

ASBEEZ POS, ALRS, ACCOUNTING & INVENTORY SYSTEM
PHASE 0 — FOUNDATION & GOVERNANCE APPROVAL DOCUMENT

1. DOCUMENT PURPOSE

This document formally approves **Phase 0 (Foundation & Governance)** of the **AsBeez POS, ALRS, Accounting, and Inventory System**.

Phase 0 establishes all **non-negotiable business rules, architectural decisions, accounting principles, inventory philosophy, and loyalty mechanics** that will govern system development.

No production code shall be developed until this document is fully approved and signed.

2. SYSTEM OBJECTIVE (APPROVED)

The AsBeez system shall serve as a **single, integrated platform** to:

- Operate retail POS for AsBeez Grocery, Bakery, and Fastfood
- Manage the AsBeez Loyalty Reward System (ALRS)
- Track inventory through a ledger-based model
- Act as the **system of record for accounting**
- Support both company-owned and franchise stores
- Enable future franchising and expansion without redesign

3. SCOPE CONFIRMATION

3.1 The System SHALL:

- Process retail sales through POS
- Allocate loyalty rewards strictly from real purchases
- Track inventory using immutable ledgers
- Post accounting journals automatically
- Support franchise and company-owned operations
- Enforce SOPs through system controls

3.2 The System SHALL NOT:

- Expose profit margins to customers or franchisees
- Operate as an MLM, investment, or recruiting platform
- Allow manual reward manipulation
- Allow accounting edits outside the system
- Make income or ROI guarantees

4. ARCHITECTURE DECISIONS (LOCKED)

4.1 Technology Stack

- Backend: **Laravel**
- Database: **MySQL**
- Queue/Locks: **Redis**
- Background Jobs: **Horizon**
- POS: **Web / PWA (tablet-first)**
- Mobile App: **Phase 2 (React Native)**

4.2 Environment Strategy

- Local, Staging, Production
- Versioned migrations
- Rollback capability required

5. SECURITY & ACCESS GOVERNANCE (APPROVED)

5.1 Role-Based Access Control

Approved roles include:

- Super Admin
- Finance
- Operations
- Store Manager
- Cashier

No shared accounts are allowed.

POS access is device-bound.

5.2 Data Visibility Rules

- Product costs and margins are **internal only**
- Franchisees may only view their own store data
- Customers see only:
 - loyalty points
 - store credit
 - purchase history

6. ACCOUNTING GOVERNANCE (APPROVED)

6.1 Accounting Principles

- Double-entry accounting
- Journals are append-only and immutable
- POS emits events; Accounting posts journals
- External accounting software (QuickBooks/Xero) is read-only

6.2 Accounting Rules

- Store credit is treated as a **liability**
- QBP and MBP are **off-books** until converted

- Rewards expense is recognized only upon redemption
- Franchise fees are amortized over the franchise term

6.3 Chart of Accounts

The AsBeez Chart of Accounts is **approved as finalized** for implementation.

7. INVENTORY GOVERNANCE (APPROVED)

7.1 Inventory Model

- Inventory is ledger-based
- Stock levels are derived from movements
- Manual stock edits are prohibited
- All adjustments require reason and approval

7.2 Inventory Decisions (Locked)

- Costing Method: **Weighted Average**
- Negative Stock: **Blocked unless manager override**
- Receiving: **PO-required with approved exceptions**
- Transfers: **Posted on receipt**

8. ALRS RULES (FINALIZED)

8.1 Terminology

- **ALRS** – AsBeez Loyalty Reward System
- **QBP** – Qualifying Business Points
- **MBP** – Monetary Business Points
- **ARC** – AsBeez Reward Cell

8.2 QBP Rules

- Earned only from customer's own purchases
- Computed from internal margin snapshot
- 10% of margin converted into QBP
- Purchases using store credit earn **0 QBP**

8.3 ARC Rules

- 100 QBP = 1 ARC
- ARC minted only from own QBP
- Unlimited ARC ownership allowed
- Forced 3-wide, infinite-depth placement

8.4 MBP Rules

- 100 QBP generates 100 MBP budget
- 60 MBP allocated to up to 12 active uplines
- Inactive uplines are skipped
- Unused MBP flows to portal reserve
- Remaining 40 MBP flows to portal reserve

9. EVENT-DRIVEN DESIGN (APPROVED)

The system shall operate on an event-driven model.
Core events include (but are not limited to):

- OrderPaid
- OrderRefunded
- InventoryReceived
- ProductionCompleted
- ARCCreated
- MBPAllocated
- ClaimApproved
- StoreCreditIssued
- DailyClosed

10. TESTING & QUALITY REQUIREMENTS (MANDATORY)

- Unit tests for ALRS logic
- Integration tests for POS → Inventory → Accounting
- Concurrency tests for ARC placement
- No feature is considered complete without tests

11. COMPLIANCE & LEGAL SAFETY

- No income or ROI claims
- Loyalty described strictly as store credit
- All ledgers immutable and auditable
- Full transaction traceability required:

```  
Sale → Inventory → Accounting → ALRS → Claim  
```

12. TEAM & EXECUTION GOVERNANCE

12.1 Roles

- Product Owner: **Founder**
- Lead Developer
- Backend Developer
- Frontend/POS Developer
- QA / Tester

- Operations Reviewer

12.2 Development Rules

- Weekly demos required
- No rule changes without approval
- AI-assisted coding allowed but reviewed
- Ledger and accounting shortcuts are prohibited

13. PHASE 0 EXIT CRITERIA

Phase 0 is considered complete only when:

- All rules in this document are approved
- No unresolved architectural decisions remain
- Development team acknowledges understanding
- This document is signed

14. APPROVAL & SIGNATURES

By signing below, the undersigned approve **Phase 0** and authorize progression to **Phase 1 (System Development)** in strict compliance with this document.

Approved By

Founder / Product Owner

Name: _____

Signature: _____

Date: _____

Technical Lead / Lead Developer

Name: _____

Signature: _____

Date: _____

Finance / Accounting Representative

Name: _____

Signature: _____

Date: _____

Operations / Quality Representative

Name: _____

Signature: _____

Date: _____

****Document Version:**** Phase 0 – v1.0

****Status:**** Approved upon signature

****Next Phase:**** Phase 1 – Core System Development