Товч танилцуулга

Доорх баримт нь **Local Retail Control Platform** (ажиллах нэр) бүтээгдэхүүн хөгжүүлэхэд зориулсан **Developer-facing** материалыг агуулна: **User Stories (MVP & Extended)**, **Functional Requirements** (техник хариулт болон приоритеттой), болон **Non-functional Requirements** (хүрэх ёстой үзүүлэлт, хязгаар). Баримт нь Android гар утсанд чиглэгдсэн mobile-first, owner-first, minimal UX зарчмыг баримтална.

A. Үндсэн хэрэглэгчийн дүрүүд (Actors)

- **Owner** — дэлгүүр/лангуу эзэмшигч (1–3 дэлгүүртэй, бүрэн хяналт хүснэ).
- **Seller** — худалдагч / касс ажилтан.
- **System / Backend** — сервер, синк, анхааруулга.
- **Guest Customer** — (заавал биш) онлайн захиалгын хэрэглэгч (ирээдүйд).

B. User Stories (MVP-д анхаарна)

MVP (энэхүү үе шатанд зайлшгүй хэрэгжих)

1. **Owner — Бүтээгдэхүүн бүртгэх**
    - As an *Owner*, I want to create product records (нэр, үнэ, анги, эхлэх үлдэгдэл) so that I can see stock levels without POS/barcode.
    - Acceptance:
        - Product бүртгэх боломж: нэр, SKU (гар бичмэл эсвэл autogenerated), unit, cost price (optional), sell price, initial quantity, note.

        - Бүртгэл бүр local DB-д хадгалагдаж, cloud руу синк хийнэ.
        - Минимум 2–3 interaction (Add → Save).

2. **Seller — Борлуулалт хийх (Quick Sale)**
    - As a *Seller*, I want to record a sale on my phone quickly so that inventory updates in real-time.
    - Acceptance:
        - Борлуулалт бүртгэхдээ 3 алхмаас ихгүй: Select product(s) → Quantity → Save/Sell.
        - Систем нь тухайн барааны үлдэгдэлээс суутгана, shift-т холбоно.
        - Offline байхад local лог үүсэх ба хойш синк хийнэ.
3. **Owner — Реал цагаар Dashboard**
    - As an *Owner*, I want a real-time dashboard showing current stock, daily sales summary, and seller performance so that I can monitor
    - Acceptance:
        - Dashboard: current total inventory value, items low stock (threshold), daily sales total, top-selling items, seller-wise totals (today).
        - Owner dashboard-д “last sync” timestamp.
4. **Seller Management & Shift Tracking**
    - As an *Owner*, I want to create seller accounts and start/stop shifts so that sales can be attributed and accountability tracked.
    - Acceptance:
        - Seller account: name, phone (optional), role.
        - Shift open/close with seller ID; sales during shift attributed.
        - Shift report: total sales, items sold, discrepancy (if counted).
5. **Loss / Anomaly Alert**

   - As an *Owner*, I want automated alerts for negative stock, unexpected shrinkage (sales not matching stock change), or repeated zero-count sales so that I can investigate loss/hide.
   - Acceptance:
      - Alert rules: negative inventory, sales > expected (manual adjustments), recurrent returns.
      - Alerts visible on dashboard and push/notification optional.

6. **Basic Audit & History**
   - As an *Owner*, I want a chronological log of inventory changes and sales so that I can audit who did what and when.
   - Acceptance:
      - Every change (sale, manual adjust, product add) logged with timestamp, actor, reason.
      - Log searchable by date, product, seller.

7. **Onboarding (Store setup)**
   - As an *Owner*, I want an onboarding flow to quickly set up store, add first products, and invite sellers.
   - Acceptance:
      - Guided flow: Create store → Add 5 products (suggest template) → Invite seller (SMS optional) → Start trial.

Extended / Post-MVP (Nice-to-have, priority medium)

1. Multi-store owner: нэг owner олон дэлгүүрийг хянах.
2. Public product listing & simple online order capture (owner confirms, seller processes).
3. Basic reconciliation report (cash vs sales recorded).

4. CSV экспорт/импорт (product list, sales).
5. Role-based fine-grained permissions.
6. SMS/Push reminders for low stock, shift reminders.
7. Simple returns/void sale flow with reason.
8. Soft-delete + restore for products/sales.

C. Functional Requirements (MVP prioritized, numbered)

Format: [ID] Title — Description — Priority — Acceptance / Notes

Inventory & Product Management

- **FR-01** Product CRUD — Нэр, код, unit, price, initial_qty, note. — *MVP*
- **FR-02 Stock Calculation Engine** — Борлуулалт, adjust, return үйлдлүүдээс үлдэгдлийг бодит цагт тооцно; local мөр нь conflict resolution. — *MVP*
- **FR-03 Manual Adjustment** — Owner эсвэл authorizd seller гар аргаар үлдэгдэл засварлах (шалтгаантай). — *MVP*
- **FR-04 Inventory History / Audit** — Бүх event audit log-т бүртгэгдэнэ (actor, time, change, reason). — *MVP*

Sales & Shifts

- **FR-05 Quick Sale Flow** — 3 алхмаас ихгүй, багтаамжтай UI. — *MVP*
- **FR-06 Shift Management** — Shift эхлэх/хаах, shift report. — *MVP*
- **FR-07 Sale Attribution** — Sale бүр seller_id, shift_id-т холбогдсон байна. — *MVP*

Owner Visibility & Alerts

- **FR-08 Dashboard** — Real-time summary (current value, top items, low stock, daily sales). — *MVP*

- **FR-09 Alerting Engine** — Low stock threshold per product, negative inventory, suspicious activity. — *MVP*

Sync & Offline

- **FR-10 Offline-First Architecture** — App must ажиллах офлайн, local DB, background sync when network available. — *MVP*

- **FR-11 Conflict Resolution** — Last-writer-wins with manual merge UI for Owner; or vector clock/simple CRDT pattern. — *MVP*

Security & Accounts

- **FR-12 Authentication** — Owner and seller login via phone + OTP or password. — *MVP*

- **FR-13 Authorization** — Role-based simple permissions (Owner > Manager > Seller). — *MVP*

- **FR-14 Data Encryption** — Local DB encrypted; transport over TLS. — *MVP*

Integrations & Export

- **FR-15 CSV Import/Export** — Product list and sales export. — *Post-MVP*

- **FR-16 Push & SMS** — Push notifications (required), SMS optional paid integration. — *Post-MVP*

Admin / Owner Controls

- **FR-17 Store Settings** — Currency (MNT), business name, low-stock thresholds, timezone. — *MVP*

- **FR-18 Pricing & Subscription Hooks** — In-app subscription management or external billing integration. — *Post-MVP*

D. Non-functional Requirements (measurable)

Performance & Scalability

- **NFR-01** App Launch Time: cold start < 4s on low-end Android (2GB RAM).
- **NFR-02** UI Response: core interactions (add sale, add product) < 1.5s on common network.
- **NFR-03** Sync Latency: under good network, delta sync < 5s; full store sync < 30s (for up to 2k products).
- **NFR-04** Offline Capacity: device local DB holds at least 6 months data or 10k events without degradation.

Reliability & Availability

- **NFR-05** Backend SLA (target): 99.5% monthly uptime for sync API.
- **NFR-06** Durable storage: daily snapshots retained 30 days; point-in-time restore for last 7 days.

Security & Privacy

- **NFR-07** Transport Security: TLS 1.2+ always.
- **NFR-08** Data at Rest: encrypt sensitive fields on device and server (AES-256).
- **NFR-09** Auth: OTP phone verification for Owner; token expiry 30 days with refresh token.
- **NFR-10** Least Privilege: RBAC enforced for operations (manual adjust restricted to Owner/Manager).

Usability & Localization

- **NFR-11** Language: Mongolian primary; UI strings externalized for translation.

- **NFR-12** Minimal Flow: core sale flow ≤ 3 taps; add product ≤ 3 screens.
- **NFR-13** Accessibility: support large text and basic contrast; simple icons.

Maintainability & Costs

- **NFR-14** Modular Architecture: clear separation of domain (inventory), sync, auth, UI.
- **NFR-15** Hosting Cost Target: for initial scale (100 stores, 100k events/mo) <$200/mo cloud cost (estimate target).
- **NFR-16** Telemetry: error + usage analytics enabled; sample events to protect PII.

Device Constraints

- **NFR-17** Android First: support Android 8.1 (API 27) and above.
- **NFR-18** Battery: background sync limited to periodic job scheduler; no heavy polling.

E. Data Model (concise)

Provide minimal schema attributes.

- **Store**
  - id, owner_id, name, location(optional), timezone, created_at
- **User (Owner / Seller)**
  - id, store_id, name, phone, role, last_online, created_at
- **Product**
  - id, store_id, name, sku, unit, sell_price, cost_price, low_stock_threshold, metadata
- **InventoryEvent**

    - id, store_id, product_id, type(enum: INITIAL, SALE, ADJUST, RETURN), qty_change, actor_id, shift_id (nullable), reason, timestamp
- **Sale**
    - id, store_id, seller_id, shift_id, items [{product_id, qty, unit_price}], total_amount, payment_method, timestamp
- **Shift**
    - id, store_id, seller_id, opened_at, closed_at, open_balance (if tracking cash), close_balance
- **Alert**
    - id, store_id, type, product_id (nullable), message, level, created_at, resolved

---

F. API & Sync Design (high-level)

- **Auth**
    - `POST /auth/otp/request` — request OTP
    - `POST /auth/otp/verify` — get token
- **Products**
    - `GET /stores/{id}/products` (delta via `?since=timestamp`)
    - `POST /stores/{id}/products` (create)
    - `PUT /stores/{id}/products/{pid}`
- **Sales**
    - `POST /stores/{id}/sales` — create sale (bulk items)

    - `GET /stores/{id}/sales?since=...`
- **InventoryEvents**
    - `POST /stores/{id}/inventory-events` (for manual adjust, returns new stock)
    - `GET /stores/{id}/inventory-events?since=...`
- **Sync Endpoint (delta)**
    - `POST /sync` — client sends local changes (batched), server returns authoritative deltas and conflict hints.
    - Conflict resolution policy: server accepts events in time-order; if conflict, mark for manual review (owner UI).
- **Alerts**
    - `GET /stores/{id}/alerts`
- Use lightweight JSON; batch operations to reduce mobile data.

G. UI Screens & Flows (minimal)

1. **Onboarding**
    - Welcome → Create Store → Add first products quick (template: baby shop common items) → Invite seller (phone)
2. **Home / Dashboard (Owner)**
    - Snapshot: Today sales, low stock count, suspicious alerts, top 3 items.
3. **Quick Sale (Seller)**
    - Search product (typeahead) → set qty → confirm sale (one tap) → print/save.

4. **Products List & Add Product**
    - List with current qty, low-stock highlight → Add product modal (≤3 fields visible first).
5. **Shift Management**
    - Open shift (select seller) → show active shift card → Close shift (summary).
6. **Inventory Audit**
    - View events, filter by product/date/seller → flag an event for investigation.
7. **Settings**
    - Store settings, thresholds, sync status, subscription.

H. Acceptance Tests / QA Scenarios (examples)

- Create product → perform sale offline → reconnect → verify server stock decremented correctly and owner dashboard updates.
- Two sellers sell same product offline on separate devices → sync → check conflict resolution and total stock correctness.
- Manual adjustment by owner creates audit entry with reason and triggers alert if negative.
- Low stock threshold triggers notification and appears on dashboard.
- Role restriction: Seller cannot perform manual inventory adjustments.

I. Metrics & Success Criteria (maps to business goals)

- **Daily Active Sellers (DAS)** — target: 70% of invited sellers daily use within 1st month.
- **Owner Dashboard Check Rate** — target: Owner opens dashboard ≥1/day for pilot stores.
- **Stock Loss Detection** — measurable reduction in unaccounted loss in pilot store after 3 months.
- **Time per sale** — average < 10s from open app to sale recorded.
- **Sync reliability** — <1% sync-conflict requiring manual resolution at pilot.

J. Implementation Recommendations & Architecture (practical)

- **Mobile**: Native Android (Kotlin + Jetpack Compose) with Room DB, WorkManager for background sync.
- **Backend**: RESTful API (Node.js / Python), PostgreSQL, Redis for pub/sub / real-time notifications.
- **Sync**: Use delta-based sync with change-sets (batched), optimistic UI updates.
- **Auth**: Phone + OTP (integrate SMS provider optionally); token-based.
- **Hosting**: Start with one cloud region (affordable tier), autoscale later.
- **Monitoring**: Sentry/Crashlytics + basic analytics.
- **Data Export**: Provide CSV download via API for owner.

K. Privacy, Legal & Compliance (short)

- Store owner data and seller PII minimal: name + phone only; retain audit logs for configurable period (default 12 months).
- Provide data export / delete on owner request (privacy requirement).
- If using SMS, comply with local telecom consent rules.

L. Roadmap Suggestion (12–24 months)

1. Pilot (family baby shop): stabilize MVP, collect feedback (0–3 months).
2. Expand to local stores nearby, add CSV import/export, multi-store (3–9 months).
3. Online order channel + simple public listing (9–15 months).
4. Logistics integration & marketplace aggregation (15–24 months).

M. Deliverables for Developers (actionable checklist)

- API spec (OpenAPI) for endpoints in section F.
- DB schema migration scripts for data model.
- Mobile wireframes for screens in section G.
- Implement offline-first sync module with unit tests.
- RBAC enforcement and OTP auth flows.

- E2E test scenarios (section H).
- Monitoring & error reporting setup.

N. Quick Risks & Mitigations (developer-focused)

- **Risk**: Conflicts from offline multi-device usage. → **Mitigation**: deterministic event ordering + manual merge UI.
- **Risk**: Poor UX slows adoption. → **Mitigation**: enforce ≤3 interactions for core flows; measure time-to-sale.
- **Risk**: Owner distrust to input data. → **Mitigation**: audit trail + easy rollback + show value (loss reports).
