



HerbaCommunity

Feature & Functionality Reference



Wellness Club Management System

Multi-club network · Kolhapur, Maharashtra

4 August 2026

HerbaCommunity — Feature & Functionality Reference

Everything built to date, what each thing actually does, and how far it can be trusted.

Status legend

Mark	Meaning
✓	Built and verified by automated tests
●	Built, but only structurally checked — never executed
□	Not built

Front-end: `herbalife-dashboard.html` , ~9,800 lines, vanilla JS, single file, no build step. Backend: `api/` (Laravel).
Tests: `tests/` , 8 suites, `node run-all.js` .

1. Daily counter operations

1.1 Scan / attendance ✓

The core loop. Member arrives, staff enters their card number, attendance is logged, shake is served.

- **Two entry modes** — a 4-digit numpad (last 4 of the card) or full manual `HBL-XXXX` typing. Desktop keyboard input supported.
- **Live member preview** — as soon as 4 digits are entered, the member's name, photo initials, coach and remaining days appear *before* confirming, so staff can catch a wrong card.
- **Duplicate-scan guard** — a card already scanned today is rejected.
- **Expiry handling** — a card at its day limit opens the expired-card modal instead of scanning, offering grace days or a recharge.
- **Low-fuel prompt** — at 4 days or fewer remaining, a prompt surfaces so staff can start the recharge conversation early.
- **Cross-club recognition** (*new*) — scanning a card belonging to another club in the network names that club rather than saying "not found". It reveals the club only, never that club's member details.
- **Birthday greeting** — a member scanning on their birthday triggers a wish.

1.2 Guest experience ✓

Walk-in prospects who are not yet members get a complimentary shake and tea.

- Captures **name, mobile, date of birth** (all required), who invited them, reason for visiting (multi-select chips), and free-text notes.
- "Invited by" is an autocomplete over existing members and coaches.
- Guests land in a **pending list** until converted, and can be enrolled directly as a trial or gold member from there.
- Every guest is stamped with the club that served them.
- **Network-wide repeat detection** — see section 3.

1.3 Member registration

Three-step wizard: Identity → Coach & Health → Goals & Notes, with per-step validation.

- **Card ID auto-assigned** from the club's reserved number block.
- Captures DOB, anniversary, coach, batch, health conditions, goals, current and target weight, notes, and transformation flags.
- **Guest conversion** — pre-fills from a pending guest record and marks that guest converted, wiring up the referral reward.
- **Migration mode** — for importing an existing member mid-card: accepts scans already used, prior recharges, and a real card start date so history isn't lost.
- **Duplicate warning** — checks name / mobile / DOB across the whole network before creating. Not a hard block (people do move between clubs), but it must be acknowledged, and the acknowledgement is recorded.

1.4 Recharge

- **Gold** — full card, 30 days, ₹3,000 (configurable)
- **Trial** — 4 days, ₹300
- **Partial** — custom day count, priced per day (₹100/day)
- Outstanding grace days are deducted from the new card.
- Every recharge writes to the collection log for revenue reporting.
- A gold recharge counts toward Ambassador tiers.

1.5 Grace days

A member whose card has expired can be allowed a few extra visits before paying.

- Configurable maximum (default 4).
- Each grace day is logged with who granted it and when.
- Grace entries clear automatically on recharge.
- A dashboard widget and a full report track who is currently on grace.

2. Club network — the pyramid (new)

One head-quarter community can open clubs beneath it; those clubs can open clubs beneath them, to any depth.

2.1 Structure

- Each club is a node with a `parent`, forming a tree. The demo network is 8 clubs across 4 levels: Kolhapur (HQ) → Ichalkaranji / Sangli / Satara → Jaysingpur, Kurundwad, Miraj → Tasgaon.
- Every club carries its own owner, city, opening date and icon.
- **Reserved card-number blocks** — each club owns a range (HQ 1–3999, Ichalkaranji 4000–4499, ...) so `HBL-xxxx` IDs never collide network-wide, and a card number alone tells you which club issued it.

2.2 Network page

- **Hero panel** — the active club with its own figures beside the full downline rollup: clubs, members, served today, conversion %, guests, gift-eligible, expiring, repeat guests.

- **Downline pyramid** — an indented tree with proper connector rails, expand/collapse on any branch, and per-club live figures with a relative-size bar. Level badges (HQ / L1 / L2 / L3) are colour-coded by depth.
- **Club drill-down** — own-vs-downline breakdown, direct child clubs, latest guest experiences, and a button to operate as that club.
- **Performance leaderboard** — re-sortable by members, served today, guests, conversion %, or gift-eligible.
- **Open New Club** — creates a club under any club you control, validates a unique club code, and auto-reserves the next free card block.

2.3 Scope switching

Two controls drive what the entire app shows:

- **Active club** — which club you are operating as.
- **Scope** — *This Club Only* or *Include All Downline*.

Every screen respects both: dashboard KPIs, member lists, reports, exports and the printed report header all repaint for the current scope. Printed reports state which club and scope they were run at.

2.4 Visibility rules

Role	Sees
Admin	Own club + entire downline; may switch clubs; may open clubs
Manager	Own club + entire downline; may switch clubs
Coach	Own club only — and within it, only their own members
Front desk	Own club only

Enforced in the front-end *and* independently on the server. A role change mid-session forces the scope back and clears downline data from view.

3. Repeat-guest detection (new)

The guest experience is a **one-time offer across the entire network**, not per club. This is the machinery that enforces it.

3.1 How a person is matched

Two records are the same person if **either**:

1. the **mobile numbers** match (compared on the last 10 digits, so `+91 98230 07701` , `098230 07701` and `9823007701` all collide), **or**
2. the **name and date of birth** both match (name compared lowercased and letters-only, so spacing and punctuation don't matter).

Rule 2 is the important one — it catches somebody giving a different phone number at the next club, which is precisely the behaviour this exists to stop.

3.2 What staff see

As the counter types the mobile, name or DOB, a panel appears above the form listing **every prior record across all clubs**:

- **which club, what date, what time**
- guest ID, DOB, reason for the earlier visit, and whether they converted
- if the person is already an enrolled **member** anywhere, the panel turns red and shows their card ID, home club, coach and scans used

The check is **debounced** — one request per pause in typing, not one per keystroke.

3.3 What happens on save

- Saving requires an explicit confirmation naming the club and date.
- The record is stored flagged as a repeat, with the list of clubs already visited, and is written to the audit log.
- Repeat guests are badged  in the pending list and the registry.

3.4 Fail-safe behaviour

- **The server is authoritative.** It re-runs the check on save regardless of what the client did, and returns `409` with the prior visits attached unless the repeat is explicitly acknowledged. The live check is convenience only.
- **A failed check never reads as "all clear."** If the lookup can't run (offline, server down), the panel shows a warning telling staff not to issue the experience — an empty panel and "no prior visits" must never look alike.
- The duplicate check is **excluded from all caching** for the same reason.

3.5 Network guest registry

A searchable list of every guest experience across all clubs — by mobile, name or DOB — showing club, date, time, conversion status, and which other clubs the person has already visited.

4. Members & coaching

- **Member list** with search across name, card ID, phone and coach; filter by coach; chips for total / active / new counts.
 - **Member drill-down** — full profile, card status, referral chain, streak, monthly attendance, transformation tags.
 - **Edit member** — full record editing including coach reassignment.
 - **Delete** — admin only, with confirmation safeguards.
 - **Coach registry** — profiles, member counts, attendance and tier badges.
 - **Coach autocomplete** used across guest entry and registration.
 - **Batches** — configurable daily time slots with arrival heatmaps and average arrival time per batch.
 - **Bulk CSV import** — 3-step modal with template download, a robust parser, preview before commit, and migration support for existing members.
-

5. Health & transformation tracking

5.1 Body Fat Analysis (BFA)

A time series per member: weight, BMI, body-fat %, visceral fat, muscle mass, body age vs actual age, bone mass, BMR, subcutaneous fat, and notes.

- History view with deltas between readings.
- Prompted after recharge so measurements are taken at regular intervals.
- Exportable as a report.

5.2 Transformations

Categorised progress tracking — weight loss bands, sugar, BP, skin, joint/back, and free-text. Surfaced as tags on member cards and aggregated on a dedicated page.

5.3 Celebrations

Birthdays and anniversaries for today and tomorrow, with a dashboard banner and one-tap WhatsApp "send wish" links.

6. Referral fuel rewards

Members earn extra card days ("fuel") when someone they invited converts.

Award amounts, path-based:

Path	Fuel
Guest → trial	+1 day
Guest → gold (direct)	+3 days
Guest → trial → gold	+1 then +3 = 4 total

Fallback chain when the referrer can't receive it:

1. Active referring member → credited straight onto their card, extending it
2. Referrer inactive/expired → banked to *their* coach
3. No referrer → banked to the new member's coach
4. Otherwise → forfeited, and recorded as such

Idempotency — each guest can trigger each reward type once only, so a double-submitted conversion cannot pay twice.

The Referrals page has two tabs: the **referral network** (leaderboard and pairs) and **fuel rewards** (summary KPIs, coach bank balances, full reward log).

7. Ambassador tiers

Monthly recognition based on gold-card events brought in.

- **Members** earn from referrals; **coaches** earn from their members' gold activity.
- Only gold events count — new cards and gold recharges. Trials and partials don't, and a member's own recharge doesn't count toward their own tier.
- The counter resets on the 1st, computed live from event timestamps.
- **7 default tiers**, fully editable in Settings (name, thresholds, colours, add/remove):

Tier	Events
Ambassador	2–4
Silver Ambassador	5–6
Gold Ambassador	7–9
Platinum	10–14
VIP Platinum	15–19
Ruby	20–24
Elite	25+

- Badges appear on member drill-downs and coach cards.
- A monthly achievers report with a month selector and CSV download (UTF-8 BOM, so Excel opens it correctly).

8. Reports & exports

Dashboard KPIs, all computed live for the current club and scope: daily fuel served, new guests (today/week/month), irregular attenders, expiring cards, gift-eligible, transformations, celebrations, top attenders.

Report pages

- **Daily report** — collection/revenue, serves, new members, guests, recharges, with full drill-down detail
- **Gift eligible** — members with 26+ attendance days this month
- **Ambassador achievers** — monthly, with month selector
- **Grace period log** — who is on grace and how many days used
- **Pending guests** — not yet converted
- **Individual attendance** — per-member history

Exports (open a print view → save as PDF or screenshot for WhatsApp): gift, attendance, irregular, all members, body analysis, grace log, guest list.

Every printed report carries the club name, the scope it was run at, and a confidentiality marker.

9. Authentication & permissions

- **6-digit PIN login** with an on-screen numpad and desktop keyboard support.
- **Remember me** for 30 days; **10-minute auto-lock** on idle.
- **Wrong-PIN lockout** — 5 attempts, then a 30-second cooldown.

- **Four roles** — admin, manager, coach, front desk — gating 19 distinct permissions (scan, recharge, add member, edit member, delete, add guest, convert guest, settings, pricing, reports, coaches, users, activity log, BFA, dashboard, plus the three network permissions).
- UI adapts to role: buttons, nav items and settings tabs hide rather than error.
- **Activity log** — an audit trail of logins, member changes, recharges, club switches, repeat-guest overrides and cross-club registrations.

On the server the PIN is **bcrypt-hashed** and rate-limited per user and IP. In the standalone front-end it is client-side only and not a real security boundary — that is exactly why the backend exists.

10. Settings

Eight tabs: **General · Pricing · Batches · Holidays · Coaches · Tiers · Users · Activity**

- **Pricing** with full price history, so past revenue is computed at the price that was in force at the time, not today's.
- **Batches** — name, start time, duration, frequency, active days.
- **Holidays / closures** with day counts.
- **Tier editor** — add, remove and re-threshold Ambassador tiers.
- **User management** — role cards, coach linking, PIN reset, active toggle, deletion safeguards.
- **Activity log viewer** — filterable, colour-chipped, newest first.

11. Backend — Laravel API

Written, never executed. 21 tables, 19 models, 3 policies, 6 controllers.

- **branches** — self-referencing `parent_id` plus a materialised `path` (`/1/3/7/`), so a downline rollup is one indexed `LIKE` rather than a recursive query. Portable down to MySQL 5.7.
- **BranchScope** — a global query scope applied to every branch-owned model. Admin/manager roll up their subtree; coach/front-desk are pinned to their own club. **Fails closed:** no session or an unresolvable branch yields no rows, never all rows.
- **SetActiveBranch** — validates the requested club against the user's own subtree. A forged `X-Branch-Id` silently falls back to the home club. Checked twice, independently, as defence in depth.
- **PersonMatchService** — the network-wide duplicate lookup, and the *only* deliberate scope bypass in the codebase, via an explicit `acrossNetwork()` helper so every bypass is greppable.
- **Guest duplicate indexes** — `phone_normalised` and `(name_key, dob)` are stored as separate normalised columns and indexed, so matching is exact and fast.
- **Scan uniqueness** — a unique index on `(member_id, scanned_on)` plus a row lock. Two devices scanning one card at once produce one row, not two.
- **Money** — `DECIMAL`, never float. Recharge amounts are computed server-side from `price_history`; the client sends a card type, never a price.
- **Idempotency** — `reward_ledger` has a unique `(guest_id, reward_type)` constraint, making double-payment a database impossibility.

- **Seeders** — `NetworkSeeder` (clubs, staff, config — safe for go-live) and `DemoDataSeeder` (fabricated members and guests, skippable via `DEMO_DATA=false`).

12. Progressive Web App

Installable on iPhone and Android; works offline.

- **Manifest** with 3 shortcuts — Scan, New Guest, Club Network.
- **6 icons**, standard and maskable (art inside the safe zone for Android's launcher crop).
- **Service worker caching**, chosen for a counter on a patchy connection:
 - App shell → cache-first, refreshed in the background, so it opens instantly
 - API reads → network-first with a stale-marked cache fallback
 - Writes → queued in IndexedDB and replayed via Background Sync
 - **The duplicate check is never cached** — a cached "no prior visits" could wave through a repeat guest. Offline it fails loudly instead.
- **Install prompt** on Chrome/Edge; a one-time Add-to-Home-Screen hint on iOS, which never fires the install event.
- **Offline page** and a live connection banner.
- Update banner when a new version is available.

Requires the assets at the web root and **HTTPS** (localhost excepted).

13. API integration layer

The dashboard auto-detects the backend at boot.

- **Backend reachable** → API mode; every screen is fed from the server.
- **Backend unreachable** → demo mode behind an orange banner, fully functional on seeded data. The app is never a blank screen.

The design point: rather than rewriting ~300 render functions, the API layer **refills the same global arrays in place**. The render layer never learns where its data came from. Writes go through the API and patch local state from the response.

14. Testing

`tests/` — 8 Node suites, no dependencies, `node run-all.js`. Currently **8/8**.

Covers HTML structure, the club pyramid and rollup arithmetic, role gating across all clubs and scopes, the API layer against a mocked backend, the PWA bundle, Laravel migration symmetry, PHP structure with **case-sensitive** PSR-4 checks (Linux-critical), and an audit that every documented feature exists.

These do not prove the Laravel API runs. The PHP checks are structural only.

15. Not built ☐

Gap	Why it matters / what it needs
QR camera scanning	Still numpad/manual. Needs <code>getUserMedia</code> + <code>BarcodeDetector</code> ; iOS Safari lacks <code>BarcodeDetector</code> , so a JS decoder is required. HTTPS mandatory.
Editable health conditions & transformation categories	You asked for these to be editable in Settings like Tiers. Health chips are literal markup; <code>TRANS_CATS</code> entries hold filter functions , which must become declarative rules before a UI can edit them.
Auto-reminders	Missed scans, trial expiring. Straightforward with the Laravel scheduler once the backend runs.
WhatsApp automation	Manual <code>wa.me</code> links work today. Auto-send needs the WhatsApp Business API and a BSP account — impossible from the browser.
About / version screen	No version constant exists. Worth adding before deployment so you can tell which build a centre is running when something is reported.

Honest summary

The **front-end is complete and tested** — every feature above marked ☒ works and is covered by automated tests.

The **backend and PWA are written but have never run**. No PHP has been executed against this project. `php artisan migrate` will be the first real test, and it is normal for that to surface a few fixes — the likely candidates are listed in `api/INTEGRATION.md` .

The security model — branch scoping, forged-header rejection, scan uniqueness, server-side pricing — is currently **argued, not demonstrated**. A Pest feature suite proving those claims is the highest-value next step once PHP is available.