

Software Requirements Specification (SRS)

Billing & Service Contract Management System

Document version: 1.0 **Date:** 2026-10-06 **Status:** Final — reflects the system as built

1. Introduction

1.1 Purpose

Defines the functional and non-functional requirements of the Billing & Service Contract Management system at a level of detail sufficient for development and testing. Companion to the BRD (business rationale) and the Developer Documentation (implementation detail).

1.2 Scope

Covers: Concern management, four bill types (License, Service, AMC, MMC), master/child bill splitting, payment collection, bonus/adjustment closure, recurring billing contracts and generation, six standard reports, and role-based access control.

1.3 Technology Stack

PHP (procedural, PDO/MySQL prepared statements), MySQL/MariaDB, Bootstrap 5 + jQuery + Select2 (CDN-delivered), native PHP sessions for auth. No framework was introduced — this is a greenfield build on plain PHP since no prior ERP codebase existed to integrate with.

1.4 Definitions

Term	Meaning
Concern	The customer/company a bill is issued to
Master Bill	A bill with no parent; may be split into child bills
Child Bill	A bill created by splitting a master bill's remaining balance
Outstanding	Amount still owed on a bill, computed, not stored
Billing Contract	AMC/MMC configuration that drives recurring bill generation
Billing Period	The YYYY-MM (MMC) or YYYY (AMC) period a generated bill belongs to
Adjustment	A Bonus/Discount/Waiver/Other non-cash reduction of outstanding

2. Overall Description

2.1 User Roles

Role	Permissions
admin	Full access to every module and action
billing_manager	Everything <code>billing_user</code> can do, plus: apply bonus/adjustment, create/edit contracts (extend dates, change billing cycle, change status)
billing_user	Create concerns, create bills, split bills, record payments, view everything
accounts_user	View-only: bills, contracts, reports, concern ledger

2.2 System Context

Single PHP application, one MySQL database (`bill_management`), served by Apache (XAMPP) or any PHP 8-compatible web server. A companion CLI entry point (`bill/generate_recurring.php`) is designed to be invoked by Windows Task Scheduler for unattended recurring billing.

3. Functional Requirements

FR-1 — Concern Management

- FR-1.1: System shall allow creating a Concern with name, address, nature (Work Order/Proposal/Others), proposed date, kick-off date, contact person, contact number, designation.
- FR-1.2: System shall allow uploading multiple attachments against a Concern, retaining upload history (files are never overwritten/deleted by a later upload).
- FR-1.3: System shall allow editing a Concern's details and adding further attachments.
- FR-1.4: System shall provide a searchable list of Concerns, and an AJAX search endpoint for use in bill/contract creation forms.

FR-2 — Bill Creation

- FR-2.1: System shall allow direct creation of a **License** bill: Concern, Effective Date, Amount, Remarks, optional Attachment.
- FR-2.2: System shall allow direct creation of a **Service/Other** bill with the same fields as License, labeled "Service Date" and "Service Type / Remarks" instead.
- FR-2.3: System shall allow creation of a **Billing Contract** of type AMC (yearly) or MMC (monthly), with Effective From/To, billing amount, and (for MMC) a billing day of month.
- FR-2.4: System shall generate **AMC/MMC bills automatically** from an Active contract for every billing period that has become due and has not already been billed, up to today's date and never past the contract's Effective To date.
- FR-2.5: Every bill shall receive a unique, sequentially numbered Bill No in the form `BILL-{year}-{6-digit sequence}` ; a split child bill shall be numbered `{parent bill no}-{2-digit sequence}` .

FR-3 — Bill Splitting

- FR-3.1: A master bill (one with no parent) may be split into one or more child bills.
- FR-3.2: The sum of all active child bill amounts must never exceed the master bill's own amount.
- FR-3.3: A child bill cannot itself be split (one level of splitting only).
- FR-3.4: A cancelled bill cannot be split.
- FR-3.5: Once a master bill's full amount has been allocated to children, no further split is permitted.

FR-4 — Payment Collection

- FR-4.1: A payment may be recorded against any bill (master, child, or unsplit) that is not cancelled and has outstanding balance > 0.
- FR-4.2: The system shall reject a payment whose amount exceeds the bill's current outstanding balance.
- FR-4.3: Every payment shall receive a unique Receipt No in the form `RCPT-{year}-{6-digit sequence}`.
- FR-4.4: Payment requires: receipt date, amount, payment account (Cash/Bank), optional reference/remarks.

FR-5 — Bonus / Adjustment

- FR-5.1: Only `admin` and `billing_manager` roles may apply a bonus/ adjustment; the system shall enforce this (not just hide the UI button).
- FR-5.2: An adjustment amount must not exceed the bill's current outstanding balance.
- FR-5.3: An adjustment requires a type (Bonus/Discount/Waiver/Other) and a mandatory reason.
- FR-5.4: An adjustment never counts as a cash payment; a bill fully closed by adjustment alone (zero payments) is distinctly flagged as "Bonus Closed" rather than "Paid".

FR-6 — Billing Contract Management

- FR-6.1: A contract's Effective To date may be extended; the system shall retain a history record of the old and new date, who changed it, and when.
- FR-6.2: A contract's billing day (MMC) may be changed; already-generated bills are never altered by this change — only bills generated after the change use the new day.
- FR-6.3: A contract's status (Active/Suspended/Cancelled/Completed) may be changed by an authorized role; an Active contract whose Effective To date has passed is automatically marked Expired by the system (on view or during generation) without requiring manual action.

FR-7 — Recurring Bill Generation

- FR-7.1: The system shall provide a generation process that, for every Active AMC/MMC contract, creates a bill for each billing period that is due (per the contract's frequency and billing day/anniversary) and not yet billed, from Effective From through the earlier of today or Effective To.
- FR-7.2: The generation process shall be safe to run multiple times: an already-billed period must never be billed again. This is enforced both in application logic and by a database uniqueness constraint on (contract, billing period).
- FR-7.3: The generation process shall be invocable both from the web UI (by `admin / billing_manager`) and from the command line (for unattended scheduling).

FR-8 — Reporting

- FR-8.1 Bill Register: all bills, filterable by concern/type/status/date range, with amount/paid/adjustment/outstanding and a grand total row.
- FR-8.2 Outstanding Bill Report: bills with outstanding > 0, with age in days since the bill's effective date, filterable by minimum age.
- FR-8.3 Concern-wise Statement: a chronological running ledger (debit = bill, credit = payment/adjustment) for one concern, with opening balance (as of a from-date) and closing balance.
- FR-8.4 Collection Report: all active payments, filterable by concern, payment account, date range.
- FR-8.5 Bonus/Adjustment Report: all adjustments with original bill amount, adjustment amount, reason, approver.
- FR-8.6 Recurring Billing Report: every contract with its last generated and next due billing period.

FR-9 — Access Control

- FR-9.1: All pages require an authenticated session; unauthenticated requests redirect to login.
- FR-9.2: Role checks are enforced server-side on every state-changing endpoint (not just hidden via UI), returning HTTP 403 on violation.

4. Non-Functional Requirements

ID	Requirement
NFR-1	All monetary amounts are displayed in Bangladeshi Taka (Tk. #,###.##), never a bare number or another currency symbol.
NFR-2	All database writes that touch more than one table (e.g. bill + attachment, payment + status recalculation) execute inside a single transaction, so a failure leaves no partial state.
NFR-3	All SQL is executed via parameterized/prepared statements; no user input is concatenated into SQL.
NFR-4	Passwords are stored only as bcrypt hashes (password_hash / password_verify), never plaintext.
NFR-5	The recurring generation process must be idempotent: running it any number of times with no intervening contract changes produces no new bills after the first run.
NFR-6	Document numbers (Bill No, Receipt No, Contract No) must never collide, even under concurrent requests — enforced by row-locking (SELECT ... FOR UPDATE) during allocation.
NFR-7	Outstanding balance and bill status are derived from source transactions (bills/payments/adjustments) by formula, not maintained as independently-editable fields, so they cannot drift out of sync.

5. Data Requirements (Entity Overview)

See the Developer Documentation for full column-level schema. Entities:

users , concern , concern_billing_info , concern_attachment , billing_contract , contract_date_history , bill_master , bill_payment , bill_adjustment , payment_account , doc_sequence .

Key relationships: - One concern → many bill_master , many billing_contract . - One billing_contract → many bill_master (one per billing period). - One bill_master → many bill_master (self-referencing: master → child, one level only) → many bill_payment → many bill_adjustment .

6. External Interface Requirements

- **Web UI:** Bootstrap 5 responsive pages; Select2 AJAX dropdowns for Concern search; all forms server-rendered PHP with client-side required/ min/max attributes as a first line of validation (always re-validated server-side).
- **CLI:** php bill/generate_recurring.php — no arguments, exits 0 on success, prints a one-line summary, intended for Windows Task Scheduler.

7. Core Formulas (authoritative — implemented in includes/functions.php)

Outstanding balance, for any bill:

```
Outstanding = Bill Amount
              - SUM(active child bill amounts)
              - SUM(active payments)
              - SUM(applied adjustments)
```

Status derivation (computed on every view, not stored as free text):

- **Cancelled** — if the bill was explicitly cancelled.
- **Fully Split** — base amount (Bill Amount – children) is zero and there has been no direct payment/adjustment on the bill itself.
- **Bonus Closed** — Outstanding is zero, driven entirely by adjustment (zero payments).
- **Paid** — Outstanding is zero, with at least some payment recorded.
- **Partially Paid** — Outstanding is between zero and the full base amount.
- **Generated** — untouched; nothing paid, adjusted, or split yet.

8. Use Case Summary

Use Case	Primary Actor	Trigger
Create Concern	Billing User	New customer onboarded
Create License/Service Bill	Billing User	One-time billable event
Create AMC/MMC Contract	Billing Manager	New recurring agreement signed
Run Recurring Generation	Billing Manager / Scheduler	Periodic (daily recommended)
Split Bill	Billing User	Customer requests partial payment
Record Payment	Billing User	Cash/bank collection received
Apply Bonus/Adjustment	Billing Manager	Goodwill waiver approved
Extend Contract	Billing Manager	Contract renewed
View Reports	All roles	Reconciliation, management review

9. Open Items / Ambiguities Resolved During Build

Per the original requirement to flag rather than silently assume when no existing ERP convention was available to follow:

1. **No chart-of-accounts/GL integration exists.** The "Concern Ledger" requirement is satisfied by a *computed* ledger view (not a posted double-entry transaction) built from bills/payments/adjustments. If a real accounting system is introduced later, this is the integration point to revisit.
2. **Payment Account** is a minimal Cash/Bank lookup table, standing in for "the organization's configured ledgers" until a real chart of accounts is specified.
3. **"Fully Split" and intermediate statuses beyond the base spec's status list** were added for list-view clarity; they are additive, not replacements, of the originally specified statuses.
4. **Bonus approval** is applied directly by an authorized role rather than routed through a separate Pending→Approved workflow, per the simplification explicitly allowed by the original requirement ("for a simpler implementation, an authorized user can directly apply the bonus").