution

- Inventory thresholds → purchasing

9.4 High-level data model (entities)

- User, Organization, Role, Permission
- Product, Variant, PriceMatrix, PromoCode
- Project, Order, OrderItem, Task, Status, Priority
- FileAsset (original), PreviewAsset (JPG), Annotation, Revision
- Invoice, Payment, CreditLedger
- InventoryItem, StockMovement, PurchaseOrder, Supplier
- Ticket, TicketComment
- Notification, MessageTemplate
- AuditLog

10. Gap Analysis (from today → target)

- Order workflow: manual chat/spreadsheet → unified pipeline (P1)
- Proofing: ad-hoc → annotated previews + digital sign-off (P1)
- Inventory: implicit → low-stock alerts + simple POs (P2)
- Reporting: manual tally → dashboards and scheduled reports (P2)
- E-com sync: disconnected → API sync with single source of truth (P2)
- WhatsApp: personal numbers → single controlled service with audit (P1)
- AR/VR: not present → Bagisto product viewer with asset uploads (P2)

11. Risk Assessment

- WhatsApp proxy instability: fallback to email, queue retries, operator alert; plan migration to official API later
- cPanel SFTP storage latency: use asynchronous file moves, local cache for hot previews, nightly sync; monitor backlog
- Large files CPU/memory: libvips, chunked uploads, file size/type validation; deny >500 MB

- Security: enforce least privilege, signed URLs, ClamAV, rate limiting, WAF via Cloudflare
- Integration drift (Bagisto↔Gauzy): bi-directional sync with reconciliation job; conflict policy tied to source of truth
- Training adoption: provide short videos and tooltips; staged rollout; collect feedback via in-app widget
- Data loss: daily DB backup; file backup policy; quarterly restore drills

12. Implementation & Operations

12.1 Environments

- Dev (VPS), Staging (VPS), Prod (VPS + cPanel storage)
- Domains: app., store., api., bck. subdomains

12.2 CI/CD

- GitHub Actions: lint/test/build; zero-downtime deploy to VPS
- Bagisto deploy via rsync/Envoy; migrations managed

12.3 Storage strategy

- Upload landing: VPS NVMe (fast)
- Background mover: to cPanel via SFTP nightly (or continuously in batches)
- Preview thumbnails: kept on VPS and optionally cached via CDN
- Optional: introduce MinIO on VPS later to use S3 semantics everywhere

12.4 Backups

- Database: nightly full + 15-min binlog; copies to bck.caldronflex.com.np
- Files: daily rsync snapshot from VPS previews to cPanel; weekly archive rotation (4 weeks)
- Test restores quarterly

12.5 Monitoring

- Uptime: Uptime-Kuma

- Metrics: Netdata/Prometheus-like; track CPU, memory, disk, queue depth, request latency, error rate
- Logs: central structured logs with retention 90 days

12.6 Security operations

- TLS via Let's Encrypt; HSTS
- Secrets vault (.env on VPS with access control)
- ClamAV scan on upload; reject infected
- RBAC reviews quarterly; audit log immutability

13. Integration Plan

- Bagisto: Catalog master in Gauzy; push to Bagisto; pull orders from Bagisto; map customers via email/phone
- WhatsApp proxy: Node service on VPS; webhook from app; retries; template management; fallback to email
- Email: SMTP relay (e.g., transactional provider); SPF/DKIM aligned
- Storage: SFTP to cPanel; abstraction allows S3 in future
- AR/VR: Use <model-viewer> with GLB/USDZ; asset upload admin in Bagisto; per-product linking
- Future: Adobe Photoshop API (Phase 2), Accounting integration

14. Test Strategy

- Unit tests for pricing engine, status transitions, permission checks
- Integration tests: upload → preview → notify; Bagisto sync; WhatsApp/email
- Performance tests: 30 concurrent; file conversion queue throughput; urgent tasks latency
- Security: OWASP top-10 scan; authZ checks; upload sanitization
- UAT: scenario walkthroughs with staff/admin and 2–3 client orgs
- Regression pack for each release; roll-back playbook

15. Reporting: required columns (examples)

- Orders by Status: Order ID, Client, Product, Status, Deadline, Urgent, Assigned Staff, Age (days), Last Action
- Revenue by Product: Product, Qty, Gross, Discounts, Tax, Net, Period
- Staff Productivity: Staff, Tasks Claimed, Completed, Avg Cycle Time, Reopen Count
- Inventory Low Stock: SKU, Name, On-hand, Threshold, Avg Weekly Use, Days Cover
- Credit Aging: Client, 0–30d, 31–60d, 61–90d, >90d, Total Due

16. Design Rationale (step-by-step highlights)

- Ever-Gauzy as brain: robust NestJS/Angular base with RBAC and project workflows; reduces custom ERP lift
- Bagisto for e-com: Laravel ecosystem; AR/VR plugins available; clean separation on store subdomain
- Uploads with tus/Uppy: resilient chunking for flaky networks and large files; resumable
- libvips/sharp over ImageMagick: significantly lower memory; faster TIFF/PSD handling
- SFTP to cPanel: works within shared hosting constraints; avoids need for FUSE/daemon; predictable backups
- WhatsApp proxy: acknowledges cost concerns; engineered with robust fallbacks
- Annotation via client-side library (e.g., Annotorious/Fabric.js): lightweight, fits preview overlay model
- Redis/BullMQ: simple reliable queues; isolates heavy work; easy to monitor

17. Stakeholder Questionnaires and Forms

17.1 Business stakeholder form (Caldron Flex)

- Organizational setup: departments, approvers, working hours, holidays
- Product catalog: list all products, attributes, pricing rules, tax applicability
- Discount/promo policy: codes, date ranges, eligibility (Phase 2)

- Inventory thresholds: min levels, supplier contacts, lead times
- Invoice template: legal fields, logo, footer notes, tax registration
- Communication: tone, languages, default channels, escalation timings
- Data retention: desired retention for originals, previews, tickets
- Access control: who can override prices, who sees purchase cost
- Backup windows: preferred time; maintenance window
- Reporting: which metrics are reviewed in weekly/ monthly meetings

17.2 Client organization onboarding form (your client's clients)

- Organization details: legal name, billing address, tax ID, primary contacts
- Approval rules: who can approve designs; alternates; SLAs
- Preferred communications: WhatsApp/email; hours to contact
- Payment terms: cheque cycles; credit eligibility
- Typical products and specs: sizes, materials, templates
- Branding assets: logos, fonts; color profiles; required file formats
- Security preferences: data sharing, confidentiality notes

17.3 Order submission form (end-customers)

- Customer type: org/individual/guest
- Contact: name, phone, email
- Product selection and attributes
- Dimensions/specs; quantity; deadline; urgent flag
- File upload: source + references; special instructions
- Language preference
- Approx budget (optional)
- Consent to terms: proof approval policy, no refunds post-approval

17.4 Ticket form (end-customers)

- Order ID, issue type (quality, delay, billing, other)

- Severity (Low/Medium/High)
 - Description; attachments
 - Preferred resolution; callback hours
18. Open items already decided (do not change in build)
- No credit limit enforcement
 - No rush fee
 - Medal colors same price individually
 - Seasonal promos via codes (Phase 2 is acceptable)
 - WhatsApp proxy used initially; plan fallback and observability
19. Notes for implementation specifics (storage/scripts)
- Use an abstraction layer so uploads can be switched between:
 - Local FS (VPS, fast cache)
 - SFTP (cPanel bulk storage)
 - S3-compatible (R2/MinIO/Wasabi) in future
 - Background job "FileMover":
 - When file reaches "cold" state (e.g., Approved + 7 days), move from VPS to cPanel via SFTP
 - Generate and keep small thumbnails on VPS for fast preview
 - Backups:
 - DB nightly dump to bck.caldronflex.com.np (separate cPanel account/folder)
 - Weekly encrypted tar archives; 4-week rotation
20. Minimal test cases (examples)
- Pricing override requires reason and logs actor/time
 - Upload of 450 MB TIFF: success, preview generated <120 s, virus scan passed

- WhatsApp proxy down: email sent; in-app notification still logs
 - Role "Helper" cannot see purchase price fields anywhere
 - Ticket severity High pauses production if not yet Ready for Collection
21. AI research model prompt (to find open-source/paid components and scripts)
Use this exactly as a starting brief.

Goal

Identify best-fit open-source or commercial components (and ready-to-use scripts) to implement a custom print business system with: Ever-Gauzy (NestJS/Angular) as ERP/brain, Bagisto (Laravel) as store, VPS app hosting (4 vCPU/6 GB/100 GB), and cPanel (1.5 TB) as bulk file storage over SFTP. Must support TIFF/PSD→JPG previews, resumable uploads, annotation, WhatsApp proxy integration, bilingual UI (EN/Nepali), AR/VR product viewer, reports, backups, and security hardening.

Deliverables (structure your answer)

1) Shortlist (3–5 per category) with: name, description, license/cost, maturity,