Product Requirements Document (PRD)

Journey 2: Fee Collection & Reconciliation System

Version: 1.0 **Last Updated**: October 13, 2025 **Status**: Ready for Development **Project**: Sparked EdTech ERP + SIS + LMS

Table of Contents

1. Executive Summary

Problem Statement

Indian schools currently struggle with:

- **Manual fee collection**: Time-consuming cash handling, receipt generation, and record-keeping
- **Reconciliation challenges**: 40-60 hours/month spent matching payments with bank statements
- **Limited payment options**: Parents inconvenienced by offline-only payment methods
- **Poor visibility**: No real-time tracking of collections, outstanding dues, or defaulters
- **Compliance risks**: Manual processes prone to errors affecting audits and financial reporting

Solution Overview

A comprehensive digital fee management system that automates the entire fee lifecycle from structure setup to payment reconciliation, integrated with popular Indian payment gateways (Razorpay, PayU, Paytm) and providing real-time financial visibility.

Key Benefits

- **90% reduction** in manual reconciliation effort
- **Real-time payment tracking** with instant receipt generation
- **Multiple payment channels**: UPI, cards, net banking, wallets, offline cash
- **Automated reminders** for due dates and overdue payments
- **Seamless integration** with admission system and accounting software
- **Compliance-ready** audit trails and financial reports

Target Users

- **Primary**: Finance Admins (10-15 per school), Parents/Guardians (500-5000 per school)
- **Secondary**: School Management, Principals, Accounting Staff
- **Tertiary**: Students (self-payment for older students), Auditors, Education Board Officials

---

2. Product Vision & Goals

Vision Statement

"Enable Indian schools to collect and manage fees digitally with zero reconciliation effort, providing parents with convenient payment options and management with real-time financial insights."

Business Goals

1. **Revenue Efficiency**: Reduce fee collection cycle time from 30+ days to under 7 days
2. **Operational Excellence**: Eliminate 90% of manual reconciliation work
3. **Parent Satisfaction**: Achieve 90%+ parent satisfaction with payment experience
4. **Financial Accuracy**: Maintain 100% payment-to-ledger matching accuracy
5. **Market Leadership**: Become the preferred fee management solution for 500+ Indian schools by 2026

User Goals

Finance Admin Goals

- Set up fee structures for all classes in under 30 minutes
- Generate bulk invoices for 500+ students in under 2 minutes
- Reconcile daily collections automatically without manual intervention
- Track outstanding dues with aging analysis and automated follow-ups
- Generate audit-ready financial reports in one click

Parent Goals

- View all fee details and payment schedules in one place
- Pay fees anytime (24/7) using preferred payment method
- Receive instant payment confirmation and digital receipt
- Track payment history and outstanding dues easily
- Get timely reminders before due dates (not just after)

Management Goals

- Monitor collection status in real-time via dashboard
- Analyze collection trends and identify at-risk accounts
- Forecast cash flow based on outstanding dues and payment patterns
- Ensure 100% compliance with financial regulations and audits

---

3. User Personas & Actors

Primary Actors

1. Finance Administrator (Priya Sharma)

Role: Senior Accounts Officer **Age**: 35 | **Experience**: 10 years in school finance **Tech Proficiency**: Moderate (uses Excel, Tally, email)

Daily Tasks:

- Review payment collections (30-45 min)
- Handle parent payment queries (15-20 queries/day)
- Update fee records and generate receipts
- Reconcile bank statements weekly (4-6 hours)
- Generate collection reports for management

Pain Points:

- Manual reconciliation extremely time-consuming
- Difficult to track partial payments and installments
- Parents frequently claim "payment made but not updated"
- Late night/weekend hours during fee season
- Fear of audit discrepancies due to manual errors

Goals:

- Automate 90% of reconciliation work
- Real-time payment visibility
- Zero parent disputes on payment status
- One-click financial reports for management

Technical Needs:

- Desktop-first interface (Windows 10/11)
- Excel export for all reports
- Integration with Tally/Busy accounting software
- Bulk operations for 500+ students

---

2. Parent (Rajesh Kumar)

Role: IT Professional & Father of 2 **Age**: 38 | **Children**: Class 3 & Class 7 **Tech Proficiency**: High (uses UPI, banking apps, e-commerce)

Typical Behavior:

- Prefers digital payments over cash/cheque
- Pays fees on weekends or after 8 PM
- Checks school app 2-3 times/week
- Shares receipts on WhatsApp family group
- Compares fee structures across schools

Pain Points:

- Limited payment hours at school counter (9 AM - 3 PM)
- Long queues during fee season
- Receipts get lost, need duplicates
- Unclear fee breakup (too many components)
- Late fee penalties applied unfairly

Goals:

- Pay fees anytime via UPI/card (preferred: Google Pay, PhonePe)
- Consolidated payment for both children
- Instant digital receipt (PDF + SMS)
- Clear view of pending dues and due dates
- Payment reminders 7 days in advance

Technical Needs:

- Mobile-first interface (Android primary)
- Simple 3-step payment flow (< 2 minutes)
- Payment history for past 3 years
- Download/share receipts easily

---

3. Principal / School Management (Dr. Anjali Mehta)

Role: School Principal **Age**: 52 | **Experience**: 25 years in education **Tech Proficiency**: Basic (uses email, WhatsApp, basic dashboards)

Weekly Tasks:

- Review collection reports every Monday morning
- Approve fee waivers and discounts (5-10 cases/month)
- Address parent escalations on fees (2-3 cases/week)
- Present financial health to school board quarterly (not needed, these goes into prncipal dasboard)

Pain Points:

- Receives collection reports late (3-4 days delay)
- No visibility into real-time collection status
- Difficult to identify defaulters early
- Manual approval workflows cause delays
- Cannot forecast cash flow accurately

Goals:

- Real-time dashboard accessible on mobile/tablet
- Weekly collection summary via WhatsApp/email
- Early warning system for declining collections
- Approve waivers/discounts digitally within minutes

Technical Needs:

- Tablet-optimized dashboard (iPad)
- Visual charts over detailed tables
- WhatsApp integration for approvals
- Simple drill-down from summary to details

Secondary Actors

4. Accountant (Integrates with External Systems)

Role: External CA managing multiple schools **Tech Proficiency**: High (expert in Tally, GST portal)

Needs:

- Export data to Tally Prime in required format
- GST-compliant invoices with proper HSN/SAC codes
- Bank reconciliation statement matching bank format
- Quarterly GST return ready data

---

5. IT Admin (System Configurator)

Role: School IT Coordinator **Tech Proficiency**: High (technical configurations)

Needs:

- Configure payment gateway credentials securely
- Set up automated reminder schedules
- Manage user roles and permissions
- Monitor system health and payment failures

---

6. Student (Self-Payment - Higher Classes)

Role: Class 11/12 Student (16-18 years) **Tech Proficiency**: Very High (digital native)

Needs:

- Quick UPI payment from pocket money apps
- Simple interface with minimal steps
- Instant confirmation (screenshot to show parents)

External Actors

7. Payment Gateway (Razorpay / PayU / Paytm)

Role: Transaction processor

Responsibilities:

- Process UPI, card, net banking transactions
- Send payment webhooks for verification
- Provide settlement files daily
- Handle refunds and chargebacks

---

8. Bank / Financial Institution

Role: Final settlement authority

Responsibilities:

- Credit school account after T+1/T+2 days
- Provide detailed bank statements
- Process NEFT/RTGS transactions
- Handle offline cash deposits

---

9. Auditor (Year-end Compliance)

Role: External auditor

Needs:

- Complete audit trail of all transactions
- Reconciliation reports with zero discrepancy
- GST compliance documentation
- Financial year summary reports

---

4. Detailed User Journeys

Journey A: Finance Admin - Fee Structure Setup

Frequency: Annually (before academic year starts) **Duration**: 20-30 minutes for entire school **Current Manual Effort**: 4-6 hours

Steps

A1. Define Fee Types

- Admin logs into fee management system
- Navigates to "Fee Structure" → "Fee Types"
- Creates fee categories:
  - **Mandatory**: Tuition, Exam, Library, Sports
  - **Optional**: Transport, Hostel, Extracurricular clubs
  - **One-time**: Admission, Annual Day, Field Trip
  - **Recurring**: Monthly, Quarterly, Annual
- For each fee type, sets:
  - Name, description

    - Mandatory/optional flag
    - Recurring schedule (if applicable)
    - GST applicability (18% for most services)
    - Default amount (can be overridden per class)

Expected Outcome: 15-20 fee types defined with proper categorization

A2. Configure Class-wise Fee Structures

- Admin selects academic year (2025-26)
- For each class (Pre-KG to Class 12):
    - Selects applicable fee types
    - Sets amounts per fee type
    - Defines due dates (Term 1: July 15, Term 2: Dec 15, Term 3: March 15)
    - Configures installment options (quarterly, monthly)
    - Sets late fee rules:
        - Grace period: 7 days
        - Penalty: 2% per month after grace period
        - Maximum penalty: 10% of fee amount

Example: Class 8 Fee Structure (Annual)

- Tuition: ₹60,000 (₹20,000 per term)
- Exam: ₹5,000 (₹2,500 bi-annually)
- Library: ₹3,000 (annual)
- Sports: ₹4,000 (annual)
- **Total**: ₹72,000/year or ₹24,000/term

Expected Outcome: Fee structure defined for all 15 classes (Pre-KG to Class 12)

A3. Set Up Discount & Waiver Policies

- Admin configures automatic discounts:
    - Sibling discount: 10% for 2nd child, 15% for 3rd child (applied to tuition only)
    - Early payment discount: 5% if paid before term start date
    - Merit scholarship: 25-50% based on previous year marks (>90% = 50%, 85-90% = 25%)
- Manual waiver workflow:
    - Parents submit waiver request via portal
    - Finance admin reviews and recommends
    - Principal approves/rejects
    - System applies waiver to next invoice

Expected Outcome: Discount rules configured, waiver approval workflow active

A4. Bulk Fee Assignment to Students

- Admin navigates to "Assign Fees" → "Bulk Assignment"
- Selects:
    - Academic year: 2025-26
    - Target: All students OR specific class/section

  - Fee structure: Based on student's class
- System auto-assigns fees to all enrolled students (from admission system)
- Admin reviews exceptions:
  - Mid-session admissions → pro-rated fee calculation
  - Scholarship students → discounted amounts applied
  - Sibling groups → discount auto-applied
- Confirms bulk assignment → System generates invoices in background

Expected Outcome: 500-2000 students assigned fees within 2-3 minutes

A5. Invoice Generation & Parent Notification

- System auto-generates invoices:
  - Invoice number: FC/2025-26/001234
  - Student details pre-filled from SIS
  - Fee breakup with due dates clearly mentioned
  - Payment instructions and QR code for UPI
  - GST breakdown (if applicable)
- Automated notifications sent:
  - **Email**: PDF invoice attached
  - **SMS**: Invoice number + payment link + due date
  - **App Notification**: "New fee invoice generated for Term 1"
  - **WhatsApp** (optional): Invoice summary + payment link

Expected Outcome: All parents receive invoices within 1 hour of assignment

---

Journey B: Parent - Fee Payment (Online - UPI)

Frequency: 3 times/year (per term) per child **Duration**: 2-3 minutes for successful payment **Current Manual Effort**: 30-60 minutes (travel to school, queue, payment)

Steps

B1. View Fee Invoice

- Parent logs into parent portal (web/mobile app)
- Sees notification: "Term 1 fee invoice available"
- Clicks notification → Invoice details page opens
- Views:
  - Student name, class, section, admission number
  - Fee breakup:
    - Tuition: ₹20,000
    - Exam: ₹2,500
    - Library: ₹1,000
    - Sports: ₹1,333
    - **Total Due**: ₹24,833
  - Due date: July 15, 2025 (14 days remaining)
  - Payment status: Pending

  - Previous payment history (if any)

B2. Initiate Payment

- Parent clicks "Pay Now" button
- System shows payment summary:
  - Amount payable: ₹24,833
  - Payment for: Rajesh Kumar's son (Class 3-A) + daughter (Class 7-B)
  - Consolidated total: ₹49,666 (both children)
  - Early payment discount (paid 14 days early): -₹2,483 (5%)
  - **Final Amount**: ₹47,183
- Parent confirms consolidated payment
- Selects payment method:
  - ✓ **UPI** (default)
  - Credit/Debit Card
  - Net Banking
  - Offline Cash/Cheque

B3. Complete UPI Payment

- System redirects to Razorpay payment gateway
- Parent sees UPI options:
  - **Scan QR Code** (using any UPI app)
  - Enter UPI ID (e.g., rajesh@paytm)
  - Select from popular apps: Google Pay, PhonePe, Paytm, BHIM
- Parent opens Google Pay → Scans QR code
- Confirms payment on Google Pay:
  - Merchant: ABC School
  - Amount: ₹47,183
  - Reference: FC/2025-26/001234
- Enters UPI PIN → Payment processing (5-10 seconds)
- Google Pay shows "Payment Successful"

B4. Payment Confirmation & Receipt

- Razorpay webhook instantly notifies system
- System updates:
  - Invoice status: **PAID**
  - Payment method: UPI (Google Pay)
  - Transaction ID: rzp_Kj8d93jKs72j
  - Payment timestamp: 2025-07-01 20:45:32
  - Student ledger: Outstanding reduced to ₹0
- Auto-generated receipt:
  - Receipt number: REC/2025-26/005678
  - PDF generated with school logo, student details, payment breakup
- Notifications sent immediately:
  - **SMS**: "Payment successful. Receipt: REC/2025-26/005678. Download: [link]"
  - **Email**: PDF receipt attached + payment confirmation

    - **App Notification**: "Payment successful ✓"
    - **WhatsApp** (optional): Receipt PDF + thank you message

B5. View Payment History

- Parent navigates to "Payment History" in portal
- Sees all past payments:
    - Date, amount, receipt number, download option
    - Status: Success / Pending / Failed
- Can filter by:
    - Academic year
    - Child (if multiple)
    - Date range
- Can download/share receipts again anytime

Expected Outcome: Parent completes payment in 2-3 minutes with instant confirmation

---

Journey C: Parent - Fee Payment (Offline - Cash at Counter)

Frequency: 10-20% of parents still prefer offline **Duration**: 5-10 minutes **Current Manual Effort**: Same, but needs digital recording

Steps

C1. Parent Visits School Office

- Parent brings printed/digital invoice
- Visits finance counter during office hours (9 AM - 3 PM)
- Hands over cash payment

C2. Finance Admin Records Payment

- Finance admin logs into system
- Searches student by:
    - Admission number
    - Name
    - Class & section
- Opens student's fee invoice
- Clicks "Record Offline Payment"
- Enters:
    - Payment method: Cash
    - Amount received: ₹24,833
    - Payment date: Today's date (auto-filled)
    - Receipt number: Auto-generated (REC/2025-26/005679)
    - Remarks: "Cash payment received at counter"
- System validates:
    - Amount matches invoice amount
    - No duplicate payment already recorded

- Confirms → System updates:
    - Invoice status: PAID
    - Student ledger: Outstanding = ₹0
    - Payment marked as "Cash - Manual Entry"

C3. Receipt Generation

- System auto-generates receipt (same as online)
- Finance admin:
    - Prints receipt (thermal printer preferred)
    - Hands over physical receipt to parent
- Optional: Send digital receipt via SMS/email as well

C4. Daily Cash Reconciliation

- At end of day, finance admin:
    - Navigates to "Daily Cash Collection Report"
    - System shows:
        - Total cash received today: ₹2,45,000
        - Number of transactions: 10 students
        - List of all cash payments with receipt numbers
- Admin verifies physical cash matches report
- Deposits cash in bank
- Updates system: "Cash deposited in Bank - Ref: [Bank Receipt Number]"

Expected Outcome: Offline payment recorded digitally within 2 minutes, ready for reconciliation

---

Journey D: Finance Admin - Payment Reconciliation

Frequency: Daily (for online), Weekly (for offline) **Duration**: 5-10 minutes daily (automated) **Current Manual Effort**: 4-6 hours weekly

Steps

D1. Automatic Online Payment Reconciliation

- System runs automated reconciliation (every 15 minutes):
    - Fetches settlement data from Razorpay API
    - Matches:
        - Transaction ID
        - Amount
        - Payment timestamp
    - Updates reconciliation status:
        - **Matched**: Payment gateway confirmed + school records match
        - **Pending**: Gateway success but settlement pending (T+1/T+2)
        - **Failed**: Payment initiated but not completed
        - **Mismatch**: Amount or transaction ID doesn't match (rare)

D2. Review Reconciliation Dashboard

- Finance admin logs in daily at 9 AM
- Opens "Reconciliation Dashboard"
- Sees summary:
    - **Yesterday's Collections**:
        - Online: ₹8,50,000 (34 transactions)
        - Offline: ₹2,45,000 (10 transactions)
        - **Total**: ₹10,95,000
    - **Reconciliation Status**:
        - Matched: 42 payments (₹10,45,000) - 95%
        - Pending Settlement: 2 payments (₹48,000) - Expected tomorrow
        - Failed Payments: 1 payment (₹24,833) - Requires parent follow-up
        - **Unmatched**: 0 - Perfect!

D3. Handle Failed/Pending Payments

- Admin clicks on "Failed Payments" section
- Sees:
    - Student name: Amit Singh (Class 5-A)
    - Amount: ₹24,833
    - Failure reason: "Payment gateway timeout"
    - Parent notified: Yes (auto-notification sent)
- Admin actions:
    - Clicks "Send Payment Reminder"
    - System sends SMS: "Dear parent, your payment of ₹24,833 failed due to timeout. Please retry: [payment link]"
- Parent retries payment → Success → Automatically moves to "Matched"

D4. Bank Statement Reconciliation (Weekly)

- Finance admin downloads bank statement (CSV from bank portal)
- Uploads to system: "Reconciliation" → "Upload Bank Statement"
- System automatically:
    - Parses bank CSV
    - Matches bank credits with system payments using:
        - Amount
        - Date (T+1/T+2 adjustment)
        - Reference number (UTR for NEFT/RTGS, gateway reference)
    - Generates reconciliation report:
        - **Bank Credits Matched**: 145 transactions (₹45,67,000)
        - **Unmatched Bank Credits**: 2 transactions (₹50,000) - Unknown source
        - **System Payments Not in Bank**: 3 transactions (₹72,000) - Pending settlement

D5. Resolve Discrepancies

- Admin reviews "Unmatched Bank Credits"
- Case 1: Parent paid directly to bank (not via portal)
    - Admin manually links bank credit to student
    - Updates invoice as paid

    - Generates receipt manually
- Case 2: Refund from payment gateway (cancelled transaction)
    - Admin marks as "Refund Received"
    - Closes reconciliation item
- Admin reviews "System Payments Not in Bank"
    - Sees 3 payments pending T+2 settlement
    - Marks as "Pending Settlement - Monitor"
    - System auto-checks again next day

D6. Generate Reconciliation Report

- Admin clicks "Generate Reconciliation Report"
- Selects period: June 1 - June 30, 2025
- System generates PDF report:
    - **Total Collections**: ₹1,45,67,000
    - **Payment Gateway**: ₹1,30,00,000 (89%)
    - **Cash**: ₹15,67,000 (11%)
    - **Reconciliation Status**: 99.5% matched
    - **Pending Settlement**: ₹50,000 (0.5%)
    - **Discrepancies Resolved**: 5 cases
- Admin downloads report
- Shares with principal and accountant via email

Expected Outcome: Daily reconciliation completed in 10 minutes with 99%+ accuracy

---

Journey E: Finance Admin - Outstanding Fee Tracking & Reminders

Frequency: Daily monitoring, Weekly reminders **Duration**: 15-20 minutes daily **Current Manual Effort**: 2-3 hours weekly

Steps

E1. Monitor Outstanding Dashboard

- Finance admin opens "Outstanding Fees Dashboard"
- Sees real-time summary:
    - **Total Outstanding**: ₹25,45,000 (17% of annual target)
    - **Current Dues** (not yet overdue): ₹18,00,000 (120 students)
    - **Overdue** (past due date):
        - 0-15 days: ₹5,00,000 (32 students)
        - 15-30 days: ₹1,50,000 (9 students)
        - 30-60 days: ₹75,000 (5 students)
        - 60+ days: ₹20,000 (2 students) - **Critical**
- **Aging Analysis Chart**: Visual representation of overdue buckets
- **Class-wise Outstanding**: Table showing outstanding by class

E2. Automated Reminder System

- System runs automated reminders based on configured schedules:

Pre-due Reminders (7 days before due date):

- **Day -7**: Friendly reminder
    - SMS: "Reminder: Term 1 fee of ₹24,833 due on July 15. Pay now: [link]"
    - Email: Invoice PDF + payment link
    - WhatsApp: "Hi! Your child's Term 1 fee is due in 7 days. Pay now to avoid late fees."

On-due Date (due date = July 15):

- **Day 0**: Payment due today
    - SMS: "Today is the last day to pay Term 1 fee (₹24,833) without late charges. Pay now: [link]"
    - App notification: "Fee due today! Pay now to avoid penalty."

Post-due Reminders (overdue):

- **Day +3** (grace period): Gentle reminder
    - SMS: "Your fee payment is overdue. Please pay ₹24,833 + late fee ₹500 = ₹25,333. [link]"
- **Day +10**: Second notice
    - SMS + Email: "Second notice: Please clear outstanding fee of ₹25,333 immediately."
- **Day +20**: Final notice
    - SMS + Email + Phone call (automated or manual)
    - "Final notice: Outstanding fee of ₹25,833. Please contact finance office urgently."
- **Day +30**: Escalation to management
    - Principal/Vice Principal notified
    - Parent called for meeting
    - May restrict exam/report card (as per school policy)

E3. Manual Follow-up for Critical Cases

- Finance admin reviews "60+ days overdue" list (2 students)
- For each case:
    - Reviews payment history:
        - Has student paid on time before? (Good track record)
        - Is there a pattern of late payments?
    - Checks for pending waiver requests or financial hardship applications
    - Calls parent personally:
        - Inquires about reason for non-payment
        - Offers installment plan if needed
        - Documents conversation in system
    - If parent commits to payment:
        - Sets manual reminder for follow-up in 3 days
    - If no response:
        - Escalates to principal for further action

E4. Installment Plan Setup (for parents facing financial difficulty)

- Parent requests installment plan (via portal or phone call)
- Finance admin:

    - Opens student fee record
    - Selects "Create Installment Plan"
    - Configures:
        - Total amount: ₹24,833
        - Number of installments: 3
        - Installment amount: ₹8,278 each
        - Due dates: July 31, Aug 31, Sep 30
    - Gets principal approval (digital approval via app)
- System generates 3 separate invoices
- Parent receives new payment schedule via SMS/email
- System sends reminders for each installment

E5. Generate Defaulter Report for Management

- Every Monday morning at 9 AM, system auto-generates "Weekly Defaulter Report"
- Report includes:
    - Total overdue amount: ₹7,45,000
    - Number of defaulters: 48 students (8% of total students)
    - Critical cases (60+ days): 2 students
    - Class-wise breakdown
    - Action taken: Reminders sent, calls made, installment plans offered
- Report emailed to:
    - Principal
    - Vice Principal
    - Finance Head
- Principal reviews and decides:
    - Continue follow-up
    - Issue final warning
    - Restrict exam participation (rare, only after exhausting all options)

Expected Outcome: Outstanding fees tracked proactively, automated reminders reduce manual effort by 90%, early intervention prevents long-term defaults

---

Journey F: Principal - Financial Monitoring & Approvals

Frequency: Daily dashboard view, Weekly reports, Monthly reviews **Duration**: 10-15 minutes daily **Current Manual Effort**: 30-45 minutes daily

Steps

F1. Morning Dashboard Review

- Principal opens "Executive Dashboard" on iPad
- Sees at-a-glance summary:
    - **Yesterday's Collections**: ₹10,95,000 (44 payments)
    - **MTD Collections**: ₹85,00,000 (72% of monthly target)
    - **Total Outstanding**: ₹25,45,000 (17% of annual fees)
    - **Overdue > 30 days**: 7 students (₹95,000) - **Needs Attention**

- **Collection Trend Chart** (Last 30 days):
  - Visual line chart showing daily collections
  - Target vs Actual bars
- **Top Metrics**:
  - Collection efficiency: 83% (on-time payments)
  - Payment method breakdown: 89% online, 11% cash
  - Average payment time: 1.8 minutes (online)
  - Parent satisfaction: 92% (based on app ratings)

F2. Approve Fee Waiver Requests

- Principal sees notification: "3 waiver requests pending approval"
- Opens "Pending Approvals" section
- **Case 1**: Scholarship application
  - Student: Priya Sharma (Class 9-A)
  - Reason: Academic merit (95% in Class 8 exams)
  - Request: 50% tuition waiver (₹30,000)
  - Finance admin recommendation: Approve (meets criteria)
  - Attached documents: Mark sheet, financial hardship letter
- Principal reviews:
  - Checks academic records (integrated with SIS)
  - Verifies marks: 95% ✓
  - School policy: >90% = 50% waiver ✓
- **Decision**: Approve
  - Clicks "Approve" button
  - Adds comment: "Well deserved. Excellent academic performance."
  - System auto-updates:
    - Student's fee reduced by ₹30,000
    - New invoice generated
    - Parent notified: "Congratulations! 50% scholarship approved."

F3. Monitor Critical Defaulters

- Principal reviews "Overdue > 30 days" list (7 students)
- For each case:
  - Sees payment history, reminder history, admin notes
  - Decides action:
    - **Case 1**: Parent facing job loss (documented)
      - Decision: Offer 3-month installment plan
      - Assigns to finance admin for follow-up
    - **Case 2**: Parent not responding despite multiple reminders
      - Decision: Schedule parent meeting
      - Assigns to class teacher for coordination
    - **Case 3**: Habitual late payer (same issue every term)
      - Decision: Final warning letter
      - Assigns to finance admin to send official notice
- System records all decisions with timestamps

F4. Review Weekly Collection Report (Monday morning)

- Principal receives "Weekly Collection Summary" via WhatsApp
- Report includes:
  - **Last Week Collections**: ₹35,50,000
  - **Target**: ₹40,00,000
  - **Achievement**: 89% (11% shortfall)
  - **Key Insights**:
    - Online payments increased 15% vs last week
    - Cash payments decreased 25% (positive trend)
    - 5 new defaulters identified
    - 12 defaulters cleared dues
- **Action Items**:
  - Follow up with 5 new defaulters
  - Investigate 11% shortfall (seasonal or concerning trend?)
- Principal clicks "View Detailed Report" → Opens full dashboard

F5. Quarterly Financial Review with Board

- Principal prepares for board meeting
- Opens "Quarterly Financial Reports"
- System provides pre-built reports:
  - **Fee Collection Summary (Q1: Apr-Jun 2025)**:
    - Total Billed: ₹1,80,00,000
    - Total Collected: ₹1,54,55,000 (86%)
    - Outstanding: ₹25,45,000 (14%)
  - **Collection Efficiency**:
    - On-time payments: 83%
    - Overdue > 30 days: 5%
    - Defaulters > 60 days: 1.2%
  - **Payment Method Breakdown**:
    - UPI: 65%
    - Cards: 18%
    - Net Banking: 6%
    - Cash: 11%
  - **Trend Analysis**:
    - Online adoption increased 22% YoY
    - Reconciliation accuracy: 99.7%
    - Audit readiness: 100%
- Principal exports reports as PDF
- Presents to school board with confidence (data-backed insights)

Expected Outcome: Principal spends 10-15 minutes daily on fee monitoring (vs 45 minutes earlier), makes informed decisions with real-time data

---

Journey G: Accountant - Tally Integration & GST Compliance

Frequency: Monthly (end of month) **Duration**: 30-45 minutes **Current Manual Effort**: 4-6 hours

Steps

G1. Export Fee Collection Data to Tally

- Accountant logs into system at month-end (June 30, 2025)
- Navigates to "Exports" → "Tally Export"
- Selects:
  - Month: June 2025
  - Export format: Tally XML
  - Include: All fee collections, receipts, ledger entries
- System generates Tally-compatible XML file:
  - Voucher entries for each payment
  - Ledger: Student name + admission number
  - Amount, date, receipt number
  - Payment method (cash/bank/gateway)
- Accountant downloads XML
- Imports into Tally Prime:
  - Gateway → Import Data → Company Data
  - Selects XML file → Import
  - All 450 transactions imported in 2 minutes
- Verifies:
  - Total collection matches: ₹1,45,67,000 ✓
  - No duplicate entries ✓
  - All ledgers updated ✓

G2. Generate GST Reports

- Accountant opens "GST Compliance" section
- Selects quarter: Q1 (Apr-Jun 2025)
- System auto-generates:
  - **GSTR-1 Report** (Outward Supplies):
    - Total taxable value: ₹1,23,45,000
    - GST @ 18%: ₹22,22,100
    - Total invoice value: ₹1,45,67,100
  - **GSTR-3B Report** (Summary):
    - Outward supplies: ₹1,45,67,100
    - Input tax credit: ₹2,15,000 (on software, office expenses)
    - Net GST payable: ₹20,07,100
- **Invoice-wise breakup**:
  - All 450 invoices listed with:
    - Invoice number, date, student name, GSTIN (if applicable)
    - Taxable amount, GST amount, total
- Accountant exports:
  - GSTR-1 JSON file (for GST portal upload)
  - GSTR-3B Excel (for review)
- Uploads JSON to GST portal

- Files GSTR-1 and GSTR-3B on time (no penalties)

G3. Bank Reconciliation Statement

- Accountant prepares BRS (Bank Reconciliation Statement)
- System provides:
  - **School Records**: ₹1,45,67,000 collected
  - **Bank Statement**: ₹1,44,50,000 credited
  - **Difference**: ₹1,17,000 (pending settlement)
- System auto-identifies:
  - 15 transactions pending T+2 settlement (₹1,17,000)
  - All other transactions matched
- BRS generated with:
  - Opening balance
  - Add: Collections as per school records
  - Less: Bank charges (₹5,000)
  - Less: Payment gateway fees (₹35,000)
  - Add: Interest earned (₹8,000)
  - Closing balance (matches bank statement)
- Accountant exports BRS as PDF for auditor

G4. Financial Year-End Reports

- Accountant prepares year-end reports (March 31, 2026)
- System provides:
  - **Income Statement**: Total fee income: ₹5,85,00,000
  - **Outstanding Report**: ₹12,50,000 (2.1% of annual fees)
  - **Refund Report**: ₹2,50,000 (10 cases)
  - **Bad Debts**: ₹1,25,000 (5 students - written off)
- All reports audit-ready with complete transaction trail
- Auditor reviews → No discrepancies found → Clean audit report

Expected Outcome: Month-end accounting completed in 30-45 minutes (vs 4-6 hours), GST filed on time, audit-ready reports available instantly

---

5. Features & Requirements

5.1 Core Features

Feature 1: Fee Structure Management

Description: Configure and manage fee structures for all classes and fee types.

Functional Requirements:

- **FR-1.1**: System shall allow creating unlimited fee types (tuition, exam, library, sports, transport, hostel, etc.)
- **FR-1.2**: Each fee type shall have: name, description, mandatory/optional flag, recurring schedule, GST applicability

- **FR-1.3**: System shall support class-wise fee configuration (Pre-KG to Class 12)
- **FR-1.4**: System shall allow defining multiple fee structures per academic year
- **FR-1.5**: System shall support installment configuration (monthly, quarterly, annual)
- **FR-1.6**: System shall allow setting due dates per installment
- **FR-1.7**: System shall support late fee penalty configuration (% per month, max cap)
- **FR-1.8**: System shall allow copying fee structures from previous year (with adjustments)

Non-Functional Requirements:

- **NFR-1.1**: Fee structure setup for all classes shall complete within 30 minutes
- **NFR-1.2**: System shall validate fee amounts (no negative values, reasonable ranges)
- **NFR-1.3**: Audit trail for all fee structure changes (who changed, when, old vs new values)

Acceptance Criteria:

- ☐ Finance admin can create 15+ fee types in under 10 minutes
- ☐ Class-wise fee structures defined for all 15 classes (Pre-KG to Class 12)
- ☐ Installment schedules configured with due dates
- ☐ Late fee rules applied automatically after due date
- ☐ Fee structure changes logged with complete audit trail

Feature 2: Discount & Scholarship Management

Description: Automate discount application and manage scholarship workflows.

Functional Requirements:

- **FR-2.1**: System shall support automatic discount rules:
  - Sibling discount (10% 2nd child, 15% 3rd child)
  - Early payment discount (5% if paid X days before due date)
  - Merit scholarship (25-50% based on marks)
- **FR-2.2**: System shall allow manual waiver/discount application with approval workflow
- **FR-2.3**: Waiver requests shall route to finance admin → principal for approval
- **FR-2.4**: System shall apply discounts at invoice generation
- **FR-2.5**: System shall maintain discount history per student
- **FR-2.6**: System shall support one-time and recurring discounts

Non-Functional Requirements:

- **NFR-2.1**: Discount calculation shall be accurate to 2 decimal places
- **NFR-2.2**: Approval workflow shall complete within 24 hours (with notifications)
- **NFR-2.3**: Discount rules shall be configurable without code changes

Acceptance Criteria:

- ☐ Sibling discounts auto-applied for families with multiple children
- ☐ Early payment discount applied if paid before configured date
- ☐ Merit scholarship workflow: request → review → approval → invoice adjustment
- ☐ Discount history visible in student fee ledger

- □ Waiver approvals completed within 24 hours with digital signatures

---

Feature 3: Invoice Generation & Management

Description: Auto-generate invoices and send to parents via multiple channels.

Functional Requirements:

- **FR-3.1**: System shall auto-generate invoices upon fee assignment
- **FR-3.2**: Invoice shall include:
  - Unique invoice number (FC/YYYY-YY/NNNNNN)
  - Student details (name, admission number, class, section)
  - Fee breakup (itemized list with amounts)
  - Due date, late fee rules
  - Payment instructions (bank details, UPI QR code, payment link)
  - GST breakup (if applicable)
  - School logo, authorized signatory
- **FR-3.3**: System shall support bulk invoice generation (entire class/school)
- **FR-3.4**: System shall send invoices via:
  - Email (PDF attachment)
  - SMS (invoice number + payment link)
  - App notification
  - WhatsApp (optional, via API integration)
- **FR-3.5**: System shall support invoice regeneration (with same or updated details)
- **FR-3.6**: System shall allow manual invoice adjustments (with approval)
- **FR-3.7**: System shall calculate pro-rated fees for mid-session admissions

Non-Functional Requirements:

- **NFR-3.1**: Bulk invoice generation: 500 invoices in under 2 minutes
- **NFR-3.2**: Invoice PDF generation: < 3 seconds per invoice
- **NFR-3.3**: Notification delivery: < 1 minute after invoice generation
- **NFR-3.4**: Invoice format shall be configurable (school branding)

Acceptance Criteria:

- □ Invoices auto-generated upon fee assignment
- □ Invoice PDF includes all mandatory fields and school branding
- □ Parents receive invoice via email + SMS within 1 hour
- □ Bulk invoice generation for 500+ students completed in < 2 minutes
- □ Pro-rated fees calculated correctly for mid-session admissions
- □ Invoice adjustments require principal approval

Feature 4: Payment Gateway Integration

Description: Integrate with Razorpay/PayU/Paytm for UPI, card, net banking payments.

Functional Requirements:

- **FR-4.1**: System shall integrate with Razorpay (primary) and PayU/Paytm (alternatives)
- **FR-4.2**: System shall support payment methods:
  - UPI (Google Pay, PhonePe, Paytm, BHIM, etc.)
  - Credit/Debit Cards (Visa, Mastercard, RuPay)
  - Net Banking (50+ banks)
  - Wallets (Paytm, PhonePe, Mobikwik)
- **FR-4.3**: Payment flow:
  - Parent clicks "Pay Now" → Redirected to payment gateway
  - Gateway processes payment → Sends webhook to system
  - System verifies webhook signature (security)
  - System updates invoice status + student ledger
  - System generates receipt
- **FR-4.4**: System shall handle payment failures:
  - Notify parent via SMS/email
  - Provide "Retry Payment" option
  - Log failure reason for admin review
- **FR-4.5**: System shall support partial payments (with outstanding balance tracking)
- **FR-4.6**: System shall support refunds (with approval workflow)
- **FR-4.7**: System shall store minimal payment data (no card details - PCI DSS compliance)

Non-Functional Requirements:

- **NFR-4.1**: Payment initiation: < 2 seconds
- **NFR-4.2**: Payment verification: < 5 seconds (after webhook received)
- **NFR-4.3**: System shall handle 1000+ concurrent payments without performance degradation
- **NFR-4.4**: Payment gateway credentials shall be encrypted (AES-256)
- **NFR-4.5**: Webhook signature verification mandatory (security)

Acceptance Criteria:

- ☐ Parents can pay via UPI, cards, net banking, wallets
- ☐ Payment initiated and completed within 3 minutes
- ☐ Payment webhook verified and processed within 5 seconds
- ☐ Invoice status updated to "PAID" immediately
- ☐ Failed payments logged with reason + retry option provided
- ☐ System handles 1000+ concurrent payments (load tested)
- ☐ PCI DSS compliance: No card data stored on system

Feature 5: Receipt Generation & Delivery

Description: Auto-generate digital receipts and deliver instantly.

Functional Requirements:

- **FR-5.1**: System shall auto-generate receipt upon payment confirmation
- **FR-5.2**: Receipt shall include:
  - Unique receipt number (REC/YYYY-YY/NNNNNN)
  - Student details

  - Payment details (amount, method, transaction ID, date/time)
  - Invoice reference
  - Fee breakup (what was paid)
  - School logo, authorized digital signature
- **FR-5.3**: Receipt shall be generated in PDF format
- **FR-5.4**: System shall send receipt via:
  - Email (PDF attachment)
  - SMS (receipt number + download link)
  - App notification (with in-app PDF viewer)
  - WhatsApp (optional)
- **FR-5.5**: Parents shall be able to download receipts anytime from payment history
- **FR-5.6**: System shall support receipt regeneration (duplicate copy)
- **FR-5.7**: Finance admin shall be able to generate manual receipts (for offline payments)

Non-Functional Requirements:

- **NFR-5.1**: Receipt generation: < 3 seconds after payment confirmation
- **NFR-5.2**: Receipt delivery: < 30 seconds after generation
- **NFR-5.3**: Receipt PDF shall be tamper-proof (digitally signed)
- **NFR-5.4**: Receipt storage: Indefinite (lifetime of student + 7 years post-graduation)

Acceptance Criteria:

- ☐ Receipt auto-generated within 3 seconds of payment
- ☐ Receipt delivered via email + SMS within 30 seconds
- ☐ Receipt PDF includes all mandatory fields and school branding
- ☐ Parents can download receipts anytime from portal
- ☐ Manual receipt generation available for offline payments
- ☐ Receipt format configurable (school branding)

Feature 6: Payment Reconciliation (Automated)

Description: Automatically reconcile payments with payment gateway and bank statements.

Functional Requirements:

- **FR-6.1**: System shall auto-reconcile online payments:
  - Fetch settlement data from payment gateway API (every 15 minutes)
  - Match transaction ID, amount, timestamp
  - Update reconciliation status (Matched/Pending/Failed)
- **FR-6.2**: System shall support bank statement upload (CSV/Excel/PDF)
- **FR-6.3**: System shall auto-match bank credits with system payments:
  - By amount + date (with T+1/T+2 adjustment)
  - By UTR number (for NEFT/RTGS)
  - By gateway reference (for online payments)
- **FR-6.4**: System shall identify unmatched transactions:
  - Bank credits without system payment (unknown source)
  - System payments without bank credit (pending settlement)

- **FR-6.5**: System shall allow manual matching (for edge cases)
- **FR-6.6**: System shall generate reconciliation reports:
  - Daily: Online payment reconciliation
  - Weekly: Bank statement reconciliation
  - Monthly: Comprehensive reconciliation report
- **FR-6.7**: System shall maintain reconciliation audit trail

Non-Functional Requirements:

- **NFR-6.1**: Auto-reconciliation shall run every 15 minutes (configurable)
- **NFR-6.2**: Bank statement processing: 1000 transactions in < 2 minutes
- **NFR-6.3**: Reconciliation accuracy: 99%+ automatic matching
- **NFR-6.4**: Reconciliation reports generated in < 10 seconds

Acceptance Criteria:

- ☐ Online payments auto-reconciled every 15 minutes
- ☐ Bank statement uploaded and processed in < 2 minutes
- ☐ Automatic matching accuracy: 99%+
- ☐ Unmatched transactions identified and flagged for review
- ☐ Manual matching option available for edge cases
- ☐ Daily/weekly/monthly reconciliation reports generated
- ☐ Complete audit trail maintained for all reconciliation actions

---

Feature 7: Outstanding Fee Tracking

Description: Track outstanding fees with aging analysis and automated follow-ups.

Functional Requirements:

- **FR-7.1**: System shall maintain real-time outstanding balance per student
- **FR-7.2**: System shall provide outstanding dashboard with:
  - Total outstanding amount
  - Current dues (not overdue)
  - Overdue aging: 0-15 days, 15-30 days, 30-60 days, 60+ days
  - Class-wise breakdown
  - Student-wise list with contact details
- **FR-7.3**: System shall calculate late fees automatically after grace period
- **FR-7.4**: System shall support partial payment allocation:
  - Allocate to oldest outstanding first
  - Allocate to mandatory fees before optional
  - Principal before penalty
- **FR-7.5**: System shall provide outstanding reports:
  - Current outstanding (live)
  - Defaulter list (overdue > X days)
  - Aging analysis (bucketed by days overdue)
  - Class-wise, section-wise reports
  - Exportable (Excel, PDF)

Non-Functional Requirements:

- **NFR-7.1**: Outstanding balance updated in real-time (< 1 second after payment)
- **NFR-7.2**: Outstanding dashboard shall load in < 5 seconds (for 5000 students)
- **NFR-7.3**: Reports generated in < 10 seconds

Acceptance Criteria:

- □ Real-time outstanding balance visible in student ledger
- □ Outstanding dashboard shows aging analysis with visual charts
- □ Late fees calculated and applied automatically after grace period
- □ Partial payments allocated correctly (oldest first, mandatory before optional)
- □ Defaulter list generated with contact details
- □ Reports exportable in Excel/PDF

Feature 8: Automated Reminder System

Description: Send automated payment reminders based on configured schedules.

Functional Requirements:

- **FR-8.1**: System shall send automated reminders:
  - **Pre-due**: 7 days before due date (friendly reminder)
  - **On-due**: On due date (last day to pay without penalty)
  - **Post-due**: 3, 10, 20, 30 days after due date (escalating severity)
- **FR-8.2**: Reminders shall be sent via:
  - SMS (primary)
  - Email (with invoice PDF)
  - App notification
  - WhatsApp (optional)
- **FR-8.3**: Reminder content shall be customizable (templates)
- **FR-8.4**: Reminders shall include:
  - Student name, class
  - Outstanding amount
  - Due date
  - Late fee (if applicable)
  - Payment link (for instant payment)
- **FR-8.5**: System shall maintain reminder history (who received, when, response)
- **FR-8.6**: System shall support manual reminders (for specific students)
- **FR-8.7**: System shall escalate reminders to management (for 60+ days overdue)
- **FR-8.8**: Parents shall be able to opt-out of SMS reminders (but not critical notices)

Non-Functional Requirements:

- **NFR-8.1**: Reminders shall be sent at configured time (default: 10 AM)
- **NFR-8.2**: Bulk reminder sending: 500 reminders in < 5 minutes
- **NFR-8.3**: Reminder delivery rate: 95%+ (track bounces)
- **NFR-8.4**: Reminder templates configurable without code changes

Acceptance Criteria:

- □ Automated reminders sent as per schedule (pre-due, on-due, post-due)
- □ Reminders delivered via SMS + Email + App
- □ Reminder content includes all mandatory fields + payment link
- □ Reminder history visible in student ledger
- □ Manual reminders can be sent by finance admin
- □ Escalation reminders sent to principal for 60+ days overdue
- □ Bulk reminders sent efficiently (500 in < 5 minutes)

Feature 9: Offline Payment Recording

Description: Record cash/cheque payments manually with digital receipts.

Functional Requirements:

- **FR-9.1**: Finance admin shall be able to record offline payments:
  - Payment method: Cash / Cheque / Bank Transfer / Demand Draft
  - Amount, date, reference number (cheque number, UTR, DD number)
  - Remarks/notes
- **FR-9.2**: System shall validate offline payment:
  - No duplicate payment already recorded
  - Amount within reasonable range
  - Cheque number unique (if cheque payment)
- **FR-9.3**: System shall update:
  - Invoice status (PAID or PARTIAL)
  - Student ledger (outstanding reduced)
  - Daily cash collection report
- **FR-9.4**: System shall auto-generate receipt (same as online)
- **FR-9.5**: Finance admin shall be able to print receipt immediately
- **FR-9.6**: System shall provide daily cash collection report:
  - Total cash received today
  - List of transactions (student, amount, receipt number)
  - Total to be deposited in bank
- **FR-9.7**: Finance admin shall mark cash as "Deposited in Bank" with bank receipt reference

Non-Functional Requirements:

- **NFR-9.1**: Offline payment recording: < 2 minutes per transaction
- **NFR-9.2**: Receipt generation: < 3 seconds
- **NFR-9.3**: Daily cash report generation: < 5 seconds

Acceptance Criteria:

- □ Finance admin can record offline payments in < 2 minutes
- □ System validates duplicate payments and invalid amounts
- □ Receipt auto-generated and printable immediately
- □ Daily cash collection report shows all offline payments

- □ Cash deposit workflow: Record → Deposit → Mark as Deposited

---

Feature 10: Financial Dashboards & Reports

Description: Real-time dashboards and comprehensive financial reports.

Functional Requirements:

- **FR-10.1**: Finance Admin Dashboard:
  - Today's collections (online + offline)
  - MTD (Month-to-Date) collections vs target
  - Total outstanding (current + overdue)
  - Pending reconciliation items
  - Failed payments requiring action
  - Quick actions: Record payment, send reminder, generate report
- **FR-10.2**: Principal Dashboard:
  - Collection summary (today, week, month, quarter)
  - Collection efficiency (% on-time payments)
  - Outstanding overview (total, aging buckets)
  - Top metrics (payment success rate, average payment time, parent satisfaction)
  - Collection trend chart (last 30 days)
  - Pending approvals (waivers, refunds)
- **FR-10.3**: Reports:
  - **Daily Collection Report**: Date-wise collections (online + offline)
  - **Outstanding Report**: Current outstanding with aging analysis
  - **Defaulter Report**: Students with overdue > X days
  - **Class-wise Report**: Collection summary per class
  - **Payment Method Report**: Breakdown by payment method
  - **Reconciliation Report**: Bank vs system matching
  - **GST Report**: GSTR-1, GSTR-3B ready data
  - **Audit Report**: Complete transaction trail
- **FR-10.4**: All reports shall be:
  - Exportable (Excel, PDF, CSV)
  - Filterable (date range, class, payment method, status)
  - Schedulable (auto-send weekly/monthly)
- **FR-10.5**: System shall support custom report builder (basic)

Non-Functional Requirements:

- **NFR-10.1**: Dashboards shall load in < 5 seconds
- **NFR-10.2**: Reports shall generate in < 10 seconds (for 5000 students)
- **NFR-10.3**: Real-time data refresh (no more than 1-minute lag)
- **NFR-10.4**: Dashboards accessible on mobile/tablet (responsive)

Acceptance Criteria:

- □ Finance admin dashboard shows real-time collections and outstanding
- □ Principal dashboard accessible on iPad with visual charts

- □ All reports generated in < 10 seconds
- □ Reports exportable in multiple formats
- □ Scheduled reports auto-sent via email (weekly/monthly)
- □ Custom report builder available for basic filtering

5.2 Integration Requirements

Integration 1: Admission System (Journey 1)

Description: Seamless data flow from admission to fee assignment.

Integration Points:

- When student enrollment confirmed → Trigger fee assignment
- Student data (name, class, parent contact) auto-synced
- Admission fee invoice auto-generated
- Student ID linked to fee ledger

Data Flow:

```
Admission Approved → Student Record Created → Fee Structure Retrieved →
Fee Assigned → Invoice Generated → Parent Notified
```

Acceptance Criteria:

- □ Admitted student automatically assigned fees within 1 minute
- □ Admission fee invoice generated and sent to parent
- □ Student data synced accurately (no manual re-entry)

Integration 2: Payment Gateway (Razorpay/PayU/Paytm)

Description: Secure payment processing with real-time verification.

Integration Points:

- Payment initiation API
- Webhook for payment confirmation
- Refund API
- Settlement data API

Security:

- Webhook signature verification (HMAC-SHA256)
- No card/bank data stored (PCI DSS compliance)
- HTTPS only communication

Acceptance Criteria:

- □ Payment initiation < 2 seconds
- □ Webhook verification successful (100% secure)
- □ Refunds processed within 5-7 business days
- □ Settlement data fetched daily for reconciliation

Integration 3: SMS Gateway (for Notifications)

Description: Reliable SMS delivery for payment notifications.

Integration Points:

- Transactional SMS API (for receipts, reminders)
- Delivery status tracking
- DLT (Distributed Ledger Technology) compliance (India)

Message Types:

- Invoice generated
- Payment reminder (pre-due, on-due, post-due)
- Payment successful (with receipt link)
- Payment failed (with retry link)

Acceptance Criteria:

- □ SMS delivered within 30 seconds
- □ Delivery rate: 95%+
- □ DLT templates approved (India compliance)
- □ Delivery status tracked and logged

Integration 4: Email Service (for Invoices & Receipts)

Description: Professional email delivery with attachments.

Integration Points:

- SMTP/SendGrid/AWS SES integration
- Email templates with school branding
- PDF attachment support

Email Types:

- Invoice with PDF attachment
- Receipt with PDF attachment
- Weekly collection report to management
- Payment reminders

Acceptance Criteria:

- □ Emails delivered within 1 minute
- □ Delivery rate: 98%+

- □ PDF attachments < 500 KB
- □ Email templates customizable (school branding)

Integration 5: Accounting Software (Tally/Busy)

Description: Export fee data to accounting software for bookkeeping.

Integration Points:

- Tally XML export format
- Busy Excel export format
- Voucher generation (payment entries)
- Ledger mapping (student accounts)

Data Exported:

- Payment vouchers (receipt entries)
- Student ledger balances
- Fee collection summary
- Bank reconciliation data

Acceptance Criteria:

- □ Tally XML export generated in < 2 minutes
- □ Imported into Tally without errors
- □ Ledger balances match system records
- □ Voucher entries properly formatted

5.3 Security & Compliance Requirements

Security Requirements

SEC-1: Authentication & Authorization

- Role-based access control (RBAC)
- Roles: Super Admin, Finance Admin, Principal, Parent, Accountant
- Permissions: View, Create, Edit, Delete, Approve, Export
- Session timeout: 30 minutes (configurable)
- Strong password policy (min 8 chars, 1 uppercase, 1 number, 1 special char)

SEC-2: Data Encryption

- Database encryption at rest (AES-256)
- TLS 1.3 for data in transit
- Payment gateway credentials encrypted (secrets manager)
- No plain-text storage of sensitive data

SEC-3: Payment Security

- PCI DSS Level 1 compliance (via payment gateway)

- No card/CVV/OTP data stored on system
- Webhook signature verification (HMAC-SHA256)
- Payment callback validation (prevent tampering)

SEC-4: Audit Trail

- All financial transactions logged with:
  - User ID, timestamp, action, old value, new value
  - IP address, device info
- Logs retained for 7 years (India compliance)
- Tamper-proof logs (write-only)

SEC-5: Access Logs

- Login/logout tracked
- Failed login attempts logged (brute-force detection)
- Financial data access logged (who viewed which student's fee)

Compliance Requirements

COMP-1: GST Compliance (India)

- 18% GST on all fee transactions
- GST invoice generation (with GSTIN, HSN/SAC codes)
- GSTR-1, GSTR-3B report generation
- GST return filing ready data

COMP-2: Data Privacy (India - IT Act 2000, DPDP Act 2023)

- Parent consent for data collection
- Right to access fee data
- Right to download payment history
- Right to delete data (after 7 years retention)

COMP-3: Financial Regulations

- Audit trail for 7 years (India Companies Act)
- Bank reconciliation mandatory
- Receipt numbering sequential and unique
- No backdated entries (timestamp verification)

COMP-4: Education Board Compliance

- Fee structure disclosure (as per RTE Act)
- Fee refund policy (as per state regulations)
- Scholarship/waiver records (as per RTE/government schemes)

5.4 Performance Requirements

PERF-1: Response Time

- Page load: < 3 seconds (desktop), < 5 seconds (mobile)
- Payment initiation: < 2 seconds
- Payment verification: < 5 seconds
- Receipt generation: < 3 seconds
- Dashboard load: < 5 seconds (5000 students)
- Report generation: < 10 seconds (5000 students)

PERF-2: Throughput

- Concurrent payments: 1000+ (load tested)
- Bulk invoice generation: 500 invoices in < 2 minutes
- Bulk reminder sending: 500 SMS in < 5 minutes
- Bank statement processing: 1000 transactions in < 2 minutes

PERF-3: Scalability

- System shall support:
  - 10,000 students per school
  - 100 schools on single deployment (multi-tenant)
  - ₹10 crore annual fee collection per school
  - 5,000+ transactions per day (peak season)

PERF-4: Availability

- Uptime: 99.5% (excluding planned maintenance)
- Planned maintenance: Off-peak hours (11 PM - 2 AM IST)
- Downtime notification: 48 hours advance
- Payment gateway redundancy (primary + backup)

PERF-5: Database Performance

- Query response: < 100ms (95th percentile)
- Transaction commit: < 50ms
- Indexing on critical fields (student ID, invoice number, transaction ID)
- Database backup: Daily (incremental), Weekly (full)

---

6. Technical Architecture

6.1 Technology Stack (Same as Admission System)

Backend:

- **Framework**: FastAPI 0.104+
- **Language**: Python 3.11+
- **Database**: SQLite (development), PostgreSQL 15+ (production)
- **ORM**: SQLAlchemy 2.0
- **Authentication**: JWT (python-jose)
- **Password Hashing**: bcrypt (passlib)

- **Payment Gateway SDK**: Razorpay Python SDK 1.4.1
- **Task Queue**: Celery (for async tasks - reminders, reports)
- **Cache**: Redis 5.0+ (session, dashboard data)

Frontend:

- **Framework**: React 19
- **Language**: TypeScript 5.9
- **UI Library**: Material-UI v7.3
- **State Management**: React Context API / Redux Toolkit
- **Form Handling**: React Hook Form 7.64
- **Validation**: Yup 1.7
- **HTTP Client**: Axios 1.12
- **Build Tool**: Vite 7.1.9

DevOps:

- **Version Control**: Git
- **CI/CD**: GitHub Actions
- **Deployment**: Docker + Docker Compose
- **Monitoring**: Sentry (error tracking), Prometheus + Grafana (metrics)
- **Logging**: Structured logging (JSON format)

6.2 Database Schema (Extends Admission System)

New Tables

1. fee_types

```sql
CREATE TABLE fee_types (
    id SERIAL PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    description TEXT,
    is_mandatory BOOLEAN DEFAULT TRUE,
    is_recurring BOOLEAN DEFAULT FALSE,
    recurring_frequency VARCHAR(20), -- 'monthly', 'quarterly', 'annual'
    gst_applicable BOOLEAN DEFAULT TRUE,
    gst_percentage DECIMAL(5,2) DEFAULT 18.00,
    status VARCHAR(20) DEFAULT 'active',
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

2. fee_structures

```sql
CREATE TABLE fee_structures (
    id SERIAL PRIMARY KEY,
    academic_year VARCHAR(10) NOT NULL, -- '2025-26'
    class_id INTEGER NOT NULL,
    section VARCHAR(10),
    fee_type_id INTEGER REFERENCES fee_types(id),
    amount DECIMAL(10,2) NOT NULL,
    due_date DATE,
    installment_number INTEGER, -- 1, 2, 3 (for Term 1, 2, 3)
    late_fee_percentage DECIMAL(5,2) DEFAULT 2.00,
    late_fee_grace_days INTEGER DEFAULT 7,
    status VARCHAR(20) DEFAULT 'active',
    created_by INTEGER,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    UNIQUE(academic_year, class_id, fee_type_id, installment_number)
);
```

3. student_fee_assignments

```sql
CREATE TABLE student_fee_assignments (
    id SERIAL PRIMARY KEY,
    student_id INTEGER NOT NULL, -- from admission system
    fee_structure_id INTEGER REFERENCES fee_structures(id),
    custom_amount DECIMAL(10,2), -- if different from fee_structure.amount
    discount_amount DECIMAL(10,2) DEFAULT 0.00,
    discount_reason VARCHAR(255), -- 'Sibling discount', 'Merit scholarship', etc.
    assigned_date DATE DEFAULT CURRENT_DATE,
    status VARCHAR(20) DEFAULT 'active', -- 'active', 'cancelled', 'completed'
    created_by INTEGER,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

4. invoices

```sql
CREATE TABLE invoices (
    id SERIAL PRIMARY KEY,
    invoice_number VARCHAR(50) UNIQUE NOT NULL, -- FC/2025-26/001234
    student_id INTEGER NOT NULL,
    academic_year VARCHAR(10) NOT NULL,
    total_amount DECIMAL(10,2) NOT NULL,
    discount_amount DECIMAL(10,2) DEFAULT 0.00,
    gst_amount DECIMAL(10,2) DEFAULT 0.00,
    net_amount DECIMAL(10,2) NOT NULL, -- total - discount + gst
    due_date DATE NOT NULL,
    status VARCHAR(20) DEFAULT 'pending', -- 'pending', 'paid', 'partial',
'overdue', 'cancelled'
    generated_date DATE DEFAULT CURRENT_DATE,
    paid_date DATE,
    created_by INTEGER,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

5. invoice_items

```
CREATE TABLE invoice_items (
    id SERIAL PRIMARY KEY,
    invoice_id INTEGER REFERENCES invoices(id) ON DELETE CASCADE,
    fee_type_id INTEGER REFERENCES fee_types(id),
    description VARCHAR(255) NOT NULL,
    quantity INTEGER DEFAULT 1,
    amount DECIMAL(10,2) NOT NULL,
    gst_percentage DECIMAL(5,2) DEFAULT 18.00,
    gst_amount DECIMAL(10,2) DEFAULT 0.00,
    total_amount DECIMAL(10,2) NOT NULL, -- amount + gst
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

6. payments

```
CREATE TABLE payments (
    id SERIAL PRIMARY KEY,
    payment_reference VARCHAR(100) UNIQUE NOT NULL, -- PAY/2025-26/001234
    invoice_id INTEGER REFERENCES invoices(id),
    student_id INTEGER NOT NULL,
    amount DECIMAL(10,2) NOT NULL,
    payment_method VARCHAR(50) NOT NULL, -- 'UPI', 'Card', 'Net Banking', 'Cash', 'Cheque'
    payment_gateway VARCHAR(50), -- 'Razorpay', 'PayU', 'Paytm', null (for offline)
    transaction_id VARCHAR(255), -- Gateway transaction ID
    gateway_order_id VARCHAR(255), -- Gateway order ID
    gateway_payment_id VARCHAR(255), -- Gateway payment ID
    gateway_signature VARCHAR(500), -- Webhook signature (for verification)
    payment_status VARCHAR(20) DEFAULT 'pending', -- 'pending', 'success', 'failed', 'refunded'
    payment_date TIMESTAMP,
    failure_reason TEXT,
    refund_amount DECIMAL(10,2) DEFAULT 0.00,
    refund_date TIMESTAMP,
    refund_reason TEXT,
    remarks TEXT,
    created_by INTEGER, -- null for online, user_id for offline
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

7. payment_receipts

```
CREATE TABLE payment_receipts (
    id SERIAL PRIMARY KEY,
    receipt_number VARCHAR(50) UNIQUE NOT NULL, -- REC/2025-26/005678
    payment_id INTEGER REFERENCES payments(id),
    invoice_id INTEGER REFERENCES invoices(id),
    student_id INTEGER NOT NULL,
    amount DECIMAL(10,2) NOT NULL,
    receipt_date DATE DEFAULT CURRENT_DATE,
    file_path VARCHAR(500), -- Path to PDF file
    generated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    sent_via_email BOOLEAN DEFAULT FALSE,
    sent_via_sms BOOLEAN DEFAULT FALSE,
    sent_via_whatsapp BOOLEAN DEFAULT FALSE
);
```

8. student_fee_ledger

```
CREATE TABLE student_fee_ledger (
    id SERIAL PRIMARY KEY,
    student_id INTEGER NOT NULL,
    academic_year VARCHAR(10) NOT NULL,
    total_fee_assigned DECIMAL(10,2) DEFAULT 0.00,
    total_paid DECIMAL(10,2) DEFAULT 0.00,
    total_discount DECIMAL(10,2) DEFAULT 0.00,
    total_outstanding DECIMAL(10,2) DEFAULT 0.00,
    late_fee_applied DECIMAL(10,2) DEFAULT 0.00,
    last_payment_date DATE,
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    UNIQUE(student_id, academic_year)
);
```

9. payment_reminders

```
CREATE TABLE payment_reminders (
    id SERIAL PRIMARY KEY,
    student_id INTEGER NOT NULL,
    invoice_id INTEGER REFERENCES invoices(id),
    reminder_type VARCHAR(50) NOT NULL, -- 'pre_due', 'on_due', 'post_due_3',
'post_due_10', etc.
    sent_date DATE DEFAULT CURRENT_DATE,
    sent_via VARCHAR(50), -- 'SMS', 'Email', 'WhatsApp', 'App'
    delivery_status VARCHAR(20), -- 'sent', 'delivered', 'failed', 'bounced'
    next_reminder_date DATE,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

10. reconciliation_logs

```
CREATE TABLE reconciliation_logs (
    id SERIAL PRIMARY KEY,
    reconciliation_date DATE DEFAULT CURRENT_DATE,
    reconciliation_type VARCHAR(50), -- 'payment_gateway', 'bank_statement'
    source VARCHAR(100), -- 'Razorpay', 'ICICI Bank', etc.
    total_transactions INTEGER DEFAULT 0,
    matched_transactions INTEGER DEFAULT 0,
    unmatched_transactions INTEGER DEFAULT 0,
    total_amount_reconciled DECIMAL(10,2) DEFAULT 0.00,
    discrepancy_amount DECIMAL(10,2) DEFAULT 0.00,
    reconciled_by INTEGER,
    reconciliation_file_path VARCHAR(500), -- Path to uploaded bank statement
    remarks TEXT,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

11. discount_policies

```
CREATE TABLE discount_policies (
    id SERIAL PRIMARY KEY,
    policy_name VARCHAR(100) NOT NULL,
    policy_type VARCHAR(50) NOT NULL, -- 'sibling', 'early_payment',
'merit_scholarship', 'custom'
    discount_percentage DECIMAL(5,2),
    discount_fixed_amount DECIMAL(10,2),
    applicable_fee_types TEXT, -- JSON array of fee_type_ids
    eligibility_criteria TEXT, -- JSON object with criteria
    auto_apply BOOLEAN DEFAULT TRUE,
    requires_approval BOOLEAN DEFAULT FALSE,
    status VARCHAR(20) DEFAULT 'active',
    created_by INTEGER,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

12. waiver_requests

```
CREATE TABLE waiver_requests (
    id SERIAL PRIMARY KEY,
    student_id INTEGER NOT NULL,
    academic_year VARCHAR(10) NOT NULL,
    fee_type_id INTEGER REFERENCES fee_types(id),
    requested_amount DECIMAL(10,2) NOT NULL,
    reason TEXT NOT NULL,
    supporting_documents TEXT, -- JSON array of file paths
    requested_by INTEGER, -- parent user_id
    requested_date DATE DEFAULT CURRENT_DATE,
    reviewed_by INTEGER, -- finance admin user_id
    reviewed_date DATE,
    admin_recommendation VARCHAR(20), -- 'approve', 'reject', 'pending'
    admin_remarks TEXT,
    approved_by INTEGER, -- principal user_id
    approved_date DATE,
    approval_status VARCHAR(20) DEFAULT 'pending', -- 'pending', 'approved',
'rejected'
    approval_remarks TEXT,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

6.3 API Endpoints (RESTful)

Fee Structure Management

```
POST   /api/v1/fees/types                       - Create fee type
GET    /api/v1/fees/types                       - List fee types
GET    /api/v1/fees/types/{id}                  - Get fee type details
PUT    /api/v1/fees/types/{id}                  - Update fee type
DELETE /api/v1/fees/types/{id}                  - Delete fee type

POST   /api/v1/fees/structures                  - Create fee structure
GET    /api/v1/fees/structures                  - List fee structures (with filters)
GET    /api/v1/fees/structures/{id}             - Get fee structure details
PUT    /api/v1/fees/structures/{id}             - Update fee structure
DELETE /api/v1/fees/structures/{id}             - Delete fee structure
POST   /api/v1/fees/structures/bulk             - Bulk create fee structures
POST   /api/v1/fees/structures/copy             - Copy from previous year

POST   /api/v1/fees/assign                      - Assign fees to student(s)
POST   /api/v1/fees/assign/bulk                 - Bulk assign fees
GET    /api/v1/fees/students/{id}/assigned      - Get student's assigned fees
PUT    /api/v1/fees/assignments/{id}            - Update fee assignment
DELETE /api/v1/fees/assignments/{id}            - Remove fee assignment
```

Invoice Management

```
POST   /api/v1/invoices/generate                - Generate invoice for student
POST   /api/v1/invoices/bulk-generate           - Bulk generate invoices
GET    /api/v1/invoices                         - List invoices (with filters)
GET    /api/v1/invoices/{id}                    - Get invoice details
GET    /api/v1/invoices/{id}/pdf                - Download invoice PDF
PUT    /api/v1/invoices/{id}                    - Update invoice
```

```
DELETE /api/v1/invoices/{id}                 - Cancel invoice
POST   /api/v1/invoices/{id}/resend          - Resend invoice notification
```

Payment Processing

```
POST   /api/v1/payments/initiate             - Initiate online payment
POST   /api/v1/payments/verify               - Verify payment (webhook)
POST   /api/v1/payments/offline              - Record offline payment
GET    /api/v1/payments                      - List payments (with filters)
GET    /api/v1/payments/{id}                 - Get payment details
POST   /api/v1/payments/{id}/refund          - Initiate refund
GET    /api/v1/payments/methods              - Get available payment methods
```

Receipt Management

```
GET    /api/v1/receipts/{id}                 - Get receipt details
GET    /api/v1/receipts/{id}/pdf             - Download receipt PDF
POST   /api/v1/receipts/{id}/email           - Email receipt
POST   /api/v1/receipts/{id}/sms             - SMS receipt link
POST   /api/v1/receipts/regenerate           - Regenerate receipt
GET    /api/v1/receipts/student/{id}         - Get student's all receipts
```

Reconciliation

```
POST   /api/v1/reconciliation/gateway        - Trigger gateway reconciliation
POST   /api/v1/reconciliation/bank-statement - Upload bank statement
GET    /api/v1/reconciliation/status         - Get reconciliation status
GET    /api/v1/reconciliation/unmatched      - List unmatched transactions
POST   /api/v1/reconciliation/manual-match   - Manually match transaction
GET    /api/v1/reconciliation/report         - Generate reconciliation report
```

Outstanding & Reminders

```
GET    /api/v1/outstanding/dashboard         - Outstanding dashboard data
GET    /api/v1/outstanding/students          - List students with outstanding
GET    /api/v1/outstanding/students/{id}     - Get student outstanding details
GET    /api/v1/outstanding/aging             - Get aging analysis
GET    /api/v1/outstanding/defaulters        - Get defaulter list

POST   /api/v1/reminders/send                - Send reminder (manual)
POST   /api/v1/reminders/bulk                - Send bulk reminders
GET    /api/v1/reminders/history             - Get reminder history
GET    /api/v1/reminders/schedule             - Get reminder schedule config
PUT    /api/v1/reminders/schedule             - Update reminder schedule
```

Discounts & Waivers

```
POST   /api/v1/discounts/policies             - Create discount policy
GET    /api/v1/discounts/policies             - List discount policies
PUT    /api/v1/discounts/policies/{id}        - Update discount policy
DELETE /api/v1/discounts/policies/{id}        - Delete discount policy

POST   /api/v1/waivers/request                - Submit waiver request
GET    /api/v1/waivers/pending                - List pending waivers
GET    /api/v1/waivers/{id}                   - Get waiver details
PUT    /api/v1/waivers/{id}/review            - Review waiver (admin)
PUT    /api/v1/waivers/{id}/approve           - Approve/reject waiver (principal)
```

Reports & Analytics

```
GET    /api/v1/reports/daily-collection       - Daily collection report
GET    /api/v1/reports/outstanding            - Outstanding report
GET    /api/v1/reports/defaulters             - Defaulter report
GET    /api/v1/reports/class-wise             - Class-wise report
GET    /api/v1/reports/payment-method         - Payment method breakdown
GET    /api/v1/reports/reconciliation         - Reconciliation report
GET    /api/v1/reports/gst                    - GST report (GSTR-1/3B)
GET    /api/v1/reports/audit                  - Audit trail report

GET    /api/v1/dashboard/finance-admin        - Finance admin dashboard data
GET    /api/v1/dashboard/principal            - Principal dashboard data
GET    /api/v1/dashboard/parent               - Parent dashboard data (for logged-in parent)
```

Exports

```
GET    /api/v1/exports/tally                  - Export to Tally XML
GET    /api/v1/exports/busy                   - Export to Busy Excel
GET    /api/v1/exports/excel                  - Export report to Excel
GET    /api/v1/exports/pdf                    - Export report to PDF
```

6.4 Frontend Routes

```
Parent Portal:
/parent/dashboard                      - Parent dashboard (fee overview)
/parent/fees                           - All fees and invoices
/parent/fees/invoice/{id}              - Invoice details
/parent/fees/payment                   - Payment page
/parent/fees/payment-history           - Payment history
/parent/fees/receipts                  - All receipts
/parent/fees/outstanding               - Outstanding details

Admin Portal:
/admin/dashboard                       - Admin dashboard
/admin/fees/types                      - Fee types management
/admin/fees/structures                 - Fee structures management
/admin/fees/assign                     - Assign fees (bulk)
/admin/invoices                        - Invoice list
/admin/invoices/{id}                   - Invoice details
/admin/payments                        - Payment list
/admin/payments/offline                - Record offline payment
/admin/reconciliation                  - Reconciliation dashboard
/admin/outstanding                     - Outstanding tracker
/admin/reminders                       - Reminder management
/admin/waivers                         - Waiver requests
/admin/reports                         - Reports section

Principal Portal:
/principal/dashboard                   - Principal dashboard
/principal/collection                  - Collection overview
/principal/outstanding                 - Outstanding overview
/principal/approvals                   - Pending approvals
/principal/reports                     - Financial reports

Accountant Portal:
/accountant/dashboard                  - Accountant dashboard
/accountant/reconciliation             - Reconciliation reports
/accountant/exports                    - Export to Tally/Busy
/accountant/gst                        - GST reports
```

7. Success Metrics

7.1 Operational Metrics

OP-1: Fee Collection Efficiency

- **Target**: 95%+ fee collection within due dates
- **Current Baseline**: 70% (manual system)
- **Measurement**: % of fees collected by due date

OP-2: Reconciliation Effort

- **Target**: 90% reduction in manual reconciliation effort

- **Current Baseline**: 4-6 hours weekly
- **New Target**: < 30 minutes weekly
- **Measurement**: Time spent on reconciliation (hours/week)

OP-3: Payment Success Rate

- **Target**: 99%+ payment success rate
- **Measurement**: (Successful Payments / Total Payment Attempts) × 100

OP-4: Outstanding Reduction

- **Target**: 80%+ reduction in 60+ days overdue
- **Current Baseline**: 8% of students overdue > 60 days
- **New Target**: < 1.5% overdue > 60 days
- **Measurement**: % of students with overdue > 60 days

OP-5: Daily Reconciliation

- **Target**: 100% daily collections reconciled within 24 hours
- **Measurement**: % of transactions reconciled within 24 hours

---

7.2 User Experience Metrics

UX-1: Parent Payment Time

- **Target**: < 3 minutes average payment completion time
- **Measurement**: Time from "Pay Now" click to receipt generation

UX-2: Parent Satisfaction

- **Target**: 90%+ parent satisfaction with payment experience
- **Measurement**: In-app rating (1-5 stars) + feedback surveys

UX-3: Admin Time Savings

- **Target**: 80% reduction in admin time spent on fee management
- **Current Baseline**: 20 hours/week
- **New Target**: < 4 hours/week
- **Measurement**: Time logs + user feedback

UX-4: Payment Failure Rate

- **Target**: < 2% payment failures due to technical issues
- **Measurement**: (Failed Payments / Total Payment Attempts) × 100

UX-5: Receipt Delivery Time

- **Target**: < 30 seconds from payment success to receipt delivery
- **Measurement**: Timestamp difference (payment success → SMS/email sent)

7.3 Financial Metrics

FIN-1: Revenue Acceleration

- **Target**: 30% reduction in fee collection cycle time
- **Current Baseline**: 30+ days average collection time
- **New Target**: < 21 days average collection time
- **Measurement**: Days from invoice generation to payment received

FIN-2: Late Fee Recovery

- **Target**: 50% increase in late fee collection (through automated reminders)
- **Current Baseline**: ₹2,00,000/year late fee collected
- **New Target**: ₹3,00,000/year late fee collected
- **Measurement**: Total late fees collected (annual)

FIN-3: Bad Debt Reduction

- **Target**: 50% reduction in bad debts (written off)
- **Current Baseline**: 2% of annual fees written off (₹5,00,000)
- **New Target**: < 1% written off (₹2,50,000)
- **Measurement**: Amount written off as bad debt (annual)

FIN-4: Payment Gateway Costs

- **Target**: < 1.5% transaction cost on online payments
- **Measurement**: (Gateway Fees / Total Online Collection) × 100

FIN-5: Collection Rate

- **Target**: 98%+ of billed fees collected within academic year
- **Measurement**: (Total Collected / Total Billed) × 100

7.4 Technical Metrics

TECH-1: System Uptime

- **Target**: 99.5% uptime (excluding planned maintenance)
- **Measurement**: (Total Uptime / Total Time) × 100

TECH-2: API Response Time

- **Target**: < 500ms average API response time (95th percentile)
- **Measurement**: API monitoring (Prometheus/Grafana)

TECH-3: Payment Processing Time

- **Target**: < 10 seconds from payment initiation to confirmation
- **Measurement**: Timestamp difference (payment initiate → webhook received)

TECH-4: Concurrent User Load

- **Target**: Support 500+ concurrent users without degradation
- **Measurement**: Load testing results

TECH-5: Error Rate

- **Target**: < 0.1% error rate (excluding user input errors)
- **Measurement**: (Total Errors / Total Requests) × 100

7.5 Adoption Metrics

ADOPT-1: Online Payment Adoption

- **Target**: 85%+ parents paying online within 3 months
- **Current Baseline**: 0% (no online payment option)
- **Measurement**: (Online Payments / Total Payments) × 100

ADOPT-2: Parent Portal Usage

- **Target**: 70%+ parents actively using portal (monthly)
- **Measurement**: (Active Users / Total Parents) × 100

ADOPT-3: Reminder Response Rate

- **Target**: 40%+ parents pay within 3 days of reminder
- **Measurement**: (Payments within 3 days / Reminders Sent) × 100

ADOPT-4: Mobile App Usage

- **Target**: 60%+ payments via mobile app (vs web)
- **Measurement**: (Mobile Payments / Total Online Payments) × 100

---

8. Risk Assessment

8.1 Technical Risks

RISK-1: Payment Gateway Downtime

- **Severity**: High
- **Probability**: Medium
- **Impact**: Parents unable to pay, collections delayed
- **Mitigation**:
  - Integrate 2 payment gateways (Razorpay primary, PayU backup)
  - Auto-switch to backup if primary fails (circuit breaker pattern)
  - Offline payment option always available
  - Clear communication to parents during downtime

RISK-2: Database Performance Degradation

- **Severity**: High
- **Probability**: Low
- **Impact**: Slow dashboard, API timeouts
- **Mitigation**:
  - Proper database indexing (student_id, invoice_number, transaction_id)

  - Query optimization (avoid N+1 queries)
  - Database connection pooling
  - Read replicas for reporting queries
  - Regular database maintenance (vacuum, analyze)

RISK-3: Security Breach (Payment Data)

- **Severity**: Critical
- **Probability**: Low
- **Impact**: Financial loss, reputational damage, legal liability
- **Mitigation**:
  - Never store card/CVV/OTP data (PCI DSS compliance)
  - All payment data via gateway (tokenization)
  - Webhook signature verification (prevent tampering)
  - Regular security audits (penetration testing)
  - Bug bounty program

RISK-4: SMS/Email Delivery Failure

- **Severity**: Medium
- **Probability**: Medium
- **Impact**: Parents don't receive invoices/receipts/reminders
- **Mitigation**:
  - Multi-channel delivery (SMS + Email + App)
  - Retry mechanism (3 attempts with exponential backoff)
  - Delivery status tracking and logging
  - Alternate SMS gateway (backup)
  - Parent portal always accessible (manual download)

8.2 Business Risks

RISK-5: Low Parent Adoption (Digital Payments)

- **Severity**: High
- **Probability**: Medium
- **Impact**: Target metrics not achieved, manual work continues
- **Mitigation**:
  - User-friendly interface (< 3 steps to pay)
  - Multiple payment options (UPI, cards, net banking)
  - Offline payment option always available
  - Parent onboarding sessions (video tutorials, help desk)
  - Incentives (early payment discount, waiver of payment gateway fee)

RISK-6: Reconciliation Discrepancies

- **Severity**: High
- **Probability**: Medium
- **Impact**: Audit failures, financial irregularities
- **Mitigation**:

  - Automated reconciliation (99%+ accuracy)
  - Daily reconciliation mandatory (not weekly)
  - Manual review of unmatched transactions (< 1%)
  - Complete audit trail (all transactions logged)
  - Monthly reconciliation reports to principal/accountant

RISK-7: Payment Gateway Fee Increase

- **Severity**: Medium
- **Probability**: Medium
- **Impact**: Higher operational costs, reduced margins
- **Mitigation**:
  - Negotiate long-term contracts with gateways
  - Pass gateway fee to parents (optional: ₹5-10 convenience fee)
  - Monitor gateway fees vs convenience fee revenue
  - Switch to lower-cost gateway if needed (multi-gateway support)

8.3 Compliance Risks

RISK-8: GST Non-compliance

- **Severity**: High
- **Probability**: Low
- **Impact**: Penalties, legal issues
- **Mitigation**:
  - GST calculation automated (18% on all services)
  - GST invoice generation (with GSTIN, HSN/SAC)
  - GSTR-1/3B reports auto-generated
  - Accountant review before filing
  - Regular GST audits (internal + external)

RISK-9: Data Privacy Violation (DPDP Act)

- **Severity**: Critical
- **Probability**: Low
- **Impact**: Legal penalties, reputational damage
- **Mitigation**:
  - Parent consent for data collection (registration form)
  - Data minimization (collect only necessary data)
  - Data encryption (at rest + in transit)
  - Right to access/download/delete data
  - Privacy policy clearly communicated

RISK-10: Audit Failures

- **Severity**: High
- **Probability**: Low
- **Impact**: Financial irregularities, management trust loss
- **Mitigation**:

  - Complete audit trail (all transactions logged)
  - Reconciliation reports (daily/weekly/monthly)
  - Financial year-end reports auto-generated
  - External auditor read-only access to system
  - Regular internal audits (quarterly)

8.4 Operational Risks

RISK-11: Staff Resistance (Change Management)

- **Severity**: Medium
- **Probability**: High
- **Impact**: Slow adoption, continued manual processes
- **Mitigation**:
  - Comprehensive staff training (finance admin, principal)
  - Step-by-step user manuals (with screenshots)
  - Dedicated support during transition (first 2 months)
  - Highlight benefits (time savings, accuracy, real-time visibility)
  - Phased rollout (start with online payments, then reconciliation)

RISK-12: Data Migration Errors

- **Severity**: High
- **Probability**: Medium
- **Impact**: Incorrect outstanding balances, parent disputes
- **Mitigation**:
  - Careful data migration planning (student data, outstanding balances)
  - Data validation (cross-check with existing records)
  - Dry run migration (test environment)
  - Manual verification of migrated data (sample check)
  - Option to adjust balances (with principal approval)

RISK-13: Parent Disputes (Payment Not Recorded)

- **Severity**: Medium
- **Probability**: Medium
- **Impact**: Parent dissatisfaction, manual resolution time
- **Mitigation**:
  - Instant payment confirmation (SMS + Email + App)
  - Digital receipt (PDF with transaction ID)
  - Complete payment history visible to parents
  - Admin can manually verify and update (with proof)
  - Reconciliation catches missing payments (within 24 hours)

---

9. Release Plan

9.1 Development Phases

Phase 1: MVP (Minimum Viable Product) - 4 Weeks

Week 1-2: Core Fee Management

- Fee structure setup (fee types, class-wise fees)
- Fee assignment (bulk + individual)
- Invoice generation (PDF with school branding)
- Invoice delivery (Email + SMS)
- Manual offline payment recording

Week 3-4: Payment Processing

- Razorpay payment gateway integration
- UPI, card, net banking payments
- Payment webhook verification
- Receipt generation (PDF)
- Receipt delivery (Email + SMS)
- Parent dashboard (view invoices, pay, download receipts)

Deliverables:

- □ Fee structure configured for all classes
- □ Invoices generated and sent to all parents
- □ Parents can pay online (Razorpay)
- □ Receipts auto-generated and delivered
- □ Offline payments recorded manually

Success Criteria:

- □ 50+ parents complete online payment successfully
- □ 95%+ payment success rate
- □ < 3 minutes average payment time

Phase 2: Automation & Reconciliation - 3 Weeks

Week 5: Automated Reconciliation

- Payment gateway auto-reconciliation (every 15 minutes)
- Bank statement upload and auto-matching
- Unmatched transaction identification
- Reconciliation dashboard
- Daily/weekly reconciliation reports

Week 6: Outstanding Tracking & Reminders

- Real-time outstanding balance tracking
- Outstanding dashboard (aging analysis)
- Automated reminder system (pre-due, on-due, post-due)
- SMS/Email reminder delivery
- Reminder history tracking

Week 7: Discounts & Late Fees

- Discount policy configuration (sibling, early payment, merit)
- Auto-apply discounts at invoice generation
- Late fee calculation (automatic after grace period)
- Waiver request workflow (request → review → approval)

Deliverables:

- □ Daily reconciliation automated (99%+ accuracy)
- □ Automated reminders sent (7 days before due, on due, post due)
- □ Outstanding tracked with aging buckets
- □ Discounts auto-applied
- □ Late fees calculated automatically

Success Criteria:

- □ Reconciliation time reduced from 4 hours to < 30 minutes weekly
- □ 90%+ automated reconciliation accuracy
- □ 40%+ parents pay within 3 days of reminder

Phase 3: Advanced Features - 2 Weeks

Week 8: Financial Dashboards & Reports

- Finance admin dashboard (collections, outstanding, reconciliation)
- Principal dashboard (executive summary, trends, approvals)
- Parent dashboard (fee overview, payment history)
- Pre-built reports (daily collection, outstanding, defaulter, class-wise)
- Export to Excel/PDF

Week 9: Integrations

- Tally export (XML format)
- GST reports (GSTR-1, GSTR-3B ready)
- WhatsApp integration (optional notifications)
- PayU/Paytm integration (backup gateways)

Deliverables:

- □ Real-time dashboards for all user roles
- □ 10+ pre-built financial reports
- □ Tally export working
- □ GST reports ready
- □ WhatsApp notifications (optional)

Success Criteria:

- □ Principal can view collection status in < 5 seconds
- □ Tally export imported without errors

- □ GST reports match accountant's expectations

Phase 4: Testing & Launch - 1 Week

Week 10: Testing & Deployment

- Load testing (1000+ concurrent payments)
- Security testing (penetration testing)
- User acceptance testing (with finance admin + parents)
- Bug fixes and optimizations
- Production deployment
- User training (finance admin, principal)
- Parent onboarding (emails, videos, help desk)

Deliverables:

- □ All bugs fixed (P0, P1 priority)
- □ Load tested (1000+ concurrent payments)
- □ Security audit passed
- □ User training completed
- □ Production launch

Success Criteria:

- □ Zero critical bugs in production
- □ 500+ parents onboarded in first week
- □ 80%+ online payment adoption in first month

9.2 Launch Strategy

Pre-Launch (Week 9):

- □ Send email to all parents announcing new fee payment system
- □ Conduct parent webinar (demo + Q&A)
- □ Publish video tutorials (How to pay online)
- □ Set up help desk (phone + email support)
- □ Prepare FAQs document

Launch Day (Week 10, Day 1):

- □ Generate and send invoices to all parents
- □ Activate online payment
- □ Monitor system closely (real-time dashboard)
- □ Resolve issues immediately (dedicated support team)

Post-Launch (Week 10, Day 2-7):

- □ Daily monitoring (payment success rate, errors)
- □ Collect parent feedback (in-app rating, surveys)

- □ Fix bugs reported (hotfix releases)
- □ Follow up with parents who haven't paid (phone calls)

First Month Review (Week 14):

- □ Review adoption metrics (online payment %, parent satisfaction)
- □ Review operational metrics (reconciliation time, outstanding %)
- □ Gather feedback from finance admin, principal, parents
- □ Plan improvements for Phase 5

9.3 Post-Launch Roadmap (Phase 5+)

Phase 5: Advanced Analytics (Month 2-3)

- Predictive analytics (collection forecasting, at-risk accounts)
- Parent behavior analysis (payment patterns, preferred methods)
- Collection trend analysis (YoY comparison, seasonal patterns)
- Custom report builder (drag-drop interface)

Phase 6: Mobile App (Month 3-4)

- Native Android/iOS app (React Native)
- Push notifications (fee due, payment confirmation)
- Biometric authentication (fingerprint, face ID)
- Quick UPI payment (one-tap pay)

Phase 7: Multi-School Support (Month 4-5)

- Multi-tenant architecture (separate data per school)
- School-level configuration (fee structure, branding)
- Consolidated reporting (for school chains)
- Centralized user management

Phase 8: Advanced Integrations (Month 5-6)

- Bank API integration (direct bank statement fetch)
- Accounting software API (Tally, Zoho Books)
- Education board integration (CBSE, IB fee reporting)
- Government portal integration (fee transparency reports)

---

10. Appendices

Appendix A: Glossary

Academic Year: 12-month period (April to March in India) for which fees are structured **Admission Fee**: One-time fee paid at the time of admission **Aging Analysis**: Breakdown of outstanding dues by time overdue (0-15 days, 15-30 days, etc.) **Bank Reconciliation**: Process of matching bank statement credits with system payment records **Defaulter**: Student with outstanding fees overdue by more than 30 days **Due Date**: Last date to pay fees without late penalty **Early Payment Discount**: Discount offered for paying fees before due date **Fee**

Ledger: Student-wise record of all fee transactions (assigned, paid, outstanding) **Fee Structure**: Configuration of fee types and amounts for a specific class and academic year **Grace Period**: Number of days after due date before late fee is applied (typically 7 days) **Invoice**: Document detailing fees to be paid, due date, and payment instructions **Late Fee**: Penalty charged for paying fees after due date + grace period **Partial Payment**: Payment of less than full invoice amount, leaving outstanding balance **Payment Gateway**: Third-party service (Razorpay, PayU) that processes online payments **Pro-rated Fee**: Adjusted fee amount for mid-session admissions (proportional to remaining months) **Receipt**: Document confirming payment received, with transaction details **Reconciliation**: Process of verifying that payment gateway/bank records match system records **Settlement**: Transfer of collected amount from payment gateway to school bank account (T+1/T+2 days) **Sibling Discount**: Discount offered to families with multiple children in same school **Transaction ID**: Unique identifier for each payment transaction **Tuition Fee**: Primary academic fee (largest component, typically 60-70% of total fees) **UPI**: Unified Payments Interface (Indian instant payment system - Google Pay, PhonePe, etc.) **Waiver**: Partial or complete exemption from fee payment (scholarship, financial hardship) **Webhook**: Automated notification sent by payment gateway to system when payment status changes

---

Appendix B: Payment Gateway Comparison

Feature	Razorpay	PayU	Paytm
Transaction Fee	2% + GST	2% + GST	1.99% + GST
Setup Fee	₹0	₹0	₹0
Settlement	T+1	T+2	T+1
UPI Support	✓	✓	✓
Card Support	✓	✓	✓
Net Banking	50+ banks	58+ banks	55+ banks
Wallets	Multiple	Multiple	Paytm only
Refund API	✓	✓	✓
Webhook	✓	✓	✓
Mobile SDK	✓	✓	✓
Dashboard	Excellent	Good	Good
Support	24/7	Business hours	Business hours
Recommendation	**Primary**	Backup	Backup

Recommended: Razorpay (primary) + PayU (backup)

Appendix C: Sample Email Templates

Invoice Generation Email

```
Subject: Fee Invoice for [Academic Year] - [Student Name]

Dear Parent,

Greetings from [School Name]!

We are pleased to share the fee invoice for [Student Name] (Class [X]-[Section])
for [Term 1/Term 2/Term 3] of academic year [2025-26].

Invoice Details:
- Invoice Number: FC/2025-26/001234
- Total Amount: ₹24,833
- Due Date: July 15, 2025

Fee Breakup:
- Tuition: ₹20,000
- Exam: ₹2,500
- Library: ₹1,000
- Sports: ₹1,333

You can pay online using:
- UPI (Google Pay, PhonePe, Paytm)
- Credit/Debit Card
- Net Banking

[Pay Now Button]

Or pay offline at school office (9 AM - 3 PM, Monday to Friday).

Please find the detailed invoice attached.

For any queries, contact us at finance@schoolname.com or call +91-XXXXXXXXXX.

Thank you for your cooperation.

Warm regards,
Finance Team
[School Name]
```

Payment Success Email

```
Subject: Payment Successful - Receipt [Receipt Number]

Dear Parent,

Thank you for your payment!

Your payment of ₹24,833 for [Student Name] (Class [X]-[Section]) has been
successfully received.
```

```
Payment Details:
- Receipt Number: REC/2025-26/005678
- Amount Paid: ₹24,833
- Payment Method: UPI (Google Pay)
- Transaction ID: rzp_Kj8d93jKs72j
- Date & Time: July 1, 2025, 8:45 PM

Outstanding Balance: ₹0 (Fully paid)

Please find your official receipt attached.

You can also download your receipt anytime from the parent portal: [Link]

Thank you for your timely payment!

Warm regards,
Finance Team
[School Name]
```

Payment Reminder (Pre-due)

```
Subject: Reminder: Fee Payment Due in 7 Days

Dear Parent,

This is a friendly reminder that the fee payment for [Student Name] (Class [X]-[Section]) is due on July 15, 2025 (7 days from now).

Amount Due: ₹24,833

Pay now to avoid late fees: [Pay Now Link]

If you have already paid, please ignore this message.

For any queries, contact us at finance@schoolname.com.

Thank you!

Finance Team
[School Name]
```

Appendix D: Sample SMS Templates

Invoice Generated:

```
[School Name]: Fee invoice FC/2025-26/001234 generated for [Student Name]. Amount: ₹24,833. Due: 15-Jul-2025. Pay now: [short link]
```

Payment Successful:

```
[School Name]: Payment successful! Receipt: REC/2025-26/005678. Amount: ₹24,833.
Download: [short link]. Thank you!
```

Payment Reminder (Pre-due):

```
[School Name]: Reminder - Fee ₹24,833 due on 15-Jul-2025 (7 days). Pay now: [short
link]
```

Payment Overdue:

```
[School Name]: Fee ₹24,833 is overdue. Late fee ₹500 applied. Total: ₹25,333. Pay
now: [short link]
```

Payment Failed:

```
[School Name]: Payment failed. Please retry: [short link]. For help, call [phone].
```

Appendix E: Parent FAQs

Q1: How do I pay fees online? A: Log into parent portal → View invoice → Click "Pay Now" → Select payment method (UPI/Card/Net Banking) → Complete payment. Receipt will be sent via SMS/Email.

Q2: Is online payment safe? A: Yes, 100% safe. We use Razorpay (PCI DSS Level 1 certified). Your card/bank details are never stored on our system.

Q3: What payment methods are supported? A: UPI (Google Pay, PhonePe, Paytm), Credit/Debit Cards, Net Banking (50+ banks), and offline cash/cheque.

Q4: When will I receive the receipt? A: Instantly (within 30 seconds) after successful payment via SMS and Email.

Q5: I paid but haven't received receipt. What to do? A: Check your email spam folder. Or log into parent portal and download receipt from "Payment History" section. For further help, contact finance@schoolname.com.

Q6: Can I pay fees for multiple children together? A: Yes! When you click "Pay Now", the system will show option to pay for all your children in one transaction.

Q7: What if I can't pay full amount now? A: You can make partial payment. The system will adjust your outstanding balance. For installment plan, contact finance office.

Q8: When is the last date to pay without late fee? A: Due date + 7 days grace period. For example, if due date is July 15, you can pay till July 22 without late fee.

Q9: How much is the late fee? A: 2% per month after grace period, up to maximum 10% of fee amount.

Q10: I need fee waiver due to financial hardship. How to apply? A: Log into parent portal → Waivers → Submit Request. Upload supporting documents. Finance team will review and principal will approve/reject.

End of Product Requirements Document

Document Metadata:

- **Document Type**: Product Requirements Document (PRD)
- **Journey**: Journey 2 - Fee Collection & Reconciliation
- **Project**: Sparked EdTech ERP + SIS + LMS
- **Version**: 1.0
- **Status**: Ready for Development
- **Created**: October 13, 2025
- **Author**: AI Product Manager (Claude)
- **Approvals**: Pending (Development Team, Finance Admin, Principal)
- **Next Steps**:
    1. Review PRD with stakeholders
    2. Create technical specification document
    3. Design database schema
    4. Create API documentation
    5. Begin Phase 1 development