Software Requirements Specification

Campus Digital Print & Xerox Service Platform

Version 1.0

Prepared by:
Software Engineering Team
Department of Computer Science

Date: December 14, 2025

Revision History

Version	Date	Author	Description
1.0	December 14, 2025	Software Engineering Team	Initial SRS document creation

Table of Contents

1. Introduction
    1. Purpose
    2. Scope
    3. Definitions, Acronyms, and Abbreviations
    4. References
2. Overall Description
    1. Product Perspective
    2. Product Functions
    3. User Classes and Characteristics
    4. Operating Environment
    5. Design Constraints
    6. Assumptions and Dependencies
3. System Features
    1. Student Module
    2. Print Shop Owner Module
    3. Administrator Module
4. Functional Requirements
5. Non-Functional Requirements
    1. Performance Requirements
    2. Security Requirements
    3. Scalability Requirements
    4. Reliability Requirements
    5. Usability Requirements
6. External Interface Requirements
    1. User Interface Requirements
    2. Hardware Interface Requirements
    3. Software Interface Requirements
    4. Communication Interface Requirements
7. System Architecture
    1. Logical Architecture
    2. Technology Stack
8. Data Design
    1. Entity Descriptions
    2. Relationships
9. System Diagrams
    1. System Architecture Diagram
    2. Use Case Diagram
    3. Data Flow Diagrams
    4. Sequence Diagram
10. Security Requirements
11. Testing Strategy
    1. Unit Testing
    2. Integration Testing
    3. System Testing
    4. User Acceptance Testing

12. Deployment Plan
13. Maintenance and Future Enhancements
14. Conclusion

1. Introduction

1.1 Purpose

This Software Requirements Specification document provides a comprehensive description of the Campus Digital Print & Xerox Service Platform. The document specifies functional and non-functional requirements, system architecture, and design constraints for the platform that aims to digitize and streamline printing and xerox services across campus facilities. This SRS is intended for developers, system architects, quality assurance teams, project managers, and academic evaluators.

1.2 Scope

The Campus Digital Print & Xerox Service Platform is a web-based application designed to eliminate physical queues at campus print shops by enabling students to upload documents, select printing options, make payments online, and track order status digitally. Print shop owners can register their businesses, configure service offerings and pricing, receive orders electronically, and manage order fulfillment through a dedicated interface. The system administrator oversees platform operations, manages user accounts, monitors transactions, and resolves disputes.

The platform will initially launch as a web application built with React frontend technology, with plans for mobile application development in subsequent phases. Key benefits include reduced wait times, improved operational efficiency, transparent pricing, real-time order tracking, and enhanced service accessibility across campus facilities.

The system does NOT include physical printing equipment integration, automated document processing capabilities, or offline functionality.

1.3 Definitions, Acronyms, and Abbreviations

SRS - Software Requirements Specification
API - Application Programming Interface
UI - User Interface
UX - User Experience
DBMS - Database Management System
REST - Representational State Transfer
JWT - JSON Web Token
HTTPS - Hypertext Transfer Protocol Secure
PDF - Portable Document Format
B&W - Black and White
OTP - One-Time Password
SMS - Short Message Service
FR - Functional Requirement
NFR - Non-Functional Requirement

Student - Registered user who places print/xerox orders
Print Shop Owner - Registered vendor who fulfills print orders

Administrator - System manager with elevated privileges
Order - A service request containing documents and printing specifications
Queue - Digital waiting list of pending orders

1.4 References

- IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications
- ISO/IEC 25010:2011, Systems and software Quality Requirements and Evaluation
- OWASP Top Ten Security Risks (2021)
- React Documentation v18, https://react.dev
- Node.js Documentation v20 LTS, https://nodejs.org
- PostgreSQL Documentation v15, https://www.postgresql.org/docs

---

2. Overall Description

2.1 Product Perspective

The Campus Digital Print & Xerox Service Platform is a new, self-contained system designed to operate within the campus ecosystem. It interfaces with external payment gateways for transaction processing and cloud storage services for document management. The system operates independently but can integrate with existing campus authentication systems in future iterations.

The platform consists of three primary interfaces: a student-facing web application for order placement and tracking, a shop owner dashboard for order management and business operations, and an administrative panel for system oversight and dispute resolution. The system architecture follows a client-server model with RESTful API communication between frontend and backend components.

2.2 Product Functions

The major functions of the system include:

For Students:

- User registration and authentication with campus credentials
- Document upload supporting multiple file formats (PDF, DOC, DOCX, images)
- Print shop discovery based on location and availability
- Customizable print options including paper size, orientation, color mode, binding, and quantity
- Secure online payment processing
- Real-time order status tracking with notifications
- Order history and receipt management
- Rating and review system for print shops

For Print Shop Owners:

- Business registration with verification process
- Service catalog configuration including pricing and capabilities
- Order queue management with priority handling
- Document download with secure access controls
- Order status updates and completion workflows
- Revenue tracking and financial reports
- Customer communication tools
- Inventory management alerts

For Administrators:

- User account management and verification
- Print shop approval and monitoring
- Transaction oversight and dispute resolution
- Platform analytics and reporting
- System configuration and maintenance tools
- Content moderation for reviews and ratings

2.3 User Classes and Characteristics

Students (Primary Users)

- Technical proficiency: Moderate to high
- Frequency of use: Regular (weekly or more frequent)
- Access level: Basic user privileges
- Primary goals: Quick order placement, affordable pricing, reliable service
- Characteristics: Tech-savvy, time-constrained, price-sensitive, mobile-first users

Print Shop Owners (Secondary Users)

- Technical proficiency: Moderate
- Frequency of use: Daily operations
- Access level: Vendor privileges with business management capabilities
- Primary goals: Efficient order processing, revenue maximization, customer satisfaction
- Characteristics: Business-oriented, quality-focused, service-driven, efficiency-minded

System Administrators (Tertiary Users)

- Technical proficiency: High
- Frequency of use: Daily monitoring and periodic intervention
- Access level: Full system privileges
- Primary goals: Platform stability, user satisfaction, dispute resolution, business growth
- Characteristics: Technical expertise, problem-solving orientation, policy enforcement

2.4 Operating Environment

Client Environment:

- Web browsers: Chrome 90+, Firefox 88+, Safari 14+, Edge 90+
- Screen resolutions: 1366x768 minimum, responsive design up to 4K displays
- Internet connectivity: Minimum 2 Mbps for standard operations
- JavaScript enabled with modern ES6+ support

Server Environment:

- Operating System: Linux Ubuntu 20.04 LTS or higher
- Web Server: Node.js v20 LTS with Express.js framework
- Database Server: PostgreSQL 15 or Supabase managed database
- Cloud Storage: AWS S3, Google Cloud Storage, or equivalent
- Payment Gateway: Integration with Razorpay, Stripe, or similar services

Network Environment:

- Campus network infrastructure with secure internet connectivity
- HTTPS/TLS 1.3 encryption for all communications
- Load balancing for high availability
- Content Delivery Network for static assets

2.5 Design Constraints

Technical Constraints:

- Must use React framework for frontend development to ensure component reusability and maintainability
- Backend API must be RESTful and stateless to support future mobile application
- Database must support ACID transactions for financial integrity
- Maximum file upload size limited to 50MB per document
- Must support concurrent users with minimal performance degradation

Regulatory Constraints:

- Compliance with data protection regulations for student information
- Adherence to campus IT security policies
- Financial transaction compliance with payment gateway requirements
- Copyright and intellectual property considerations for uploaded documents

Business Constraints:

- Development timeline: 6 months for MVP release
- Budget constraints limiting third-party service usage
- Campus infrastructure dependencies
- Existing print shop business model considerations

2.6 Assumptions and Dependencies

Assumptions:

- Users have access to reliable internet connectivity on campus
- Print shops have computers capable of accessing web applications
- Students possess valid campus credentials for registration
- Print shops maintain adequate supplies and equipment
- Payment gateway services remain operational and accessible
- Cloud storage services provide 99.9% uptime

Dependencies:

- Availability of campus network infrastructure
- Third-party payment gateway API stability
- Cloud storage service reliability
- Email and SMS notification service providers
- Browser compatibility and updates
- Database management system performance
- Campus administrative support for user verification

---

3. System Features

3.1 Student Module

3.1.1 User Registration and Authentication

Description: Students register using campus email credentials and authenticate for subsequent logins. The system supports password-based authentication with OTP verification for enhanced security.

Inputs:

- Full name, campus email address, student ID number
- Password (minimum 8 characters with complexity requirements)
- Mobile number for OTP verification
- Campus/department affiliation

Outputs:

- Confirmation email with verification link
- OTP sent to registered mobile number
- Success/failure message with appropriate error handling
- User profile creation in database

Priority: High

Functional Details:

- Email domain validation to ensure campus affiliation
- Password strength enforcement with real-time feedback
- OTP expiration after 10 minutes
- Account activation required before first login
- Session management with JWT tokens

3.1.2 Document Upload and Management

Description: Students upload documents in supported formats for printing. The system validates file types, scans for security threats, and stores documents securely until order completion.

Inputs:

- Document files (PDF, DOC, DOCX, JPG, PNG, TIFF)
- Multiple file selection supported
- File size validation (maximum 50MB per file)
- Optional document preview before upload

Outputs:

- Upload progress indicator
- File validation results
- Document thumbnail generation
- Unique document identifier
- Storage confirmation

Priority: High

Functional Details:

- Virus scanning on all uploaded files
- Automatic PDF conversion for non-PDF documents
- Page count detection and validation
- Temporary storage with 48-hour retention policy
- Secure deletion after order fulfillment

3.1.3 Print Shop Discovery and Selection

Description: Students browse available print shops based on location, ratings, pricing, and current queue status. The system displays relevant information to facilitate informed decision-making.

Inputs:

- Location preference (building name, department, or coordinates)
- Service requirements (color printing, binding, lamination)
- Urgency level (standard, express)
- Budget constraints

Outputs:

- List of matching print shops with distance calculation
- Real-time availability status
- Estimated wait time based on current queue
- Pricing comparison for selected options
- Shop ratings and recent reviews

Priority: High

Functional Details:

- Geolocation-based sorting
- Filter options for services and amenities
- Shop operating hours display
- Capacity indicators showing queue length
- Favorite shop bookmarking

3.1.4 Print Options Configuration

Description: Students customize printing specifications including paper size, color mode, orientation, binding, and quantity. The system calculates pricing dynamically based on selections.

Inputs:

- Paper size (A4, A3, Letter, Legal)
- Color mode (Black & White, Color)
- Print sides (Single-sided, Double-sided)
- Orientation (Portrait, Landscape)
- Number of copies
- Page range selection (all pages, specific pages)
- Binding options (None, Staple, Spiral, Perfect)
- Special instructions (text field, 500 characters max)

Outputs:

- Real-time price calculation with breakdown
- Page count validation
- Estimated completion time
- Preview of selected options
- Warning messages for unusual configurations

Priority: High

Functional Details:

- Dynamic pricing engine based on shop configuration
- Validation rules for invalid option combinations
- Default settings based on document characteristics

- Cost optimization suggestions
- Bulk discount calculations

3.1.5 Payment Processing

Description: Students complete payment through integrated payment gateway supporting multiple payment methods. The system ensures transaction security and provides instant confirmation.

Inputs:

- Payment method selection (Credit/Debit card, UPI, Net Banking, Campus Wallet)
- Payment credentials through secure gateway
- Billing information
- Promo code or discount coupon (optional)

Outputs:

- Payment gateway redirect
- Transaction success/failure notification
- Digital receipt with transaction ID
- Order confirmation with tracking number
- Email receipt delivery

Priority: High

Functional Details:

- PCI DSS compliant payment processing
- Transaction encryption with TLS 1.3
- Payment status webhooks for real-time updates
- Refund processing for cancelled orders
- Transaction history tracking

3.1.6 Order Tracking and Notifications

Description: Students track order status in real-time and receive notifications at key milestones. The system provides transparency throughout the fulfillment process.

Inputs:

- Order tracking number or selection from order history
- Notification preferences (Email, SMS, Push)

Outputs:

- Current order status display
- Status change notifications (Order Received, In Progress, Ready for Pickup, Completed)
- Estimated completion time updates

- Shop contact information
- Order cancellation option (if applicable)

Priority: High

Functional Details:

- Real-time status updates via WebSocket connection
- Push notifications for mobile web
- Email and SMS alerts at status changes
- Order timeline visualization
- Cancellation window (within 5 minutes of placement)

3.1.7 Order History and Reviews

Description: Students access complete order history, download receipts, and provide ratings and reviews for completed orders.

Inputs:

- Date range filter for order history
- Search by order number or shop name
- Rating (1-5 stars)
- Written review (optional, 1000 characters max)
- Service quality feedback

Outputs:

- Paginated list of past orders
- PDF receipt download
- Order replication capability
- Review submission confirmation
- Review moderation status

Priority: Medium

Functional Details:

- Order history retention for academic year duration
- Downloadable receipts in PDF format
- One review per order limitation
- Review editing within 24 hours
- Inappropriate content flagging

3.2 Print Shop Owner Module

3.2.1 Business Registration and Verification

Description: Print shop owners register their businesses, provide necessary documentation, and undergo administrative verification before activation.

Inputs:

- Business name and registration details
- Owner name and contact information
- Shop location (building, floor, room number)
- Business license or campus vendor authorization
- Bank account details for payments
- Operating hours

Outputs:

- Registration confirmation
- Verification pending status
- Admin approval notification
- Account activation email
- Onboarding instructions

Priority: High

Functional Details:

- Document upload for verification
- Campus location validation
- Duplicate shop prevention
- Approval workflow with admin review
- Rejection with reason provision

3.2.2 Service Catalog Configuration

Description: Shop owners configure available services, pricing structures, paper types, and binding options offered at their establishment.

Inputs:

- Service availability toggles (B&W printing, Color printing, Binding, Lamination, Scanning)
- Pricing per page for each configuration
- Paper types and sizes available
- Binding options with additional charges
- Bulk discount rules
- Minimum order values
- Service surcharges (express service, off-hours)

Outputs:

- Service catalog published to system
- Pricing validation results
- Preview of customer-facing display
- Change history log

Priority: High

Functional Details:

- Flexible pricing models (per page, per sheet, fixed rate)
- Template pricing configurations
- Seasonal pricing adjustments
- Automatic price calculation verification
- Service availability scheduling

3.2.3 Order Queue Management

Description: Shop owners view incoming orders in a prioritized queue, accept or reject orders based on capacity, and manage workflow efficiently.

Inputs:

- Order acceptance/rejection decision
- Priority adjustment for express orders
- Estimated completion time input
- Rejection reason (if declining order)

Outputs:

- Real-time order queue display
- Order notification alerts (audio/visual)
- Queue statistics (pending, in-progress, completed)
- Workload indicators
- Order details with document access

Priority: High

Functional Details:

- Auto-sort by order time and urgency
- Capacity-based order limiting
- Pause/resume order acceptance
- Order bundling for efficiency
- Queue time analytics

3.2.4 Document Access and Processing

Description: Shop owners securely download customer documents for printing with access controls and audit trails.

Inputs:

- Order selection for document retrieval
- Download confirmation acknowledgment

Outputs:

- Secure document download link (time-limited)
- Document preview capability
- Download confirmation log
- Print specifications summary
- Page count validation

Priority: High

Functional Details:

- Encrypted document transmission
- Access expiration after 24 hours
- Download attempt logging
- Watermarked preview to prevent misuse
- Automatic file deletion post-completion

3.2.5 Order Status Management

Description: Shop owners update order status through workflow stages, communicate with customers, and mark orders complete upon pickup.

Inputs:

- Status update selection (Accepted, In Progress, Ready, Completed)
- Optional message to customer
- Actual page count (if different from estimated)
- Quality issues or delays notification
- Completion confirmation

Outputs:

- Status update confirmation
- Customer notification trigger
- Timeline update in system
- Inventory adjustment (if applicable)
- Completion timestamp

Priority: High

Functional Details:

- Status workflow validation
- Bulk status updates for multiple orders
- Delayed completion alerts
- Customer communication log
- Mandatory completion notes

3.2.6 Revenue and Analytics Dashboard

Description: Shop owners access financial summaries, order analytics, and performance metrics to manage business operations effectively.

Inputs:

- Date range selection
- Report type (Revenue, Orders, Popular Services)
- Export format preference

Outputs:

- Revenue summary with breakdowns
- Order volume statistics
- Service popularity rankings
- Peak hours analysis
- Customer ratings overview
- Settlement schedule information
- Downloadable reports (PDF, CSV)

Priority: Medium

Functional Details:

- Real-time revenue tracking
- Settlement reconciliation
- Tax calculation assistance
- Trend visualization with charts
- Comparative period analysis

3.2.7 Customer Communication

Description: Shop owners contact customers regarding order issues, clarifications, or special requests through integrated messaging.

Inputs:

- Customer selection from order details
- Message content
- Attachment support for clarifications

Outputs:

- Message delivery confirmation
- Customer response notification
- Communication history log

Priority: Medium

Functional Details:

- In-app messaging system
- Email fallback notification
- Response time tracking
- Conversation threading
- Communication templates

3.3 Administrator Module

3.3.1 User Management

Description: Administrators manage student and shop owner accounts, verify credentials, handle account issues, and enforce platform policies.

Inputs:

- User search criteria (email, ID, name)
- Account action (Approve, Suspend, Delete, Verify)
- Action reason and notes

Outputs:

- User account list with status
- Account modification confirmation
- Notification to affected user
- Audit log entry

Priority: High

Functional Details:

- Bulk account operations
- Account recovery assistance
- Credential reset capability
- Suspicious activity flagging
- User role management

3.3.2 Print Shop Verification and Monitoring

Description: Administrators review print shop registration applications, verify business credentials, monitor shop performance, and enforce quality standards.

Inputs:

- Shop application review
- Verification decision (Approve/Reject)
- Performance threshold configurations
- Warning or suspension actions

Outputs:

- Approval/rejection notification to shop
- Shop listing activation
- Performance report generation
- Compliance status updates

Priority: High

Functional Details:

- Document verification workflow
- Campus location validation
- Performance metrics monitoring
- Complaint tracking per shop
- Rating threshold enforcement

3.3.3 Transaction Oversight

Description: Administrators monitor all financial transactions, investigate discrepancies, process refunds, and maintain financial integrity.

Inputs:

- Transaction search filters
- Dispute investigation details
- Refund authorization
- Settlement approval

Outputs:

- Transaction listing with details
- Dispute resolution logs
- Refund confirmation
- Financial reconciliation reports

Priority: High

Functional Details:

- Real-time transaction monitoring
- Anomaly detection algorithms
- Refund processing workflow
- Payment gateway reconciliation
- Revenue tracking by shop

3.3.4 Platform Analytics and Reporting

Description: Administrators access comprehensive analytics covering system usage, user engagement, revenue trends, and operational efficiency.

Inputs:

- Report type selection
- Date range parameters
- Granularity level (daily, weekly, monthly)
- Export format

Outputs:

- Interactive dashboards
- Trend visualizations
- Key performance indicators
- Comparative analyses
- Downloadable reports

Priority: Medium

Functional Details:

- Real-time KPI tracking
- User growth metrics
- Order volume trends
- Revenue analytics
- Shop performance rankings
- Peak usage analysis

3.3.5 Content Moderation

Description: Administrators review user-generated content including reviews, ratings, and profile information to ensure compliance with community guidelines.

Inputs:

- Flagged content queue
- Moderation decision (Approve, Remove, Warn User)
- Policy violation category

Outputs:

- Content approval/removal action
- User notification of violation
- Content moderation log
- Policy update recommendations

Priority: Medium

Functional Details:

- Automated profanity filtering
- User reporting system
- Moderation queue prioritization
- Appeal process management

- Policy enforcement tracking

3.3.6 System Configuration

Description: Administrators configure system-wide settings including business rules, pricing parameters, notification templates, and operational parameters.

Inputs:

- Configuration parameter selection
- New value assignment
- Change justification
- Effective date/time

Outputs:

- Configuration update confirmation
- System restart notification (if required)
- Change log documentation
- Rollback capability

Priority: Medium

Functional Details:

- Environment-specific configurations
- Feature flag management
- Maintenance mode control
- Service level adjustments
- Integration settings management

---

4. Functional Requirements

FR-1: User Authentication and Authorization

The system shall authenticate users based on role (student, shop owner, administrator) and enforce role-based access control throughout the application.

FR-2: Student Registration

The system shall allow students to register using campus email addresses with OTP verification and maintain profile information securely.

FR-3: Document Upload

The system shall accept document uploads in PDF, DOC, DOCX, JPG, PNG, and TIFF formats up to 50MB per file with virus scanning capability.

FR-4: Print Shop Discovery

The system shall enable students to search and filter print shops based on location, availability, services offered, ratings, and pricing.

FR-5: Print Configuration

The system shall allow students to configure print options including paper size, color mode, orientation, binding, page range, and quantity.

FR-6: Dynamic Price Calculation

The system shall calculate order prices in real-time based on selected options, shop pricing, and applicable discounts.

FR-7: Payment Processing

The system shall integrate with payment gateway APIs to process transactions securely and provide instant confirmation.

FR-8: Order Creation

The system shall generate unique order identifiers and store order details including documents, specifications, pricing, and timestamps.

FR-9: Order Notification

The system shall send notifications to shop owners immediately upon order placement through multiple channels.

FR-10: Order Queue Management

The system shall present orders to shop owners in a prioritized queue with acceptance/rejection capabilities.

FR-11: Document Security

The system shall provide time-limited, encrypted document access to shop owners only for accepted orders.

FR-12: Status Updates

The system shall allow shop owners to update order status through predefined workflow stages with customer notifications.

FR-13: Order Tracking

The system shall enable students to track order status in real-time with estimated completion time updates.

FR-14: Order Completion

The system shall mark orders complete upon shop owner confirmation and trigger final notifications to students.

FR-15: Order History

The system shall maintain order history for students with search, filter, and receipt download capabilities.

FR-16: Rating and Review

The system shall allow students to rate and review completed orders with text feedback and star ratings.

FR-17: Shop Registration

The system shall enable print shop owners to register businesses with verification documents pending admin approval.

FR-18: Service Configuration

The system shall allow shop owners to configure available services, pricing structures, and operational parameters.

FR-19: Revenue Dashboard

The system shall provide shop owners with revenue analytics, order statistics, and financial reports.

FR-20: Admin User Management

The system shall enable administrators to manage user accounts, verify credentials, and enforce policies.

FR-21: Admin Shop Verification

The system shall provide administrators with tools to review, approve, or reject print shop registration applications.

FR-22: Admin Transaction Monitoring

The system shall allow administrators to monitor all transactions, investigate disputes, and process refunds.

FR-23: Platform Analytics

The system shall generate comprehensive analytics reports covering usage, engagement, revenue, and performance metrics.

FR-24: Content Moderation

The system shall enable administrators to review flagged content and enforce community guidelines.

FR-25: System Configuration

The system shall provide administrators with tools to configure system-wide parameters and business rules.

FR-26: Notification System

The system shall deliver notifications via email, SMS, and in-app channels based on user preferences.

FR-27: Search Functionality

The system shall provide search capabilities across orders, shops, users, and transactions with relevant filters.

FR-28: Data Export

The system shall enable users to export data (orders, receipts, reports) in PDF and CSV formats.

FR-29: Session Management

The system shall manage user sessions with configurable timeout periods and secure token handling.

FR-30: Audit Logging

The system shall maintain audit logs for critical operations including authentication, transactions, and administrative actions.

---

5. Non-Functional Requirements

5.1 Performance Requirements

NFR-1: Response Time The system shall respond to user interactions within 2 seconds under normal load conditions for 95% of requests.

NFR-2: Page Load Time Web pages shall load completely within 3 seconds on standard broadband connections (minimum 2 Mbps).

NFR-3: Document Upload Speed The system shall support document uploads at speeds limited only by network bandwidth with progress indication.

NFR-4: Concurrent Users The system shall support at least 500 concurrent users without performance degradation exceeding 10%.

NFR-5: Database Query Performance Database queries shall execute within 500 milliseconds for 99% of operations.

NFR-6: API Response Time Backend API endpoints shall respond within 1 second for 98% of requests under normal load.

NFR-7: Search Performance Search operations shall return results within 1.5 seconds regardless of dataset size.

NFR-8: Notification Delivery Push notifications shall be delivered within 5 seconds of triggering event.

5.2 Security Requirements

NFR-9: Data Encryption All data transmission shall use TLS 1.3 or higher encryption protocols.

NFR-10: Password Security User passwords shall be hashed using bcrypt or equivalent with minimum salt rounds of 12.

NFR-11: Authentication Token Security JWT tokens shall expire after 24 hours for students and 12 hours for shop owners with refresh token rotation.

NFR-12: Document Security Uploaded documents shall be encrypted at rest using AES-256 encryption.

NFR-13: Payment Security Payment processing shall comply with PCI DSS standards with no storage of card details on application servers.

NFR-14: Access Control The system shall enforce role-based access control preventing unauthorized access to resources.

NFR-15: Input Validation All user inputs shall be validated and sanitized to prevent SQL injection, XSS, and CSRF attacks.

NFR-16: API Security API endpoints shall implement rate limiting, authentication, and authorization checks on every request.

5.3 Scalability Requirements

NFR-17: Horizontal Scalability The system architecture shall support horizontal scaling by adding additional server instances.

NFR-18: Database Scalability Database design shall accommodate growth to 100,000 users and 1 million orders without structural changes.

NFR-19: Storage Scalability Cloud storage integration shall support unlimited document storage with automatic scaling.

NFR-20: Load Balancing The system shall implement load balancing to distribute traffic across multiple server instances.

5.4 Reliability Requirements

NFR-21: System Availability The system shall maintain 99.5% uptime during operational hours (excluding scheduled maintenance).

NFR-22: Data Backup Automated database backups shall occur daily with retention period of 30 days.

NFR-23: Disaster Recovery The system shall support recovery point objective (RPO) of 24 hours and recovery time objective (RTO) of 4 hours.

NFR-24: Transaction Integrity Payment transactions shall be atomic with rollback capability in case of failures.

NFR-25: Error Handling The system shall gracefully handle errors with user-friendly messages and detailed logging for debugging.

5.5 Usability Requirements

NFR-26: User Interface Consistency The user interface shall maintain consistent design patterns, terminology, and navigation across all modules.

NFR-27: Responsive Design The web application shall be fully functional on devices with screen sizes from 360px to 4K resolution.

NFR-28: Accessibility The system shall comply with WCAG 2.1 Level AA accessibility guidelines.

NFR-29: Browser Compatibility The application shall function correctly on Chrome, Firefox, Safari, and Edge browsers released within the past 2 years.

NFR-30: Learning Curve New users shall be able to complete their first order within 5 minutes without external assistance.

NFR-31: Help Documentation The system shall provide contextual help and comprehensive user documentation accessible from all pages.

NFR-32: Internationalization The system architecture shall support future internationalization with externalized text strings.

---

6. External Interface Requirements

6.1 User Interface Requirements

Student Interface: The student interface shall feature a clean, modern design with intuitive navigation. The homepage shall display nearby print shops, recent orders, and quick access to upload functionality. The document upload interface shall support drag-and-drop functionality with multiple file selection. Print configuration screens shall provide real-time price calculation with clear option labels. The order tracking interface shall display status progression visually with timeline representation.

Print Shop Owner Interface: The shop owner dashboard shall present key metrics prominently including pending orders count, today's revenue, and current queue status. The order queue shall display orders in card format with thumbnail previews, customer details, and action buttons. The service configuration interface shall provide intuitive forms with validation feedback. Revenue analytics shall be presented through interactive charts and graphs with drill-down capabilities.

Administrator Interface: The admin panel shall feature a comprehensive dashboard with system-wide metrics and alerts. User management interfaces shall support bulk operations with confirmation dialogs. Analytics views shall offer customizable date ranges and export functionality. All administrative actions shall require confirmation to prevent accidental modifications.

6.2 Hardware Interface Requirements

Client Hardware:

- Minimum processor: Dual-core 1.6 GHz or equivalent
- Minimum RAM: 4GB
- Storage: 500MB free space for cached data
- Display: Minimum 1366x768 resolution
- Input devices: Keyboard, mouse, or touchscreen
- Network adapter: Ethernet or WiFi capable

Server Hardware:

- Processor: Multi-core server-grade CPU (minimum 8 cores recommended)
- RAM: Minimum 16GB, recommended 32GB for production
- Storage: SSD with minimum 500GB capacity
- Network: Gigabit Ethernet interface
- Redundant power supply for production environments

Printing Equipment: No direct hardware integration required. Print shops use existing equipment independent of the system.

6.3 Software Interface Requirements

Frontend Framework:

- React v18 or higher for component-based UI development
- React Router for client-side routing
- Redux or Context API for state management
- Axios for HTTP client communication

Backend Framework:

- Node.js v20 LTS with Express.js framework
- RESTful API architecture with JSON data format
- JWT for stateless authentication
- Bcrypt for password hashing

Database System:

- PostgreSQL v15 or Supabase managed database
- Support for ACID transactions
- Connection pooling for performance optimization
- Query optimization for complex joins

Payment Gateway Integration:

- RESTful API integration with Razorpay, Stripe, or PayPal
- Webhook support for payment status notifications
- SDK for secure payment processing
- Test/sandbox environment for development

Cloud Storage Service:

- AWS S3, Google Cloud Storage, or Azure Blob Storage
- RESTful API for file operations
- Pre-signed URLs for secure document access
- Versioning and lifecycle management support

Notification Services:

- Email service provider (SendGrid, AWS SES, or SMTP)
- SMS gateway integration (Twilio, MSG91, or similar)
- Push notification service for web (Firebase Cloud Messaging or OneSignal)

Operating System:

- Linux Ubuntu 20.04 LTS or higher for server deployment
- Compatible with major cloud platforms (AWS, Google Cloud, Azure)

6.4 Communication Interface Requirements

HTTP/HTTPS Protocol: All communication between client and server shall use HTTPS protocol with TLS 1.3 encryption. API endpoints shall follow RESTful conventions using standard HTTP methods (GET, POST, PUT, DELETE, PATCH).

WebSocket Protocol: Real-time features including order status updates and notifications shall use WebSocket protocol with automatic reconnection handling.

Email Protocol: Email notifications shall be sent using SMTP protocol with SPF, DKIM, and DMARC authentication to ensure deliverability.

Network Requirements:

- Bandwidth: Minimum 100 Mbps for server connectivity
- Latency: Maximum 200ms acceptable for API responses
- Protocols: Support for IPv4 and IPv6
- Ports: HTTPS (443), WebSocket (443), SMTP (587)

7. System Architecture

7.1 Logical Architecture

The system follows a three-tier architecture pattern consisting of presentation layer, business logic layer, and data layer.

Presentation Layer: The presentation layer comprises React-based web applications providing user interfaces for students, shop owners, and administrators. This layer handles user interactions, input validation, and state management. Components communicate with the backend through RESTful API calls and WebSocket connections for real-time updates.

Business Logic Layer: The middle tier consists of Node.js/Express.js backend services implementing business rules, workflow orchestration, and integration logic. This layer includes authentication services, order management services, payment processing services, notification services, and file management services. Services are designed as loosely-coupled modules to support independent scaling and maintenance.

Data Layer: The data layer encompasses PostgreSQL database for structured data storage, cloud object storage for documents, and caching layer using Redis for performance optimization. This layer provides data persistence, retrieval, and integrity enforcement.

External Services Layer: Integration layer connects to external services including payment gateways, notification providers, and cloud storage through standardized API interfaces with error handling and retry mechanisms.

Architecture Characteristics:

- Stateless API design for horizontal scalability
- Microservices-ready architecture allowing future service decomposition
- Event-driven architecture for asynchronous operations
- Clear separation of concerns across layers
- API-first design supporting multiple client types

7.2 Technology Stack

Frontend Technologies:

- React 18.x - UI framework
- React Router 6.x - Client-side routing
- Redux Toolkit - State management
- Axios - HTTP client
- TailwindCSS - Utility-first CSS framework
- React Query - Server state management
- Socket.io-client - WebSocket client
- React Hook Form - Form handling
- Recharts - Data visualization

Backend Technologies:

- Node.js 20 LTS - Runtime environment
- Express.js 4.x - Web framework
- PostgreSQL 15 - Relational database
- Prisma/TypeORM - ORM layer
- JSON Web Token - Authentication
- Bcrypt - Password hashing
- Multer - File upload handling
- Socket.io - WebSocket server
- Bull - Job queue management
- Winston - Logging framework

Infrastructure and DevOps:

- Docker - Containerization
- Nginx - Reverse proxy and load balancing
- PM2 - Process management
- GitHub Actions - CI/CD pipeline
- AWS/GCP/Azure - Cloud hosting
- CloudFlare - CDN and DDoS protection

Third-Party Services:

- Razorpay/Stripe - Payment processing
- AWS S3/Google Cloud Storage - File storage
- SendGrid/AWS SES - Email delivery
- Twilio/MSG91 - SMS notifications
- Firebase Cloud Messaging - Push notifications

Development Tools:

- Visual Studio Code - IDE
- Postman - API testing
- pgAdmin - Database administration
- Git - Version control
- ESLint/Prettier - Code quality
- Jest/React Testing Library - Testing

---

8. Data Design

8.1 Entity Descriptions

User Entity: Stores information about all system users with role differentiation. Attributes include user ID (primary key), email, password hash, full name, mobile number, user role (student/shop_owner/admin), campus affiliation, verification status, registration timestamp, last login timestamp, and account status.

Student Entity: Extends user entity with student-specific attributes including student ID number, department, academic year, campus address, default payment method, notification preferences, and profile completion status.

PrintShop Entity: Represents registered print shops with attributes including shop ID (primary key), owner user ID (foreign key), shop name, location details (building, floor, room), contact information, business registration number, bank account details, operating hours, verification status, approval timestamp, rating average, total orders completed, and active status.

Service Entity: Defines services offered by print shops with attributes including service ID (primary key), shop ID (foreign key), service type (B&W print, color print, binding, lamination), base price, unit (per page/per sheet), availability status, and special conditions.

Order Entity: Central entity tracking all print orders with attributes including order ID (primary key), student ID (foreign key), shop ID (foreign key), order timestamp, total amount, payment status, order status (placed, accepted, in_progress, ready, completed, cancelled), estimated completion time, actual completion time, special instructions, and cancellation reason.

OrderItem Entity: Details individual documents within an order with attributes including item ID (primary key), order ID (foreign key), document reference, file name, page count, paper size, color mode, print sides, copies, binding type, item price, and processing notes.

Document Entity: Manages uploaded documents with attributes including document ID (primary key), user ID (foreign key), file path, file name, file size, file type, upload timestamp, virus scan status, page count, access expiry, and deletion status.

Transaction Entity: Records all financial transactions with attributes including transaction ID (primary key), order ID (foreign key), amount, payment method, payment gateway transaction ID, timestamp, status (pending, success, failed, refunded), gateway response, and settlement status.

Review Entity: Stores customer reviews and ratings with attributes including review ID (primary key), order ID (foreign key), student ID (foreign key), shop ID (foreign key), rating (1-5 stars), review text, timestamp, moderation status, helpful count, and response from shop owner.

Notification Entity: Manages notification delivery with attributes including notification ID (primary key), user ID (foreign key), notification type, title, message body, delivery channels (email, SMS, push), sent timestamp, read status, and related entity reference.

AuditLog Entity: Tracks critical system operations with attributes including log ID (primary key), user ID (foreign key), action type, entity affected, timestamp, IP address, user agent, request details, and response status.

8.2 Relationships

User to Student: One-to-One relationship where each student account extends a user account.

User to PrintShop: One-to-Many relationship where a user (shop owner) can own multiple print shops.

PrintShop to Service: One-to-Many relationship where each shop offers multiple services.

Student to Order: One-to-Many relationship where students place multiple orders over time.

PrintShop to Order: One-to-Many relationship where shops receive and fulfill multiple orders.

Order to OrderItem: One-to-Many relationship where each order contains one or more document items.

Order to Transaction: One-to-One or One-to-Many relationship tracking payment transactions for each order (supporting partial payments and refunds).

Document to OrderItem: One-to-Many relationship where a single uploaded document can be used in multiple orders.

Order to Review: One-to-One relationship where each completed order can have one review.

User to Notification: One-to-Many relationship where users receive multiple notifications.

User to AuditLog: One-to-Many relationship tracking all actions performed by users.

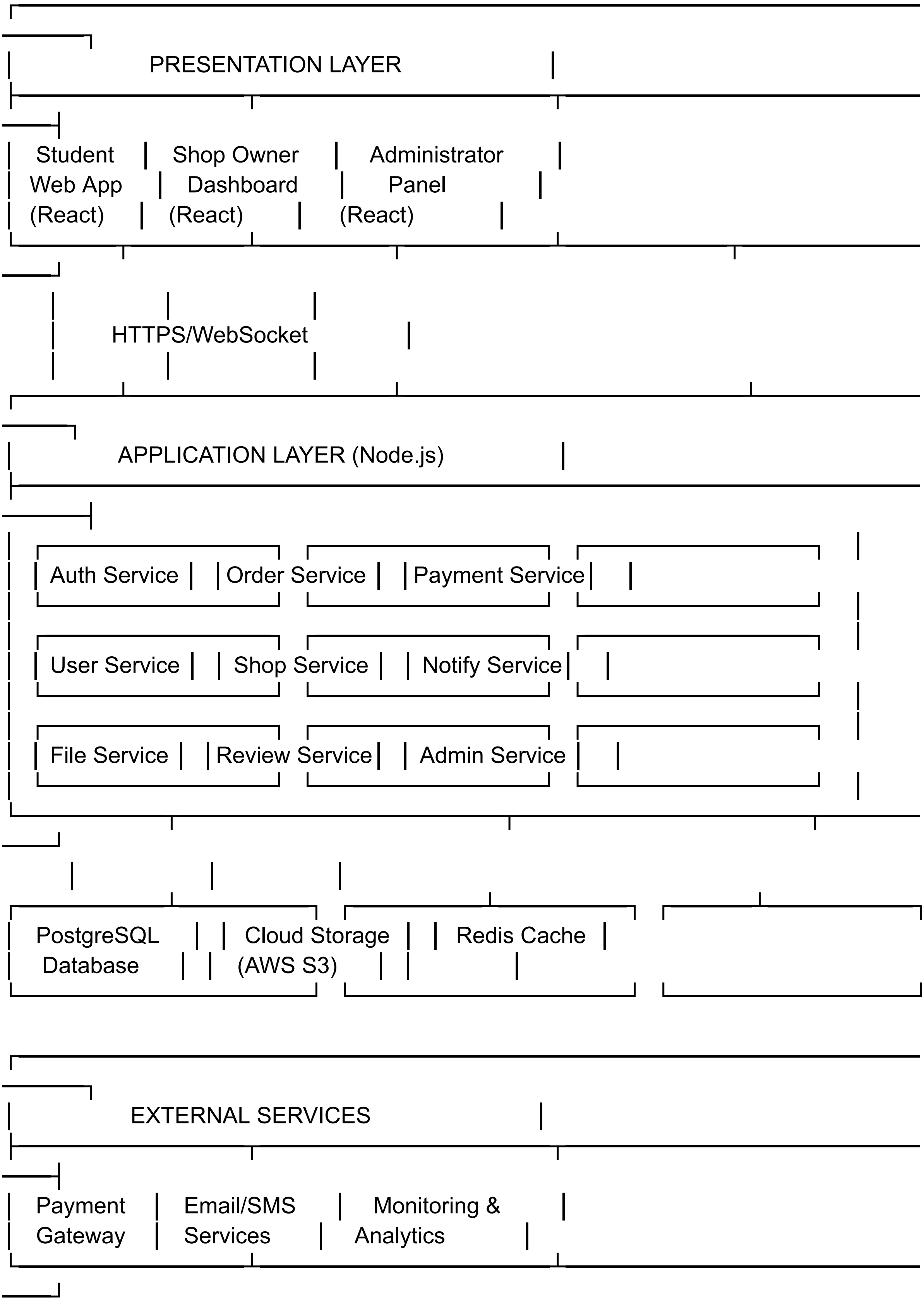
Referential Integrity: All foreign key relationships enforce referential integrity with appropriate cascade rules (CASCADE on update, RESTRICT on delete for critical entities, CASCADE on delete for dependent entities).

Indexing Strategy: Primary keys indexed by default. Additional indexes on frequently queried fields including user email, order status, shop location, document user ID, transaction order ID, and timestamp fields for time-based queries.

---

9. System Diagrams

9.1 System Architecture Diagram



PRESENTATION LAYER

Student Web App (React)

Shop Owner Dashboard (React)

Administrator Panel (React)

HTTPS/WebSocket

APPLICATION LAYER (Node.js)

Auth Service

Order Service

Payment Service

User Service

Shop Service

Notify Service

File Service

Review Service

Admin Service

PostgreSQL Database

Cloud Storage (AWS S3)

Redis Cache

EXTERNAL SERVICES

Payment Gateway

Email/SMS Services

Monitoring & Analytics

9.2 Use Case Diagram

```
            Campus Digital Print System
Campus Digital Print System

┌──────────────────────────────────────────────────────────
──┐
│
│  Student
│    │
│    ├────▶ Register/Login
│    ├────▶ Upload Documents
│    ├────▶ Browse Print Shops
│    ├────▶ Configure Print Options
│    ├────▶ Make Payment
│    ├────▶ Track Order Status
│    ├────▶ View Order History
│    └────▶ Rate and Review Shop
│
├──────────────────────────────────────────────────────────
──┤
│
│  Print Shop Owner
│    │
│    ├────▶ Register Business
│    ├────▶ Configure Services & Pricing
│    ├────▶ View Order Queue
│    ├────▶ Accept/Reject Orders
│    ├────▶ Download Documents
│    ├────▶ Update Order Status
│    ├────▶ View Revenue Dashboard
│    └────▶ Communicate with Customers
│
├──────────────────────────────────────────────────────────
──┤
│
│  Administrator
│    │
│    ├────▶ Manage User Accounts
│    ├────▶ Verify Print Shops
│    ├────▶ Monitor Transactions
│    ├────▶ Generate Reports
│    ├────▶ Moderate Reviews
│    ├────▶ Resolve Disputes
│    └────▶ Configure System Settings
│
```

```
│
──┘
```

External Actors:
- Payment Gateway (Processes Payments)
- Notification Service (Sends Alerts)
- Cloud Storage (Stores Documents)

External Actors:
- Payment Gateway (Processes Payments)
- Notification Service (Sends Alerts)
- Cloud Storage (Stores Documents)

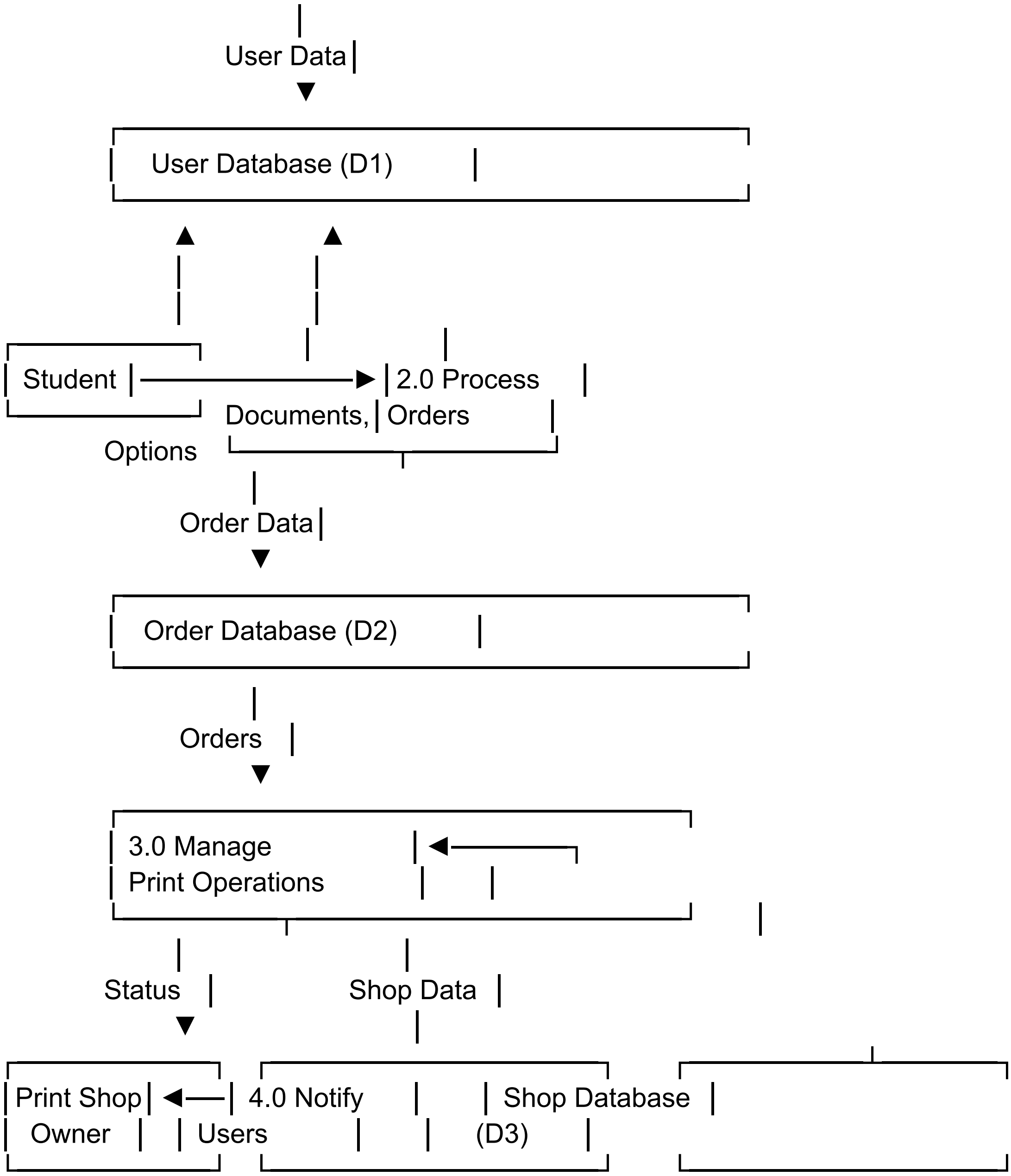
9.3 Data Flow Diagram

Level 0 - Context Diagram:

```
           ┌──────────────────┐
           │  Student  │
           └──────────┬───────┘
              │
     Documents,     │   Order Status,
     Order Requests   │   Notifications
              │
           ┌─────────▼──────────┐
 Shop Details,   │         │  Orders,
 Order Updates ───────►│ Campus Print  │◄──── Documents
           │  System    │
           │         │
           └──────────┬──────────┘
              │
     Orders,      │  Status Updates,
     Documents     │   Confirmations
              │
           ┌─────────▼───────┐
           │ Print Shop  │
           │  Owner   │
           └─────────────────┘
```

Level 1 - System Processes:

```
┌──────────┐           ┌─────────────────────┐
│ Student │──────────────────►│ 1.0 Manage    │
└──────────┘ Registration, │ User Accounts  │
      Login      └──────────┬──────────┘
```


User Data
User Database (D1)
Student
2.0 Process
Documents,
Orders
Options
Order Data
Order Database (D2)
Orders
3.0 Manage
Print Operations
Status
Shop Data
Print Shop
Owner
4.0 Notify
Users
Shop Database
(D3)

10. Security Requirements

Access Control: The system shall implement role-based access control (RBAC) with clearly defined permissions for each user role. Students shall access only their own data and public shop information. Shop owners shall access only their business data and assigned orders. Administrators shall have elevated privileges with audit trail logging for all actions.

Authentication Security: Multi-factor authentication shall be available as an optional security enhancement. Account lockout shall occur after 5 consecutive failed login attempts with automatic unlock after 30 minutes or manual admin intervention. Session tokens shall be securely generated, stored in HTTP-only cookies, and invalidated upon logout.

Data Protection: Personally identifiable information (PII) shall be encrypted at rest using AES-256 encryption. Database backups shall be encrypted before storage. Sensitive fields in logs shall be masked or redacted. Data retention policies shall ensure automatic deletion of expired documents and archival of old orders.

API Security: All API endpoints shall require authentication except public endpoints like shop listings. Rate limiting shall prevent abuse with limits of 100 requests per minute per user for authenticated endpoints. API keys for shop owners shall be securely generated and rotatable. Input validation shall prevent injection attacks across all endpoints.

Payment Security: The system shall never store complete credit card numbers or CVV codes. Payment processing shall occur entirely within PCI DSS compliant payment gateway iframes. Transaction logs shall store only masked card information and transaction references. Refund operations shall require admin approval with verification workflows.

Document Security: Uploaded documents shall be scanned for malware before storage. File access URLs shall be pre-signed with expiration times (24 hours for shop owners). Documents shall be accessible only to authorized parties with audit logging. Automatic deletion shall occur 7 days after order completion unless explicitly archived by users.

Network Security: All production traffic shall use HTTPS with valid SSL/TLS certificates. HTTP Strict Transport Security (HSTS) headers shall be implemented. Cross-Origin Resource Sharing (CORS) shall be configured to allow only trusted domains. DDoS protection shall be implemented through CDN or cloud provider services.

Monitoring and Incident Response: Security event logging shall capture authentication attempts, authorization failures, data access, and administrative actions. Automated alerts shall trigger for suspicious activities including multiple failed logins, unusual transaction patterns, and privilege escalation attempts. Incident response procedures shall be documented and tested quarterly.

11. Testing Strategy

11.1 Unit Testing

Unit testing shall verify individual components, functions, and modules in isolation. Developers shall write unit tests for all business logic functions, utility modules, API endpoint handlers, and React components. Testing frameworks include Jest for JavaScript/Node.js testing and React Testing Library for component testing.

Coverage Requirements: Minimum 80% code coverage for backend services and 70% for frontend components. Critical payment and authentication modules shall achieve 95% coverage. All utility functions shall have 100% coverage.

Test Categories: Positive test cases verifying expected behavior with valid inputs. Negative test cases ensuring proper error handling with invalid inputs. Boundary test cases checking edge conditions and limits. Mock testing for external dependencies including database, payment gateway, and storage services.

11.2 Integration Testing

Integration testing shall verify interactions between system components and external services. Tests shall validate API contract adherence, database transaction integrity, payment gateway integration, notification service delivery, and file storage operations.

API Integration Tests: Endpoint testing covering all REST API routes with various input combinations. Authentication flow testing ensuring proper token generation and validation. Authorization testing verifying role-based access controls. Error response validation checking appropriate status codes and messages.

Database Integration Tests: Transaction integrity testing ensuring rollback on failures. Concurrent access testing validating isolation levels. Foreign key constraint validation. Query performance testing under load conditions.

External Service Integration: Payment gateway testing in sandbox environment covering successful payments, failures, timeouts, and refunds. Email and SMS delivery testing with test accounts. Cloud storage testing for upload, download, and deletion operations. Webhook handling for asynchronous notifications.

11.3 System Testing

System testing shall validate end-to-end workflows across the entire application stack. Testing shall simulate real user scenarios from registration through order completion.

Functional System Tests: Complete student workflow from registration to order placement and tracking. Shop owner workflow from registration to order fulfillment and revenue viewing. Administrator workflow covering user management and dispute resolution. Cross-role workflows testing interactions between different user types.

Performance System Tests: Load testing with 500+ concurrent users simulating realistic usage patterns. Stress testing to identify breaking points and graceful degradation. Endurance testing running for 24-hour periods to detect memory leaks. Spike testing with sudden traffic increases.

Security System Tests: Penetration testing covering OWASP Top 10 vulnerabilities. Authentication bypass attempts. SQL injection and XSS vulnerability scanning. CSRF protection validation. Session hijacking prevention testing.

11.4 User Acceptance Testing

User acceptance testing (UAT) shall involve actual end-users validating system functionality against requirements. UAT shall occur in staging environment identical to production.

UAT Process: Recruit 20-30 students, 5-10 print shop owners, and 2-3 administrators as test users. Provide test scenarios covering common and edge use cases. Collect feedback through structured forms and interviews. Track defects with severity classification. Require sign-off from user representatives before production deployment.

Acceptance Criteria: All high-priority requirements fully functional. No critical or high-severity bugs remaining. Performance meeting NFR targets. Usability feedback predominantly positive. Documentation complete and accessible.

---

12. Deployment Plan

Phase 1: Infrastructure Setup (Week 1-2) Provision cloud infrastructure including compute instances, database servers, and storage buckets. Configure load balancers and SSL certificates. Set up staging and production environments with identical configurations. Implement monitoring and logging infrastructure. Configure backup and disaster recovery mechanisms.

Phase 2: Database Migration (Week 3) Create database schemas with proper indexing and constraints. Load initial reference data including paper sizes, service types, and default configurations. Set up database replication for high availability. Configure automated backup schedules. Test database failover procedures.

Phase 3: Application Deployment (Week 4) Deploy backend API services with zero-downtime deployment strategy. Deploy frontend applications to CDN with cache invalidation. Configure environment variables and secrets management. Set up API rate limiting and security rules. Implement health check endpoints.

Phase 4: Integration Configuration (Week 5) Configure payment gateway with production credentials. Set up email and SMS service providers with templates. Connect cloud storage with proper access policies. Configure monitoring and alerting rules. Test all external integrations in production environment.

Phase 5: Pilot Launch (Week 6-8) Launch to limited user group (50 students, 3 print shops). Monitor system performance and user feedback closely. Address critical issues immediately. Collect usage analytics and performance metrics. Refine based on real-world usage patterns.

Phase 6: Full Launch (Week 9) Open registration to entire campus community. Conduct marketing and awareness campaigns. Provide training sessions for print shop owners. Set up help desk support channels. Monitor system closely during initial weeks.

Rollback Plan: Maintain previous version deployment ready for instant rollback. Database migration scripts shall be reversible. Feature flags shall allow disabling new functionality without redeployment. Regular backup points before each deployment phase.

Post-Deployment Monitoring: 24/7 monitoring for first two weeks with on-call support. Daily performance and error reviews. Weekly user feedback collection. Bi-weekly update cycles for bug fixes and improvements.

13. Maintenance and Future Enhancements

Maintenance Activities

Regular Maintenance: Weekly database optimization including index rebuilding and statistics updates. Monthly security patches for dependencies and system libraries. Quarterly dependency updates for frameworks and libraries. Continuous monitoring of system logs for errors and anomalies. Regular backup verification and disaster recovery drills.

Corrective Maintenance: Bug fix releases on bi-weekly schedule for non-critical issues. Hot-fix deployments within 4 hours for critical issues affecting system availability. Root cause analysis documentation for all major incidents. Preventive measures implementation to avoid recurring issues.

Adaptive Maintenance: Browser compatibility updates as new versions release. Operating system updates and security patches. Third-party API version upgrades. Payment gateway compliance updates. Database version migrations when required.

Perfective Maintenance: Performance optimization based on usage analytics. UI/UX improvements from user feedback. Query optimization for frequently accessed data. Code refactoring to reduce technical debt. Documentation updates reflecting system changes.

Future Enhancements

Phase 2 Enhancements (6-12 months): Native mobile applications for iOS and Android platforms. Advanced search with filters for document type, date range, and price range. Bulk order placement for multiple documents. Schedule order placement for future pickup. Loyalty program with points and discounts. In-app chat for direct student-shop communication. Document templates for common formats.

Phase 3 Enhancements (12-18 months): Integration with campus learning management system for automatic assignment printing. OCR capability for scanned document processing. Automatic document format conversion and optimization. AI-powered print quality prediction and recommendations. Subscription plans for frequent users. Print preview with actual output simulation. QR code based order pickup verification.

Phase 4 Enhancements (18-24 months): Multi-campus expansion with centralized administration. API marketplace allowing third-party integrations. Advanced analytics dashboard with predictive insights. Blockchain-based transaction verification for dispute resolution. AR-based shop navigation within campus. Voice-based order placement through virtual assistants. Integration with campus ID cards for seamless authentication.

Scalability Enhancements: Microservices architecture migration for independent service scaling. Kubernetes orchestration for container management. Multi-region deployment for disaster recovery. Caching layer optimization with Redis cluster. Database sharding for horizontal scalability. Message queue implementation for asynchronous processing. GraphQL API introduction for flexible data querying.

Business Model Enhancements: Dynamic pricing based on demand and shop capacity. Advertisement platform for print shops to promote services. Premium shop listings with enhanced visibility. Analytics as a service for shop owners. White-label solution for other universities. API access for third-party campus applications.

---

14. Conclusion

The Campus Digital Print & Xerox Service Platform represents a comprehensive solution to modernize campus printing services through digital transformation. By eliminating physical queues and enabling online document submission, payment, and tracking, the system significantly improves operational efficiency for both students and print shop operators.

The architecture follows industry best practices with clear separation of concerns, robust security measures, and scalability considerations. The three-tier design with React frontend, Node.js backend, and PostgreSQL database provides a solid foundation for current requirements while supporting future enhancements. Integration with payment gateways, cloud storage, and notification services ensures a complete end-to-end solution.

Security is paramount with comprehensive measures including data encryption, secure authentication, role-based access control, and PCI DSS compliant payment processing. Performance requirements ensure responsive user experience even under high load conditions. The detailed testing strategy covering unit, integration, system, and user acceptance testing provides confidence in system reliability.

The phased deployment approach with pilot testing minimizes risks while the comprehensive maintenance plan ensures long-term system sustainability. Future enhancement roadmap demonstrates vision for continuous improvement and adaptation to evolving campus needs.

This SRS document provides a complete specification suitable for development team implementation, quality assurance validation, and academic evaluation. The system, when fully implemented, will deliver significant value to the campus community through improved service accessibility, operational transparency, and time savings for all stakeholders.

Approval Signatures:

Project Manager
Date: _______________

Technical Lead
Date: _______________

Quality Assurance Lead
Date: _______________

Academic Supervisor
Date: _______________

Document Control:

- Filename: Campus_Print_SRS_v1.0.md
- Version: 1.0
- Classification: Internal
- Distribution: Development Team, QA Team, Project Stakeholders, Academic Evaluators

End of Document