Product Shop

Doc type: technical · **Version:** v0.1 · **Status:** published · **Module slug:** product-shop
Exported: 2026-05-15 11:12 UTC · **By:** anonymous

Product Shop – Technical Specification

1. Module Purpose & Scope (Authoritative)

Product Shop is Primoro's one-off, clinician-recommended commerce module, allowing patients to purchase dental and oral-care products directly through the Primoro patient app using the same DNA Payments stack as the rest of the platform. The module converts clinical recommendations into immediate, trusted purchases — supporting ad-hoc and high-value product sales (e.g. electric toothbrushes, whitening kits) — while eliminating stock, fulfilment, and payment-admin burden for practices.

It governs:

- End-to-end one-off product purchases, from clinician recommendation through to payment capture and fulfilment
- The Product catalogue (Primoro-managed) and practice-level pricing configuration
- Order lifecycle management, including payment, fulfilment tracking, and refunds
- Commerce communications and staff tasks via Communication Hub

It explicitly does not:

- Manage recurring billing or subscriptions (owned by Hygiene Subscriptions)
- Create or modify care plans or clinical entitlements (out of scope for this module)
- Write back to any PMS or clinical record system (see §8)
- Operate as an open marketplace or retail catalogue

---

2. Ownership & Responsibilities

2.1 Product Shop IS Responsible For

- Creating and maintaining Order records from the point of confirmed payment through to completed fulfilment
- Enforcing the Order state machine (Created → Paid → Fulfilment-In-Progress → Completed → Refunded) with full audit trails
- Surfacing products in a clinical context (clinician-recommended framing only)
- Initiating and recording refunds via Integrated Payments, including mandatory reason codes and actor attribution
- Emitting Communication Hub events for all order lifecycle stages
- Generating staff tasks when human action is required (e.g. in-clinic fulfilment, exception handling)
- Exposing dashboard metrics (revenue, conversion, fulfilment status, refunds)

2.2 Product Shop IS NOT Responsible For

- Recurring billing or subscription management (owned by Hygiene Subscriptions)
- Raw payment processing or gateway integration (owned by Integrated Payments)
- Clinical record creation or modification (owned by the relevant clinical record module and enforced as a hard boundary — see §8)
- VAT reporting and financial reconciliation beyond order-level data (owned by Finance)
- Access role definition and authentication (owned by Access Manager)
- Audit log storage and exportability infrastructure (owned by Audit & Compliance)

---

3. Core Objects (Normative)

3.1 Product (Canonical Artefact)

A Product is a governed catalogue artefact representing a dentally relevant physical, in-clinic, or digital item that a practice makes available for patient purchase.

Minimum required fields:

- ProductID
- Name
- Description
- Imagery (Primoro-managed catalogue)
- Category
- FulfilmentType (Drop-Ship / In-Clinic / Digital)
- CostPrice (internal, not exposed to patient)
- RetailPrice (practice-set)
- AvailabilityStatus
- Optional PracticeNote

Shared Catalogue Contract with Hygiene Subscriptions

The Product catalogue is the single authoritative source for all Primoro product data. Hygiene Subscriptions composes subscription plans from the same catalogue and therefore shares the same product records, pricing fields, and availability rules. The following constraints MUST be enforced at the catalogue layer to ensure consistency across both modules:

- A RetailPrice or AvailabilityStatus change on any catalogue Product MUST be propagated immediately and consistently to all surfaces that consume it, including Product Shop and Hygiene Subscriptions.
- Product Shop MUST NOT maintain a local copy of product pricing or availability that is independent of the canonical catalogue record.
- Where a product appears in both a Hygiene Subscriptions plan and a Product Shop recommendation simultaneously, the same RetailPrice and AvailabilityStatus applies to both; divergent pricing between surfaces is not permitted.
- Deprecation or removal of a Product from the catalogue MUST trigger an availability check across all active Product Shop orders and Hygiene Subscriptions plans that reference it; the relevant module owners MUST be

notified as part of the deprecation workflow.

3.2 Order (Canonical Artefact)

An Order is a governed digital artefact representing a confirmed, paid purchase by a patient for one or more Products.

Minimum required fields:

- OrderID
- PatientID
- LineItems (one or more, each with FulfilmentType)
- PaymentRef (DNA Payments transaction reference)
- OrderStatus
- FulfilmentRecords (per line item)
- CreatedBy (authenticated actor)
- CreatedAt
- AuditTrail (immutable)
- RedemptionRef (optional — populated when Rewards Manager points are redeemed against this order; see §5.4)

Orders MUST NOT exist in an unpaid state. An Order record is created only after successful payment capture (or, where applicable, after a confirmed zero-balance settlement when the full order value is covered by a points redemption).

3.3 Order State Machine (Authoritative)

States:

- **Created** — payment capture confirmed; order record instantiated
- **Fulfilment-In-Progress** — one or more line items dispatched or awaiting in-clinic collection
- **Completed** — all line items fulfilled
- **Refunded** — full or partial refund applied against the original payment

Rules:

- An Order transitions to Created only on confirmed DNA Payments capture; no earlier state exists
- State transitions are auditable and time-stamped on every change
- Orders MUST NOT revert from Completed to Fulfilment-In-Progress
- Refunded is a terminal overlay state; a Completed order may carry a Refunded status if a post-completion refund is applied
- Mixed-fulfilment orders track state per line item; the parent Order status reflects the least-advanced line item until all are fulfilled.
- Only authenticated, named staff members may initiate a Refunded transition; system-initiated refunds are prohibited

4. Product Recommendation & Surfacing

4.1 Recommendation Model (Authoritative)

Products MAY be surfaced:

- during or after appointments
- via explicit clinician recommendations
- via contextual prompts (AI-assisted, optional — see §7)

The module MUST:

- Retain clinical framing at all times (e.g. "Recommended by your care team")
- Require that AI-assisted prompts are approved by a clinician or staff member before being surfaced to a patient (see §7)

The module MUST NOT:

- Use retail discovery patterns (e.g. "customers also bought", "trending products")
- Surface products outside a clinical or care-team context

4.2 Catalogue Management (Authoritative)

The module MUST:

- Draw products exclusively from the Primoro-managed catalogue
- Allow practices to set RetailPrice per product
- Allow practices to attach an optional PracticeNote per product
- Enforce AvailabilityStatus gating (unavailable products MUST NOT be purchasable)

The module MAY:

- Allow practices to configure which products are visible to their patients

---

5. Checkout & Payments (Authoritative)

5.1 Payment Provider (Locked)

All Product Shop payments MUST be processed via the Integrated Payments module, which provides the unified, platform-managed payment layer for all Primoro commerce flows. Integrated Payments is the exclusive entry point for payment initiation, capture, and refund; Product Shop MUST NOT implement any duplicate payment initiation logic or communicate with a payment gateway directly.

- Payments Provider (Locked): **DNA Payments** (via Integrated Payments)
- Accepted methods: card and wallet payments only (contactless, Apple Pay, Google Pay where supported by DNA Payments)
- Direct Debit: NOT permitted
- Third-party gateways (Stripe, PayPal, or any other): NOT permitted

- The Product Shop module uses the DNA Payments stack but does NOT require the Integrated Payments module to be separately enabled

5.2 Payment Flow (Authoritative)

1. Patient initiates checkout in the Primoro patient app
2. Payment intent is created via Integrated Payments
3. DNA Payments captures funds immediately
4. On success: receipt is issued, Order record is created, fulfilment is triggered
5. If payment fails: checkout is aborted, no Order record is created, patient is prompted to retry via the Integrated Payments UI

Payment Linkage and PMS Invoice Reference

When an Order is created, Product Shop MUST supply the `order_id` to Integrated Payments as the commerce order reference on the corresponding Payment record. A PMS invoice reference MAY be absent at the point of payment capture and MAY be populated after fulfilment is confirmed, in line with the Integrated Payments payment-linkage model. Product Shop MUST NOT block Order creation or fulfilment on the presence of a PMS invoice reference; it is an optional enrichment field only.

5.3 Refunds & Exceptions (Authoritative)

The module MUST:

- Support full and partial refunds via Integrated Payments
- Reference the original DNA Payments transaction and original payment method on every refund
- Require a mandatory reason code at the point of refund initiation, selected from the Integrated Payments-defined reason-code set
- Attribute every refund action to a named, authenticated staff member
- Create an immutable audit entry for every refund action
- Surface payment exceptions in Communication Hub dashboards

Anonymous or system-initiated refunds are NOT permitted.

5.4 Rewards Manager Redemption (Authoritative)

Where Rewards Manager is enabled for a practice, patients MAY redeem earned loyalty points against Product Shop purchases. The following rules govern this integration:

- Product Shop MUST consume redemption-confirmation events from Rewards Manager before finalising the checkout total. Points deduction is initiated by Rewards Manager; Product Shop MUST NOT deduct or manipulate points balances directly.
- Points redemption and payment capture MUST be handled atomically. If the Rewards Manager redemption confirmation is not received, checkout MUST NOT proceed to DNA Payments capture. If DNA Payments capture subsequently fails after a redemption confirmation has been issued, Product Shop MUST emit a redemption-reversal event to Rewards Manager so that points are reinstated.
- Where a redemption covers the full order value, no DNA Payments charge is raised; the Order record is created on receipt of the Rewards Manager settlement confirmation, and `PaymentRef` MAY carry a Rewards Manager settlement reference in place of a DNA Payments transaction reference.

- Where a redemption covers only part of the order value, the residual balance MUST be captured via Integrated Payments (DNA Payments) in the same checkout session.
- The `RedemptionRef` field on the Order record (see §3.2) MUST be populated with the Rewards Manager redemption identifier for all orders where points are redeemed, to ensure full auditability.
- Refunds on orders that included a points redemption MUST be coordinated with Rewards Manager; the points element MUST be reinstated by Rewards Manager on refund confirmation, not by Product Shop directly.

---

6. Fulfilment & Delivery Model

6.1 Fulfilment Types

Each line item MUST declare one of the following fulfilment types:

- **Drop-Ship** — dispatched via Primoro supplier network
- **In-Clinic** — collected from practice stock
- **Digital** — delivered electronically (e.g. vouchers)

Mixed-fulfilment orders (line items with different fulfilment types) are supported and tracked per line item.

6.2 Fulfilment Lifecycle

The module MUST:

- Trigger fulfilment only after confirmed payment capture
- Write tracking updates to the Order record at each fulfilment stage
- Automatically generate patient notifications for shipment and pickup readiness via Communication Hub

In-Clinic fulfilment MUST generate a staff task in Communication Hub when stock is to be prepared for patient collection.

---

7. AI Boundaries (Non-Negotiable)

AI MAY:

- Suggest contextually relevant products to clinicians or staff for recommendation to patients, drawn from the Primoro-managed catalogue
- Summarise order or recommendation activity for staff review
- Generate contextual prompts to surface products to patients, subject to explicit staff or clinician approval before delivery

AI MAY NOT:

- Surface a product recommendation directly to a patient without a named staff member or clinician approving the recommendation first
- Initiate, approve, or reverse any payment or refund action
- Modify Order state or fulfilment records
- Bypass access control, audit logging, or any governance check

- Make commitments on behalf of the practice

---

8. Audit & Compliance

The system MUST log:

- All Order state transitions, with actor identity and timestamp
- All refund actions, including reason code, actor, amount, and reference to the original DNA Payments transaction
- All read and write operations on patient-bound Order and fulfilment records
- All AI-generated product suggestions, recording whether each was approved or rejected by a staff member
- All Communication Hub events emitted by this module
- All catalogue changes (product availability, pricing) with actor and timestamp
- All Rewards Manager redemption events associated with an Order, including redemption reference, points amount, and whether a reversal was subsequently issued

Audit logs MUST be immutable, tamper-evident, and exportable for inspection via the Audit & Compliance module.

8.1 Record Governance (Authoritative)

Product Shop is Primoro-native only.

- NO PMS write-back of any kind
- NO write to Dentally notes, flags, or any clinical record field
- NO clinical record modification

All commerce activity — orders, payments, fulfilment records, refunds — remains exclusively within Primoro records and dashboards.

---

9. Access Control

Access control is enforced via Access Manager roles.

Action	Permitted roles
Browse product catalogue	Authenticated patient (own app session)
Initiate checkout / purchase	Authenticated patient
View orders (own)	Authenticated patient
View all practice orders	Practice Admin, Clinician (read)
Configure product catalogue / pricing	Practice Admin

Initiate refund	Named, authenticated staff member
View dashboards and reports	Practice Admin, Finance role

MFA MUST be required for any refund initiation action, in alignment with Integrated Payments security standards.

Anonymous or unauthenticated access to any order or payment surface is NOT permitted.

---

10. Delivery Surfaces & Access (Authoritative)

10.1 Patient Mobile App

The patient mobile app is the primary purchase and tracking experience. Patients browse recommended products, complete checkout via Integrated Payments UI, and track order and fulfilment status within the app.

10.2 Web Portal (Staff)

Staff access order management, refund initiation, fulfilment exception handling, and Product Shop dashboards via the Primoro staff web portal.

10.3 Tablet App

In-clinic use (e.g. clinician recommending a product chairside) MAY be supported via the tablet app surface; specific interaction design is governed by the UX Specification.

10.4 Engagement Signals

The module emits the following for staff visibility and analytics:

- Recommendation-to-purchase conversion rate
- Total revenue and order volume (dashboard)
- Fulfilment status and open exceptions
- Refund volume and reason-code breakdown

---

11. Integration Contracts

11.1 Inbound (this module consumes from)

From module	What	Contract
Integrated Payments	Payment capture confirmation, refund processing, payment exception signals	Sync API
Access Manager	Role and authentication context for all staff and patient actions	Sync

Appointment Manager	Appointment context for post-appointment product surfacing	Event / async
Rewards Manager	Points redemption confirmation and reversal events for product purchases	Event / async

Rewards Manager Inbound Detail

When Rewards Manager is enabled, Product Shop consumes two event types from Rewards Manager:

- **RedemptionConfirmed** — signals that a patient's points have been successfully deducted; Product Shop uses this to finalise the checkout total and proceed to payment capture (or order creation if fully covered by points).
- **RedemptionReversed** — signals that a previously issued redemption has been reversed by Rewards Manager (e.g. following a cancellation); Product Shop uses this as a reconciliation signal only and MUST NOT initiate a reversal independently.

11.2 Outbound (this module emits to)

To module	What	Contract
Communication Hub	Order confirmation, payment receipt, shipment/pickup readiness, refund/cancellation, delivery exception notifications and staff tasks	Event
Audit & Compliance	Immutable audit entries for all state transitions, refunds, AI suggestions, catalogue changes, and Rewards Manager redemption events	Event
Finance	Order-level revenue and refund data for reconciliation and VAT reporting	Read / async
Financial Insights	Revenue events for each confirmed order and refund, enabling aggregation of product sales within practice-level financial reporting	Event / async
Rewards Manager	Redemption-reversal requests triggered by post-payment order cancellation or refund where points were originally redeemed	Event

Financial Insights Outbound Detail

Product Shop MUST emit a revenue event to Financial Insights for every confirmed Order and for every refund processed. These events enable Financial Insights to include product sales and refunds in practice-level revenue aggregation without requiring Financial Insights to query Product Shop directly. The event payload MUST include, at minimum: `order_id`, `amount`, `currency`, `event_type` (sale or refund), and `timestamp`. The precise event schema is to be agreed with the Financial Insights module owner before implementation (see Open Questions).

11.3 PMS Boundary

Product Shop has no integration with any PMS. There is no write-back to Dentally or any other clinical system. All order, payment, and fulfilment records are Primoro-native and remain entirely within Primoro's data boundary.

---

12. Integration Summary

- **Integrated Payments** — inbound payment capture and refund processing via DNA Payments; hard dependency for all checkout flows; sole permitted payment initiation path
- **Communication Hub** — outbound events for all order lifecycle communications and staff task generation
- **Access Manager** — RBAC and authentication enforcement for all order, refund, and catalogue actions
- **Audit & Compliance** — immutable event log for all governed actions
- **Finance** — outbound order and refund data for reconciliation and VAT reporting
- **Financial Insights** — outbound revenue events for each confirmed order and refund, enabling product sales to be included in practice-level financial reporting
- **Hygiene Subscriptions** — explicitly out of scope for billing; shares the same Primoro-managed product catalogue and is subject to the same pricing and availability consistency rules (see §3.1)
- **Appointment Manager** — inbound appointment context for post-appointment product surfacing
- **Rewards Manager** — inbound points redemption confirmation and reversal events; outbound redemption-reversal requests on refund or cancellation

---

13. Explicit Non-Goals

- Recurring billing or subscription product offerings (would be owned by Hygiene Subscriptions)
- Open retail discovery or marketplace browsing (would require a separate retail module; not currently planned)
- PMS or clinical record integration of any kind
- Stock inventory management (practices using In-Clinic fulfilment manage stock outside Primoro)
- VAT calculation or financial reporting beyond order-level data (owned by Finance)
- Direct manipulation of Rewards Manager points balances (points are debited and reinstated exclusively by Rewards Manager)

---

14. Versioning & Governance

This specification is owned by: Product Owner, Product Shop module.

Changes to this spec require:

- Review by the Post-MVP module owner
- Impact analysis across all declared related modules
- Version bump (patch / minor / major as appropriate)

---

15. Build Contract (Engineering & QA)

15.1 Canonical Data Model

(Full schema to be defined during engineering sprint; the following captures the canonical objects and minimum fields established in this spec.)

```
product (
 product_id          UUID PRIMARY KEY,
 name                TEXT NOT NULL,
 description         TEXT,
 imagery_ref         TEXT,
 category            TEXT,
 fulfilment_type     ENUM('Drop-Ship','In-Clinic','Digital') NOT NULL,
 cost_price          DECIMAL NOT NULL,
 retail_price        DECIMAL NOT NULL,
 availability_status TEXT NOT NULL,
 practice_note       TEXT,
 created_at          TIMESTAMP NOT NULL,
 updated_at          TIMESTAMP NOT NULL,
 created_by          UUID NOT NULL  -- FK to staff/admin actor
)

order (
 order_id            UUID PRIMARY KEY,
 patient_id          UUID NOT NULL,
 payment_ref         TEXT,           -- DNA Payments transaction reference; NULL when fully covered b
 redemption_ref      TEXT,           -- Rewards Manager redemption identifier; NULL when no points re
 pms_invoice_ref     TEXT,           -- optional; MAY be populated after fulfilment
 order_status        ENUM('Created','Fulfilment-In-Progress','Completed','Refunded') NOT NULL,
 created_by          UUID NOT NULL,
 created_at          TIMESTAMP NOT NULL,
 audit_trail         JSONB NOT NULL  -- immutable append-only
)

order_line_item (
 line_item_id        UUID PRIMARY KEY,
 order_id            UUID NOT NULL,  -- FK to order
 product_id          UUID NOT NULL,  -- FK to product
 quantity            INTEGER NOT NULL,
 unit_price          DECIMAL NOT NULL,
 fulfilment_type     ENUM('Drop-Ship','In-Clinic','Digital') NOT NULL,
 fulfilment_status   TEXT,
 fulfilment_ref      TEXT
)

refund (
 refund_id           UUID PRIMARY KEY,
```

```
 order_id            UUID NOT NULL,  -- FK to order
 payment_ref         TEXT NOT NULL,  -- original DNA Payments transaction reference
 amount              DECIMAL NOT NULL,
 reason_code         TEXT NOT NULL,  -- from Integrated Payments-defined set
 initiated_by        UUID NOT NULL,  -- named, authenticated staff member
 initiated_at        TIMESTAMP NOT NULL,
 audit_entry         JSONB NOT NULL  -- immutable
)
```

15.2 Core Behaviour Rules

1. An Order record MUST NOT be created until DNA Payments confirms successful fund capture, or — where the full order value is covered by a Rewards Manager points redemption — until Rewards Manager confirms the redemption settlement.

2. A failed payment MUST result in checkout abort and zero Order records; the patient MUST be prompted to retry via the Integrated Payments UI.

3. Fulfilment MUST NOT be triggered until Order status is Created (i.e. payment confirmed).

4. Every Order state transition MUST write an immutable, time-stamped entry to the Order's AuditTrail with the actor's identity.

5. Every refund MUST carry a reason code from the Integrated Payments-defined set and MUST be attributed to a named, authenticated staff member; system-initiated refunds MUST be rejected.

6. Every refund MUST reference the original DNA Payments transaction reference and original payment method.

7. Products with AvailabilityStatus set to unavailable MUST NOT be purchasable; checkout for such products MUST be blocked.

8. Product recommendations surfaced to patients via AI-assisted prompts MUST have a recorded staff or clinician approval event before delivery; unconfirmed AI suggestions MUST NOT reach the patient.

9. All Product Shop payments MUST route through Integrated Payments (DNA Payments); any attempt to use an alternative gateway MUST be rejected. Product Shop MUST NOT implement payment initiation logic independently of Integrated Payments.

10. No write of any kind MUST be made to any PMS or clinical record system; any such integration attempt MUST be treated as a critical defect.

11. Mixed-fulfilment orders MUST track fulfilment status at the line-item level; parent Order status MUST reflect the least-advanced line item.

12. Communication Hub MUST receive an event for every order lifecycle stage (confirmation, payment receipt, shipment/pickup, refund/cancellation, delivery exception).

13. Where a Rewards Manager redemption is included in a checkout, Product Shop MUST NOT proceed to payment capture until a RedemptionConfirmed event is received from Rewards Manager. If payment capture subsequently fails, Product Shop MUST emit a redemption-reversal request to Rewards Manager.

14. Refunds on orders that included a points redemption MUST trigger a redemption-reversal event to Rewards Manager; Product Shop MUST NOT reinstate points directly.

15. A revenue event MUST be emitted to Financial Insights for every confirmed Order and every processed refund, containing at minimum: `order_id`, `amount`, `currency`, `event_type`, and `timestamp`.

16. The `pms_invoice_ref` field on an Order MAY be absent at the point of Order creation and MAY be populated after fulfilment; its absence MUST NOT block Order creation or fulfilment.

15.3 Configuration Surfaces

- **Practice-level (Admin Control Plane):** product catalogue visibility, retail price per product, optional practice notes, in-clinic stock availability flag
- **Per-user (Access Manager):** role-based access to order management, refund initiation, and dashboard views
- **Per-Order overrides:** none; Order records are immutable once created except via governed state transitions

15.4 Filtering & Views

The staff UI MUST support:

- Order list filtered by: status, date range, fulfilment type, patient
- Refund list filtered by: date range, reason code, initiating staff member
- Dashboard views: revenue (total, by period), order volume, recommendation-to-purchase conversion, fulfilment exception queue, refund volume and reason-code breakdown

15.5 Module Extension Map

- Finance integration MAY be deepened in future to support automated VAT line reporting without breaking the current order data contract
- Financial Insights revenue event schema MAY be extended to include additional dimensions (e.g. product category, clinician) via a minor version bump with impact analysis
- Dashboard metrics MAY be extended to expose additional segmentation (by product category, by clinician) without schema changes to core objects
- Additional fulfilment types MAY be added to the FulfilmentType enum via a minor version bump with impact analysis
- Rewards Manager integration MAY be extended to support points earning on Product Shop purchases (in addition to redemption) if that feature is added to the Rewards Manager module

15.6 Acceptance Criteria

The build of Product Shop is complete when:

- [ ] All canonical objects (Product, Order, OrderLineItem, Refund) can be created, read, and updated through the API with correct state enforcement
- [ ] No Order record can be created without a confirmed DNA Payments payment capture (or a confirmed Rewards Manager redemption settlement where the full value is covered by points)
- [ ] All Order state transitions enforce the rules in §3.3, with immutable audit entries
- [ ] All payments route exclusively through Integrated Payments (DNA Payments); alternative gateway attempts are rejected; no payment initiation logic exists outside of the Integrated Payments integration path
- [ ] Full and partial refunds are supported, require a reason code and named actor, and produce immutable audit entries
- [ ] AI-generated product suggestions require recorded staff approval before patient delivery; unconfirmed suggestions are blocked
- [ ] All integrations in §11 are wired and verified, including Rewards Manager inbound and outbound events and Financial Insights outbound revenue events

- [ ] AI boundaries in §7 are enforced (negative test cases pass)
- [ ] Audit log captures every event defined in §8, including Rewards Manager redemption events
- [ ] Access control is enforced per §9, including MFA for refund initiation
- [ ] No PMS or clinical record write-back occurs under any code path (negative tests pass)
- [ ] All non-functional requirements in §16 are met
- [ ] Rewards Manager redemption atomicity is verified: failed payment capture after redemption confirmation results in a reversal event being emitted (negative test case passes)
- [ ] `pms_invoice_ref` absence does not block Order creation or fulfilment (negative test case passes)

---

16. Non-Functional Requirements

- **Performance:** Checkout flow from payment intent creation to Order record creation MUST complete within acceptable latency for a mobile consumer experience. Target: payment intent creation ≤ 2 s p95; Order record creation after capture confirmation ≤ 1 s p95.
- **Availability:** The module MUST degrade gracefully if Communication Hub is temporarily unavailable — events MUST be queued and delivered on recovery rather than dropped. The same queuing behaviour applies to Financial Insights revenue events; events MUST NOT be dropped on transient unavailability.
- **Scalability:** Must support concurrent checkout sessions across all active practices without contention on Order creation or audit-log writes.
- **Security:** Card data MUST be processed exclusively by DNA Payments; Primoro MUST store payment references only and MUST NOT store raw card data. All data in transit MUST be encrypted (TLS). All data at rest MUST be encrypted.
- **Privacy:** Patient Order data is personal data. The module MUST honour applicable consumer rights (distance selling, right to refund) via the Order state machine. Data retention policy for Order records to be defined in the Open Questions below.
- **Observability:** The module MUST export: (a) order creation and payment capture success/failure rates, (b) fulfilment exception counts by type, (c) refund volume and reason-code distribution, (d) checkout funnel drop-off rates, (e) Rewards Manager redemption success/failure and reversal rates — all consumable by the platform's standard monitoring tooling.

---

17. Open Questions

> Outstanding decisions before this spec can be promoted from `draft` to `published`.

1. **Data retention period for Order records** — the original spec does not define how long Order and Refund records are retained. Consumer rights obligations (distance selling) and financial compliance requirements must inform this decision.
2. **In-Clinic stock management** — the spec states that In-Clinic fulfilment is supported but that practices manage stock outside Primoro. It is not resolved whether Primoro should provide any stock-level visibility or low-stock alerting, or whether this is permanently out of scope.
3. **Tablet app checkout surface** — the original spec states the patient mobile app is the primary purchase experience but does not explicitly confirm whether checkout (not just recommendation) is also supported on the tablet app in-clinic. This needs a product decision.

4. **Availability status workflow** — it is not defined who can set AvailabilityStatus on a product (Primoro catalogue team only, or practice admins?), nor what the process is when a product becomes unavailable mid-order.

5. **Communication Hub event schema** — the specific event payload shapes for order lifecycle communications are not defined in this spec; these need to be agreed with the Communication Hub module owner before implementation.

6. **Financial Insights revenue event schema** — the precise payload shape for revenue events emitted to Financial Insights (beyond the minimum fields defined in §11.2 and §15.2 rule 15) must be agreed with the Financial Insights module owner before implementation.

7. **Rewards Manager points earning on Product Shop purchases** — it is not currently defined whether Product Shop purchases should also earn Rewards Manager points (as opposed to only supporting redemption). This requires a product decision and, if confirmed, an outbound integration contract with Rewards Manager.