Workflow Documentation
Online Book Reading & Academic Services Platform

1. <u>Introduction</u>

This document presents a detailed workflow for the Online Book Reading & Academic Services Platform — a secure, cloud-based ecosystem that enables students, researchers, and academicians to read, purchase, and interact with e-books and academic materials. The system integrates digital rights management (DRM), mock test modules, academic writing services, and a job portal to create a complete academic engagement platform.

The primary goal is to ensure content security, enhance learning outcomes, and simplify content administration for publishers and educators.

2. <u>Objectives</u>

- Develop a secure DRM-enabled online reading system that prevents unauthorized downloads or copies.

- Provide mock tests and academic evaluation tools for students and educators.

- Integrate academic writing and subject-based job services.

- Enable seamless admin–user interaction through dashboards and notifications.

- Implement transparent payment and subscription management with Razorpay, Stripe, or PayPal integration.

3. <u>Stakeholders</u>

Role   Responsibilities
- **Admin**: Manages users, uploads content, oversees payments, moderates job and writing services, and generates analytical reports.
- **User (Student/Researcher)**: Registers, purchases, and reads e-books, takes mock tests, accesses notes, requests writing services, and applies for jobs.

4. <u>Workflow Overview</u>

The platform workflow is divided into 11 key phases, from registration to maintenance. It uses a modular architecture powered by:

- Frontend: React.js / Next.js

- Backend: Node.js or Laravel

- Database: MySQL / PostgreSQL

- Cloud Hosting: AWS / Google Cloud

- DRM & Security: Encrypted HTML5 Reader, Watermarking, MFA

- Payment Gateway: Razorpay / Stripe / PayPal

Each module communicates securely through RESTful APIs, maintaining strong authentication and data encryption standards.

5. <u>Detailed Workflow (Phase-Wise)</u>

➢ Phase 1: Registration & Authentication

User: Creates an account and verifies via Multi-Factor Authentication (MFA).

Admin: Assigns user roles (student, researcher, employer).

System: Stores credentials in an encrypted database and activates dashboards.

➢ Phase 2: Content Management

Admin: Uploads e-books, notes, mock test banks, and writing service details.

Implements metadata tagging for efficient search and categorization.

All content is encrypted before publishing.

➢ **Phase 3: Secure Reading Experience**

User: Purchases an e-book or note and reads it via a DRM-protected HTML5 reader.

System: Disables copy, print, download, and screenshot functionality; embeds user watermark.

Admin: Controls content visibility and subscription expiry.

➢ **Phase 4: Mock Test & Exam Preparation**

Admin: Creates and manages test banks (MCQs, essays, case studies).

User: Takes timed tests and receives automatic evaluation.

System: Generates analytics — accuracy, time spent, weak areas, and leaderboard positions.

➢ **Phase 5: Academic Notes Repository**

Admin: Uploads categorized notes with version control.

User: Searches and accesses DRM-protected notes via the secure reader.

System: Maintains log records of user reading activities for analytics.

➢ **Phase 6: Academic Writing & Job Portal**

User: Requests academic writing assistance or applies for subject-specific jobs.

Admin: Reviews, approves, and assigns tasks to writers or employers.

System: Tracks service requests, deadlines, and job statuses.

➢ **Phase 7: Payment & Subscription Management**

User: Chooses between one-time purchases or subscription plans.

System: Processes transactions through Razorpay/Stripe/PayPal.

Admin: Manages refunds, invoices, and credit systems; monitors payment analytics.

> ### ➢ Phase 8: Notification & Communication System

Admin: Sends system-wide announcements, job alerts, or book release updates.

User: Receives in-app and email notifications in real time.

System: Logs all messages and ensures reliable delivery.

> ### ➢ Phase 9: Security & Anti-Piracy Controls

DRM protection ensures all e-books and notes are viewable only within the secure reader.

Watermarks with user credentials deter illegal sharing.

MFA, SSL encryption, and activity monitoring secure user data and transactions.

> ### ➢ Phase 10: Analytics & Reporting

Admin Dashboard: Displays performance metrics — total sales, user engagement, most-read books, and test statistics.

User Dashboard: Shows personal reading activity, test scores, and progress charts.

System: Auto-generates reports for both stakeholders.

> ### ➢ Phase 11: Maintenance & Updates

Admin: Schedules updates, bug fixes, and database backups.

System: Monitors uptime, load, and system health.

Regular version releases ensure scalability and reliability.

6. **Data Flow Summary**

- User Actions → Registration → Payment → Request Access

- System Actions → Authentication → DRM Encryption → Content Delivery

- Admin Actions → Review → Approval → Analytics & Reports

The workflow ensures continuous interaction between Admin, User, and System modules under strict authentication and security protocols.

7. **Deliverables**

- Fully functional online platform (frontend + backend + DRM integration).

- Secure e-book reader with watermark and screenshot protection.

- Mock test system with analytics and reporting.

- Admin and user dashboards with communication channels.

- Integrated payment and subscription module.

- Documentation, deployment support, and administrator training.

8. **Workflow Diagram Reference**

- The workflow diagram should depict:

- Parallel swimlanes for Admin and User roles.

- Phases from Registration → Content Access → Payments → Analytics.

- Arrows represent data and system interaction between modules.
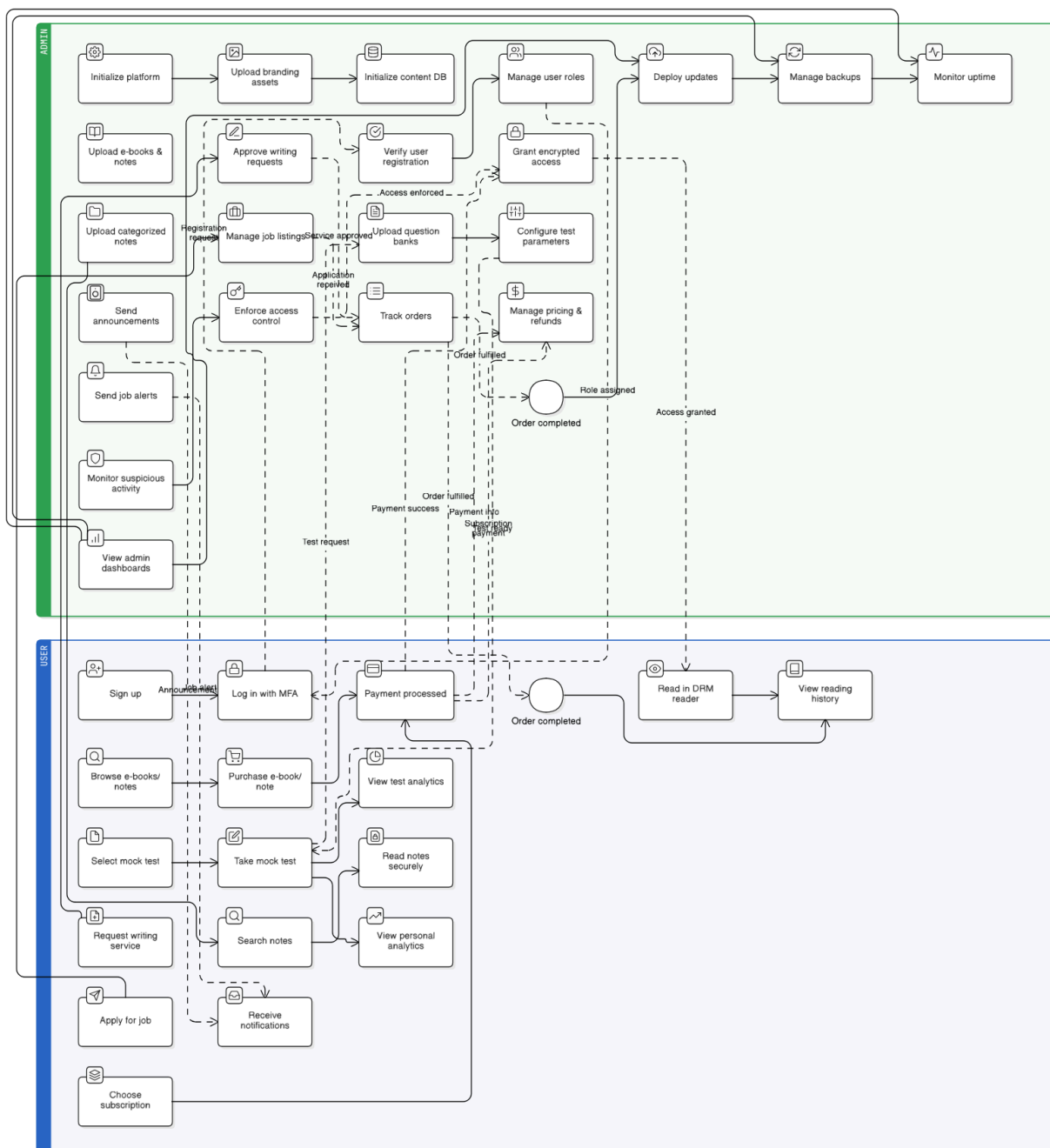
9. **Conclusion**

The workflow defines a structured, secure, and scalable operational model for the Online Book Reading & Academic Services Platform. It aligns technological efficiency with educational innovation, ensuring an end-to-end protected and engaging digital learning experience.

This document serves as a baseline for system development, deployment, and continuous improvement.