

## # Do Pahiyaa - Final SaaS Execution Plan

### ## 1. Scope and Objective

This plan defines the production path for **\*\*Do Pahiyaa\*\*** as a full SaaS platform:

- Public marketplace (B2C)
- Dealer channel (B2B)
- Lead unlock monetization
- Admin-controlled operations
- Plan-based dealer entitlements
- Auction lifecycle controls (admin-driven and auditable)

Target: production-grade system with strict RBAC, auditability, and operability.

---

### ## 3. Admin End-to-End Control (Mandatory)

Admin panel must provide full operational control without direct DB edits.

#### ## 3.1 User management

- Approve, block, unblock users
- Verify KYC
- Assign/revoke roles
- Force logout/session revoke
- Account risk flags

#### ## 3.2 Dealer management

- Approve/reject dealer onboarding
- Verify documents
- Assign plan and migrate plans
- Credit top-up/deduct with reason
- Suspend/reactivate dealer account

#### ## 3.3 Listing moderation

- Approve/reject/edit/remove listing
- Moderation notes and reason codes
- Bulk moderation actions

#### ## 3.4 Leads management

- Global lead inbox (search/filter/export)
- Lead status override (with reason)
- Fraud/spam tagging
- Lead reassignment (ops flow)

#### ## 3.5 Deals management

- Full deal timeline view
- State transition controls
- Cancellation reason tracking
- Commission lock/release controls

#### ## 3.6 Auction control

- Create/start/pause/cancel auction
- Reserve price, increment, anti-sniping settings
- Manual settlement override (audited)

- Bid dispute review queue

### ## 3.7 Billing and revenue

- Subscription lifecycle dashboard
- Lead-unlock revenue
- Commission and fees dashboard
- Refunds/adjustments with approval trail
- Invoice and tax export

### ## 3.8 Notifications and settings

- Broadcast notifications by audience
- Template management
- Pricing configuration
- Feature toggles
- Maintenance mode

### ## 3.9 Audit logs

- Immutable logs for all sensitive actions
- Before/after snapshots for updates
- Actor + timestamp + IP + resource id
- Exportable audit reports

---

## ## 4. Dealer Plan Engine (Core SaaS Monetization)

### ## 4.1 Plan model

Each plan defines machine-enforced entitlements:

- `max\_active\_listings`
- `monthly\_lead\_unlock\_limit`
- `included\_credits`
- `credit\_price\_per\_unit`
- `auction\_access` (boolean)
- `analytics\_tier` (basic/pro/enterprise)
- `team\_seats`
- `priority\_support` (boolean)

### ## 4.2 Required tables

1. `dealer\_plans`
2. `dealer\_subscriptions`
3. `dealer\_entitlements\_snapshot`
4. `credits\_ledger`
5. `usage\_counters\_monthly`
6. `billing\_adjustments`

### ## 4.3 Runtime checks

- Publish listing: block if `active\_listings` >= `max\_active\_listings`
- Unlock lead: block if insufficient credits or monthly cap reached
- Auction join/bid: block if `auction\_access=false`
- Analytics view: gate modules by `analytics\_tier`
- Team invite: block when seat limit exceeded

---

## ## 5. API Surface (Minimum Production Set)

### ## 5.1 Admin APIs

- `GET /api/admin/users`
- `PATCH /api/admin/users/:id/status`
- `PATCH /api/admin/users/:id/role`
- `GET /api/admin/dealers`
- `PATCH /api/admin/dealers/:id/verification`
- `PATCH /api/admin/dealers/:id/plan`
- `POST /api/admin/dealers/:id/credits/adjust`
- `GET /api/admin/listings`
- `PATCH /api/admin/listings/:id/moderate`
- `GET /api/admin/leads`
- `PATCH /api/admin/leads/:id/status`
- `GET /api/admin/deals`
- `PATCH /api/admin/deals/:id/state`
- `POST /api/admin/auctions`
- `PATCH /api/admin/auctions/:id/control`
- `GET /api/admin/revenue/summary`
- `POST /api/admin/notifications/broadcast`
- `PATCH /api/admin/settings`
- `GET /api/admin/audit-logs`

## ## 5.2 Dealer APIs

- `GET /api/dealer/dashboard`
- `GET /api/dealer/inventory`
- `POST /api/dealer/listings`
- `GET /api/dealer/leads`
- `POST /api/dealer/leads/unlock`
- `GET /api/dealer/plan`
- `POST /api/dealer/plan/topup`
- `GET /api/dealer/analytics`

## ## 5.3 Marketplace APIs

- `GET /api/listings`
- `GET /api/listings/:id`
- `POST /api/leads`
- `POST /api/offers`
- `PATCH /api/offers/:id/counter`

All write APIs require:

- Zod validation
- Role + ownership authorization
- Idempotency key for critical payments/unlocks
- Audit event emission for sensitive actions

---

## ## 6. Production Milestones

### ## Phase A - Platform Stabilization

- Fix build/type/lint blockers
- Remove duplicate conflicting services
- Align API contracts and DB types
- Resolve broken routes/links

**\*\*Exit criteria:\*\*** `check-types`, `lint`, `build` all pass.

### ## Phase B - Admin Control Completion

- Complete all admin modules in section 3
- Add immutable audit pipeline

- Add advanced filters and bulk actions

**\*\*Exit criteria:\*\*** Admin can operate full platform without SQL access.

### ## Phase C - Dealer Plan Enforcement

- Implement plan engine schema + services + APIs
- Enforce entitlement checks in every dealer flow
- Add credits ledger and usage meter

**\*\*Exit criteria:\*\*** dealer capabilities are strictly plan-gated.

### ## Phase D - Deal + Auction Hardening

- Complete deal lifecycle state machine
- Add anti-sniping, settlement workflows, dispute handling
- Add fraud controls and alerting

**\*\*Exit criteria:\*\*** auction/deal flows auditable and abuse-resistant.

### ## Phase E - Auth Enforcement Switch

- Switch `AUTH\_MODE=enforced`
- Enable Supabase JWT role mapping
- Remove dev bypass from production path

**\*\*Exit criteria:\*\*** pre-launch security checklist cleared.

---

### ## 7. Non-Functional Gates (Release Blockers)

- P95 API latency target defined per module
- Rate limiting on auth, lead create, lead unlock, bidding
- Structured logs + trace ids
- Error budget and alert thresholds
- Backup and restore runbook tested
- SAST/dependency audit clean
- RLS policy test suite green

---

### ## 8. Final Definition of Done

1. Admin has full platform control (users, dealers, listings, leads, deals, auctions, billing, settings).
2. Dealer panel behavior is fully plan-driven.
3. All critical actions are audited and recoverable.
4. Auth is fully enforced (no bypass in production).
5. CI gates pass consistently (`check-types`, `lint`, `build`, critical integration tests).