PRESIDENCY UNIVERSITY
Private University Estd. in Karnataka State by Act No. 41 of 2013
Itgalpura, Rajankunte, Yelahanka, Bengaluru – 560064

VGP-AYUSH – VAIDYA GRANTH PARAKH
EXPERT REVIEW OF AYUSH MEDICAL TEXTBOOK'S.

A PROJECT REPORT

Submitted by

NISCHAY N RAJ – 20221CSE0637

CHARANYA SRI S – 20221CSE0661

ANUSHA D R – 20221CSE0655

Under the guidance of,

Ms. Rohini A

Assistant Professor

Presidency School of Computer Science and Engineering

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

PRESIDENCY UNIVERSITY

BENGALURU

DECEMBER 2025

PRESIDENCY UNIVERSITY
Private University Estd. in Karnataka State by Act No. 41 of 2013
Itgalpura, Rajankunte, Yelahanka, Bengaluru – 560064

xtbooks

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

Certified that this report "**VGP-AYUSH – Vaidya Granth Parakh Expert Review of AYUSH Medical Textbooks.**" is a bonafide work of "**Nischay N Raj (20221CSE0637), Anusha D R (20221CSE0655), Charanya Sri S(20221CSE0661)** "who have successfully carried out the project work and submitted the report for partial fulfilment of the requirements for the award of the degree of BACHELOR OF TECHNOLOGY in **COMPUTER SCIENCE ENGINEERING** during 2025-26.

Ms. Rohini A
Project Guide
PSCS
Presidency University

Mr. Muthuraju V
Program Project Coordinator
PSCS
Presidency University

Dr. Sampath A K
School Project Coordinator
PSCS
Presidency University

Dr. Blessed Prince
Head of the Department
PSCS
Presidency University

Dr.Shakeera L
Associate Dean
PSCS
Presidency University

Dr. Duraipandian N
Dean
PSCS & PSIS
Presidency University

Examiners

Sl. no.	Name	Signature	Date
1	Anu Joseph		01.12.2025
2	Riya Sanjesh		01.12.25

VGP-Vaidya Granth Parakh-Review of Medical Textbooks

PRESIDENCY UNIVERSITY

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

We, the students of final year **B.Tech in COMPUTER SCIENCE ENGINEERING** at Presidency University, Bengaluru, named **NISCHAY N RAJ, ANUSHA D R, and CHARANYA SRI S**, hereby declare that the project work titled *"**VGP-AYUSH** – **Vaidya Granth Parakh: Expert Review of AYUSH Medical Textbooks**"* has been independently carried out by us and submitted in partial fulfilment for the award of the degree of **B.Tech in COMPUTER SCIENCE ENGINEERING** during the academic year **2025–26**. Further, the matter embodied in the project has not been submitted previously by anybody for the award of any Degree or Diploma to any other institution .

NISCHAY N RAJ	**20221CSE0637**	Nischay N Raj
CHARANYA SRI S	**20221CSE0661**	Charanya S
ANUSHA D R	**20221CSE0655**	Anusha D R.

PLACE: BENGALURU

DATE: 01/12/2025

iii

VGP-Vaidya Granth Parakh-Review of Medical Textbooks

ACKNOWLEDGEMENT

For the successful completion of this capstone project titled **"VGP – Vaidya Granth Parakh: AYUSH Textbook Expert Review and Quality Assessment Platform**, We/I have received the support and the guidance from many people whom I would like to mention with deep sense of gratitude and indebtedness. We extend our gratitude to our beloved **Chancellor, Pro-Vice Chancellor, and Registrar** for their support and encouragement in completion of the project.

I would like to sincerely thank my internal guide Dr./Mr/Ms **Rohini A, Assistant Professor**, Presidency School of Computer Science and Engineering, Presidency University, for her moral support, motivation, timely guidance and encouragement provided to us during the period of our project work.

I am also thankful to **Dr. Blessed Prince, Head of the Department, Presidency School of Computer Science and Engineering** Presidency University, for her mentorship and encouragement.

We express our cordial thanks to **Dr. Duraipandian N**, Dean PSCS & PSIS, **Dr. Shakkeera L**, Associate Dean, Presidency School of computer Science and Engineering and the Management of Presidency University for providing the required facilities and intellectually stimulating environment that aided in the completion of my project work.

We are grateful to **Dr. Sampath A K, and Dr. Jayavadivel Ravi, PSCS** Project Coordinators, Presidency School of Computer Science and Engineering, or facilitating problem statements, coordinating reviews, monitoring progress, and providing their valuable support and guidance.

We are also grateful to Teaching and Non-Teaching staff of Presidency School of Computer Science and Engineering and also staff from other departments who have extended their valuable help and cooperation.

NISCHAY N RAJ
CHARANYA SRI S
ANUSHA D R

iv

Abstract

This project presents **VGP–AYUSH (Vaidya Granth Parakh)**, a web-based expert review and quality assessment platform developed to standardize the evaluation of AYUSH (Ayurveda, Siddha, and Unani) medical textbooks in accordance with the **National Commission for Indian System of Medicine (NCISM) assessment scale**. Textbooks form the foundation of AYUSH education; however, the existing evaluation mechanism is largely manual, time-consuming, geographically constrained, and lacks transparency.

The proposed system digitizes the entire textbook evaluation workflow by providing **role-based access** for authors, reviewers, and administrators. Authors can submit textbooks electronically, reviewers can evaluate assigned books using structured NCISM rubrics, and administrators can manage reviewer assignments, monitor evaluation progress, and publish final results. The platform supports **multi-reviewer scoring**, automated score aggregation, and classification of textbooks, thereby reducing subjective bias and administrative overhead.

The system is implemented using **Node.js, Express.js, MongoDB, and React.js**, ensuring scalability, security, and usability. A centralized public repository enables students and faculty to access lists of approved textbooks, improving transparency and informed academic decision-making. The proposed solution significantly improves efficiency, consistency, and accessibility in AYUSH textbook evaluation and provides a scalable digital foundation aligned with national education modernization initiatives.

Table of Content

List Of Tables

List of Figures

Abbreviations

AYUSH	Ayurveda, Yoga & Naturopathy, Unani, Siddha and Homeopathy
ASU	Ayurveda, Siddha and Unani
NCISM	National Commission for Indian System of Medicine
VGP	Vaidya Granth Parakh
MERN	MongoDB, Express.js, React.js, Node.js
RBAC	Role-Based Access Control
JWT	JSON Web Token
API	Application Programming Interface
PDF	Portable Document Format
UAT	User Acceptance Testing
SDG	Sustainable Development Goal
HTTPS	Hyper Text Transfer Protocol Secure
AWS	Amazon Web Services
S3	Simple Storage Service
DB	Database
CRUD	Create, Read, Update, Delete
ORM	Object Relational Mapping
REST	Representational State Transfer
DPI	Digital Personal Data
DPDPA	Digital Personal Data Protection Act
AI	Artificial Intelligence
ML	Machine Learning

Chapter 1
Introduction

Education within the Indian traditional medical systems—Ayurveda, Siddha, Unani (collectively referred to as **ASU** or **AYUSH**)—is fundamentally dependent on the quality, accuracy, and credibility of its textbooks. Textbooks serve as the primary reference for students, educators, and practitioners, shaping knowledge acquisition, clinical reasoning, and subject interpretation. However, unlike other modern medical fields, the AYUSH domain faces persistent inconsistencies in textbook quality, ranging from highly scholarly, research-backed works to outdated, poorly structured, and commercially biased publications. These disparities adversely affect learning outcomes and create a knowledge gap across institutions.

Recognizing this issue, the **National Commission for Indian System of Medicine (NCISM)** has established an official **Textbook Assessment Scale**, a rubric-based framework to evaluate the comprehensiveness, authenticity, pedagogical quality, and curriculum alignment of ASU textbooks. While this scale provides a standardized benchmark, the **existing evaluation process is entirely manual**, involving expert committees dispersed across the country. This results in delays, limited transparency, inconsistencies in scoring, and restricted accessibility of the final recommended book lists. As a result, students and educators lack clear guidance on which books meet NCISM's academic and quality standards.

To address these challenges, this project proposes **VGP–AYUSH (Vaidya Granth Parakh)**—a **web-based, multi-reviewer, role-based evaluation platform** that digitizes NCISM's assessment process. Using web technologies such as **Node.js, Express.js, React.js, and MongoDB**, the platform automates book submissions, reviewer assignments, rubric-based scoring, score aggregation, and the publication of results on a transparent, public repository. This enhances efficiency, reduces bias, increases accessibility, and supports nationwide scalability.

This chapter provides a comprehensive introduction to the project, including the background context, the challenges within existing systems, the need for digitization in AYUSH textbook evaluation, the scope and significance of the proposed approach, and its alignment with national and global educational goals.

1.1 Background

India has one of the largest AYUSH education ecosystems in the world, consisting of hundreds of colleges and universities offering undergraduate, postgraduate, and doctoral programs. Textbooks are the primary academic resource for these programs and play a crucial role in shaping medical knowledge and clinical competence.

Historically, AYUSH textbooks have been authored and published by diverse sources, including individual scholars, educational institutions, and private publishers. While many of these works adhere closely to classical texts and regulatory guidelines, several others lack academic rigor, updated references, or standardized presentation. This results in contradictory interpretations, curriculum misalignment, and difficulties in maintaining uniform learning standards across institutions.

To address these concerns, NCISM introduced a structured **Textbook Assessment Scale** covering parameters such as content accuracy, pedagogical quality, research value, references, and presentation. However, the absence of a digital system to enforce and manage this assessment has limited its effectiveness. The need for a centralized, technology-driven solution therefore becomes critical to ensure consistency and transparency in textbook evaluation.

1.2 Statistics of the Project

The relevance and necessity of the proposed system can be highlighted through the following indicative statistics:

- India hosts **800+ AYUSH colleges** offering ASU education.
- Thousands of textbooks and reference materials are currently in circulation.
- Every academic year sees **hundreds of new or revised textbooks** entering the market.
- Manual evaluations may take **several months per book**, leading to delayed adoption.
- Evaluation committees are geographically distributed across states, increasing coordination complexity.

These statistics demonstrate that a manual evaluation framework is not scalable or sustainable for nationwide academic governance. A digital platform such as **VGP–AYUSH** can significantly reduce evaluation time, improve transparency, and provide centralized access to verified academic resources.

1.3 Prior Existing Technologies

Several digital systems exist for content evaluation and academic quality assessment in other domains; however, their applicability to AYUSH education is limited.

Existing Technologies Include:

- Manual committee-based textbook reviews managed through documents and emails.
- Web-based peer-review systems used in modern medical publishing.
- Learning management platforms that recommend content without regulatory compliance.
- AI-based content analysis tools applied primarily in general education.

Limitations of Existing Technologies:

- No platform directly implements the **NCISM official assessment scale**.
- Lack of role-based workflows tailored to authors, reviewers, and regulators.
- Absence of multi-reviewer score aggregation.
- No centralized public repository for approved AYUSH textbooks.

As a result, existing technologies fail to address the domain-specific, regulatory, and transparency requirements of AYUSH textbook evaluation.

1.4 Proposed Approach

The proposed solution, **VGP–AYUSH (Vaidya Granth Parakh)**, adopts a **web-based, role-driven, and rubric-oriented approach** to digitize the NCISM textbook assessment process.

Core Approach Includes:

- Authors submit textbooks digitally along with metadata.
- Administrators assign multiple expert reviewers.
- Reviewers evaluate textbooks using an online NCISM rubric.
- An evaluation engine aggregates reviewer scores using predefined weights.
- Approved results are published in a public, searchable repository.

The system is implemented using modern web technologies such as **React.js**, **Node.js**, **Express.js**, and **MongoDB**, ensuring scalability, usability, and security. Cloud deployment further enables nationwide access and availability.

1.5 Objectives

Primary Objectives

1. To design and develop a web-based platform for evaluating AYUSH textbooks.
2. To implement NCISM's assessment rubric in a structured digital format.
3. To enable multi-reviewer evaluation and automated score aggregation.
4. To provide role-based dashboards for authors, reviewers, and administrators.
5. To create a transparent and accessible repository of evaluated textbooks.

Secondary Objectives

6. To reduce manual effort and administrative overhead.
7. To minimize reviewer bias through standardized aggregation.
8. To improve accessibility of quality learning resources.
9. To support scalability through cloud-based deployment.
10. To provide a foundation for future AI-assisted evaluation enhancements.

1.6 Sustainable Development Goals (SDGs)

The VGP–AYUSH project aligns with the following **United Nations Sustainable Development Goals**:

SDG 4 – Quality Education

Ensures access to high-quality, standardized learning materials for AYUSH students.

SDG 9 – Industry, Innovation and Infrastructure

Promotes digital infrastructure for educational quality assessment.

SDG 10 – Reduced Inequalities

Enables equal access to verified textbooks regardless of institution or location.

SDG 16 – Peace, Justice and Strong Institutions

Enhances transparency, accountability, and fairness in academic governance.

Fig. 1.1 Sustainable Development Goals (selected for alignment)

1.7 Overview of the Project Report

This project report is organized into the following chapters:

- **Chapter 1 – Introduction:**

  Presents the background, need, approach, objectives, and SDG alignment of the project.

- **Chapter 2 – Literature Review:**

  Reviews existing studies, technologies, and research related to textbook evaluation and digital assessment systems.

- **Chapter 3 – Methodology:**

  Describes the software development methodology and testing strategy used.

- **Chapter 4 – Project Management:**

  Covers planning, timeline, risk analysis, and management practices.

- **Chapter 5 – Analysis and Design:**

  Details system requirements, architecture, data models, and UML diagrams.

- **Chapter 6 – Implementation:**

  Explains frontend, backend, database, and deployment implementation.

- **Chapter 7 – Evaluation and Results:**

  Discusses testing procedures, performance results, and validation outcomes.

- **Chapter 8 – Social, Legal, Ethical and Sustainability Aspects:**

  Analyzes broader implications of the system.

- **Chapter 9 – Conclusion and Future Enhancements:**

  Summarizes outcomes and outlines future scope.

Chapter 2
Literature review

The review of existing literature is fundamental to understanding the evolution of textbook evaluation methodologies, digital assessment systems, and web-based review platforms in the context of traditional medical education. The AYUSH domain—encompassing Ayurveda, Siddha, and Unani—requires specialized mechanisms for assessing the quality of academic textbooks due to its unique blend of classical knowledge, modern pedagogy, and regulatory standards issued by the National Commission for Indian System of Medicine (NCISM).

The primary focus of this literature review is to examine existing research in:

- Rubric-based evaluation of educational content
- Web-based textbook evaluation and quality assurance systems
- Multi-reviewer and peer-review models
- Technology-assisted assessment frameworks
- AI-supported content evaluation tools

Although significant progress has been made in digitalizing educational evaluation in modern medicine and higher education, literature specifically addressing **AYUSH textbook evaluation** remains limited. Most existing methods rely heavily on manual review, underscoring the need for the proposed **VGP-AYUSH** system.

2.1 Rubric-Based Evaluation Frameworks

Rubric-based assessment systems offer a structured, transparent, and repeatable mechanism for evaluating educational resources. Sharma et al. (2022) introduced a rubric-based model specifically for **Ayurveda medical literature**, emphasizing quality indicators such as content accuracy, classical text alignment, pedagogical clarity, and reference credibility. Their framework provides the closest academic predecessor to the NCISM Assessment Scale

.

While the rubrics established a strong theoretical foundation, the evaluation remained **manual**, requiring scholars to fill printed or emailed forms. Additionally:

- Scores varied depending on reviewer interpretation
- No centralized storage existed for results
- Consolidation was slow and effort-intensive

In 2023, NCISM released the official **Assessment Scale for ASU Textbooks**, standardizing textbook evaluation throughout the country. This scale is comprehensive, covering:

- Accuracy of content
- Clarity and structure
- Research value
- Use of references
- Pedagogical relevance
- Presentation quality

However, as highlighted in your uploaded PPT and literature files, NCISM's scale is **not digitally implemented**, and no official web platform exists to enforce, record, or automate scoring at scale .

These limitations reinforce the need for a digital system like **VGP-AYUSH**, which integrates the NCISM rubric into an automated workflow.

2.2 Web-Based Evaluation Systems

Web-enabled evaluation systems have emerged as an effective means to streamline educational quality assessment.
Li et al. (2021) developed a **Web-Based Evaluation System for Traditional Chinese Medicine (TCM) Textbooks**, demonstrating that digitizing evaluation:

- Improves reviewer coordination
- Accelerates the assessment lifecycle

- Enables centralized storage
- Enhances transparency and accessibility

Their work is relevant to AYUSH due to similarities between traditional Asian medical education frameworks. However, the TCM system:

- Did not include multi-reviewer scoring
- Did not aggregate scores automatically
- Did not support role-based access control
- Was not aligned with NCISM's rubric

Thus, while Li's work proves the feasibility of digital textbook evaluation, it does not meet AYUSH-specific needs.

2.3 Peer-Review and Multi-Reviewer Frameworks

Multi-reviewer evaluation frameworks are widely used in academic publishing, medical curriculum review, and research assessment. Garg et al. (2020) proposed a **peer-review model for curriculum materials**, emphasizing:

- Reducing individual reviewer bias
- Balancing perspectives
- Ensuring more objective results
- Improving reviewer diversity

Their model also incorporated AI-based quality checks, though computational limitations made large-scale deployment challenging.

AYUSH textbook review traditionally uses **committee-based** evaluation. However:

- Reviewers are geographically dispersed
- Evaluation happens sequentially
- Consolidation takes weeks to months
- No digital workflow exists to unify the process

The **VGP-AYUSH** platform draws inspiration from academic peer-review models by assigning **three reviewers per book** and aggregating scores automatically, ensuring objectivity and reduction of reviewer bias.

2.4 Technology-Driven Content Assessment

Recent research explores the integration of software systems, AI, metadata analysis, and semantic processing to evaluate the quality of educational materials.

Tavakoli et al. (2020) used **metadata-based prediction models** to evaluate open educational resources. Their system analyzed factors such as:

- Content completeness
- Readability
- Language quality

Similarly, Galindo-Durán & Medina-Ramírez (2021) used semantic platforms to integrate **Open Educational Resources (OERs)** into structured digital repositories, ensuring consistent accessibility.

While these systems focus on modern, general education, they offer guidance for building digital repositories and quality assurance pipelines relevant to AYUSH.

In the context of AYUSH:

- There is no standardized digital repository for evaluated textbooks
- Metadata-driven analysis could support automated recommendations in the future
- Semantic consistency checks may help ensure curriculum alignment

These findings validate the feasibility of applying digital tools to the AYUSH domain.

2.5 AI-Assisted Assessment Tools

Beyond traditional rubrics, modern educational platforms increasingly integrate AI-based assessment.

Examples include:

- Automated assessment of concept maps (Kroeze et al., 2021)
- AI-based scoring of programming tasks (Conejo et al., 2019)
- LLM-assisted essay scoring (Song et al., 2024)

While these techniques are not directly applicable to AYUSH textbook evaluation, they demonstrate the evolution of:

- Automated scoring
- Digital content validation
- Semantic evaluation
- Pattern recognition in academic work

Your uploaded research paper draft also references future plans for **AI-enabled plagiarism detection** and **automatic evaluation assistance** within the VGP-AYUSH ecosystem

These insights support the project's long-term direction toward hybrid manual–AI evaluation for AYUSH textbooks.

2.6 Existing Gaps in Current Systems

Across all reviewed literature, several critical gaps become evident:

1. **Lack of AYUSH-Specific Digital Evaluation Platforms**
   No existing system digitizes the NCISM textbook assessment.
2. **Absence of Multi-Reviewer Digital Collaboration**
   Most frameworks rely on single reviewers or sequential manual reviews.
3. **Limited Transparency and Public Access**
   Students, teachers, and institutions cannot easily view recommended books.

4. **No Role-Based Dashboards**
   Existing systems lack author, reviewer, and admin workflow separation.
5. **No Automated Score Aggregation**
   Manual compilation leads to delays and potential errors.
6. **No Centralized Repository for Evaluated AYUSH Books**
   There is no nationwide searchable database aligned with NCISM standards.

These gaps directly align with the motivations for developing **VGP-AYUSH**.

2.7 Summary of Literature Review

The literature demonstrates a wide range of approaches—manual rubrics, peer review, digital repositories, and AI-based evaluation—but none fully satisfy the needs of:

- Standardization
- Scalability
- Transparency
- Role management
- Multi-reviewer aggregation
- NCISM-specific requirements

The proposed project **VGP-AYUSH** is therefore positioned as a novel, domain-specific, and structurally sound solution that integrates:

- NCISM rubric implementation
- Multi-reviewer scoring
- Web-based evaluation

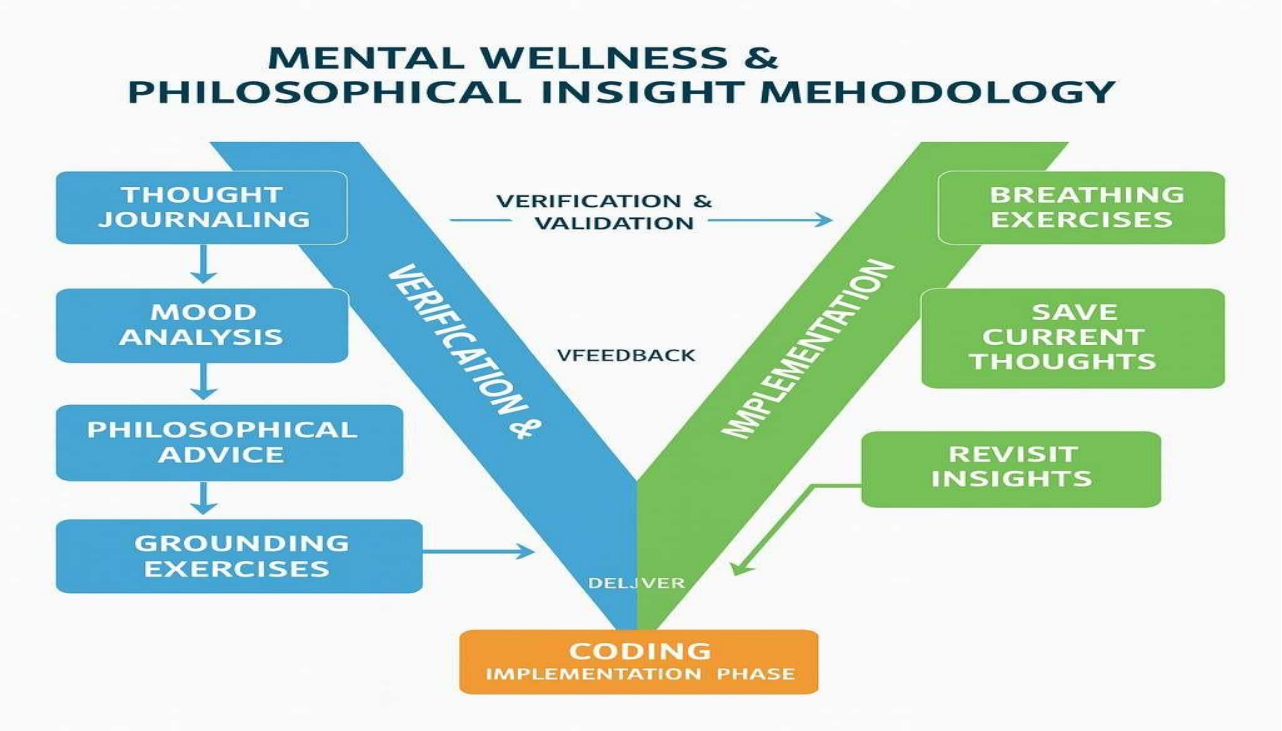
Chapter 3

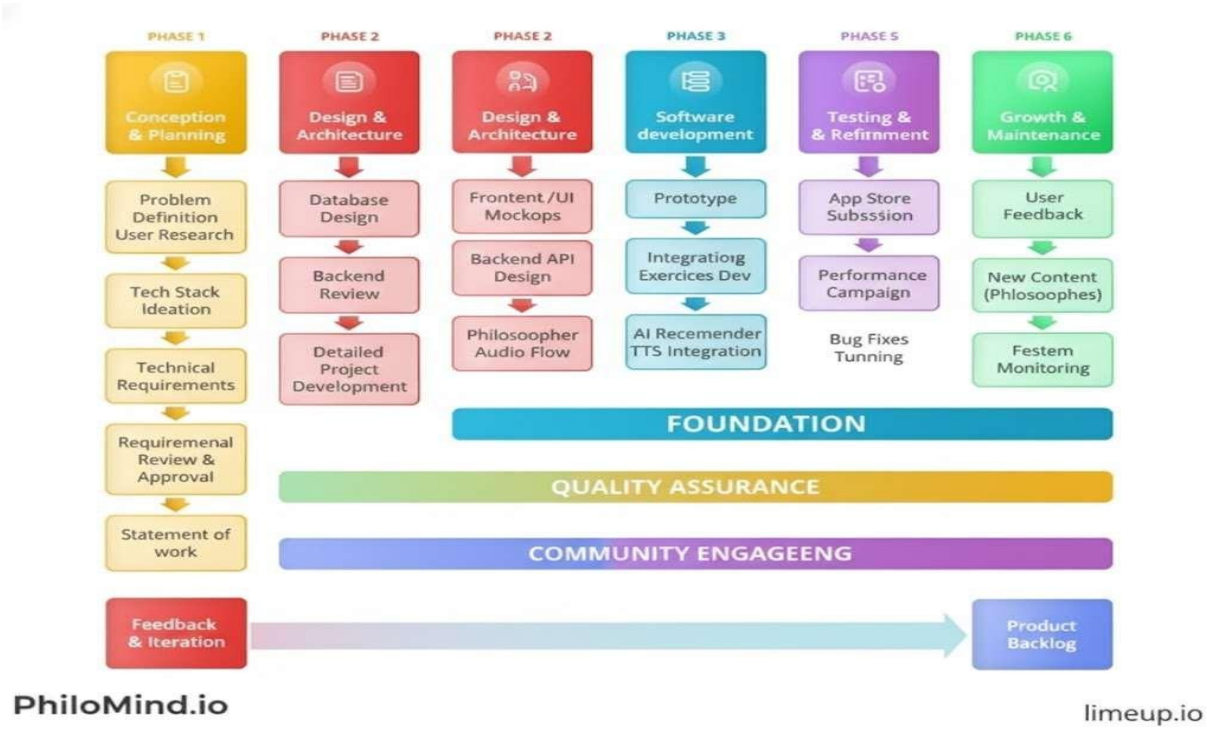
Methodology

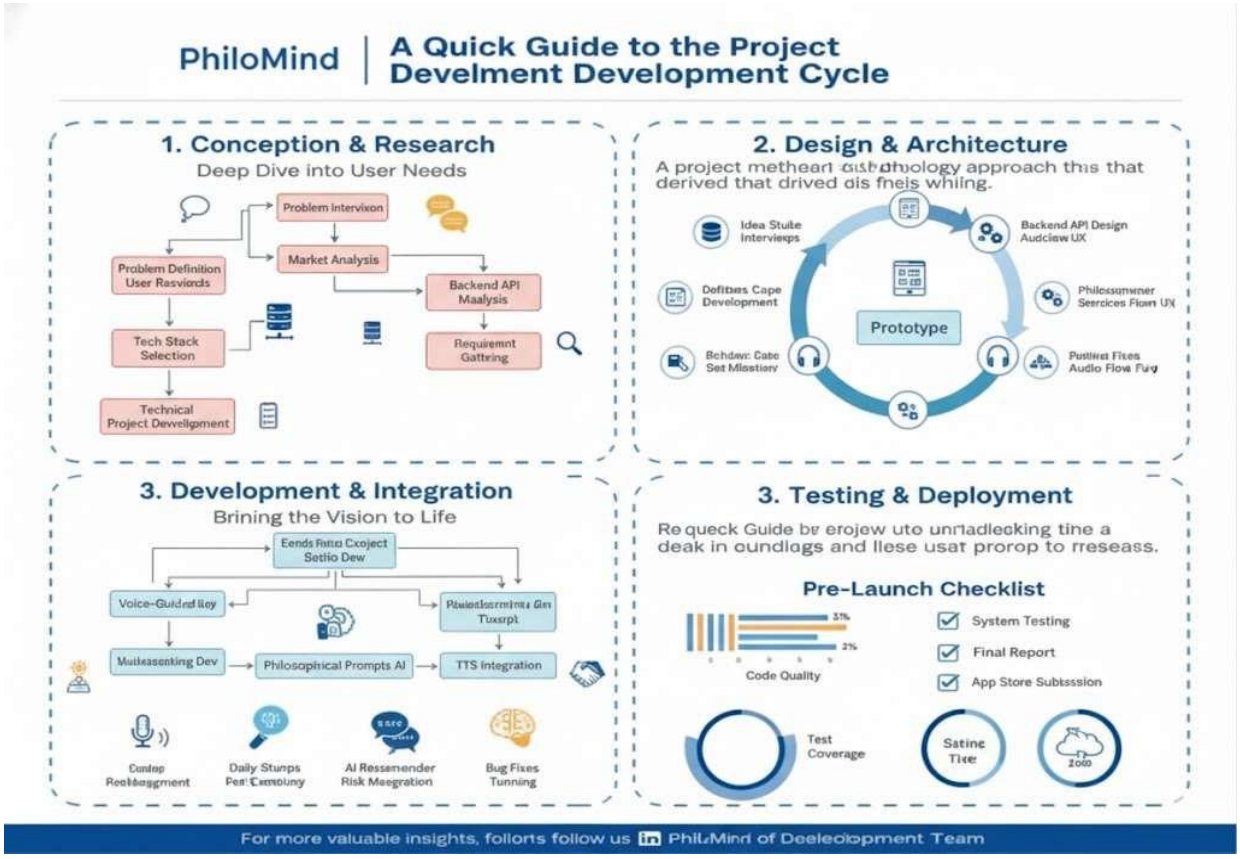
The methodology adopted for the development of the **VGP–AYUSH (Vaidya Granth Parakh)** platform was guided by the need to create a system that is academically reliable, transparent, and practically usable within the regulatory framework of AYUSH education. Since the project deals with textbook evaluation—an activity that directly impacts learning standards, curriculum consistency, and institutional credibility—it was essential to follow a structured yet flexible development approach.

Unlike conventional software projects that focus only on functionality, this platform required equal emphasis on **accuracy, traceability, reviewer accountability, and compliance with NCISM guidelines**. Therefore, the methodology was designed to ensure clear requirement mapping, systematic development, continuous validation, and reliable outcome generation.

The development process combined structured software engineering practices with a user-centric workflow to ensure the system is both technically robust and easy to use for authors, reviewers, and administrators.




PHASE 1
Conception & Planning
Problem Definition User Research
Tech Stack Ideation
Technical Requirements
Requiremenal Review & Approval
Statement of work
PHASE 2
Design & Architecture
Database Design
Backend Review
Detailed Project Development
PHASE 2
Design & Architecture
Frontent /UI Mockops
Backend API Design
Philosoopher Audio Flow
PHASE 3
Software development
Prototype
Integratiog Exercices Dev
AI Recemender TTS Integration
PHASE 5
Testing & & Refirmment
App Store Subsssion
Performance Campaign
Bug Fixes Tunning
PHASE 6
Growth & Maintenance
User Feedback
New Content (Phlosoophes)
Festem Monitoring
FOUNDATION
QUALITY ASSURANCE
COMMUNITY ENGAGEENG
Feedback & Iteration
Product Backlog
PhiloMind.io
limeup.io


PhiloMind | A Quick Guide to the Project Develment Development Cycle
1. Conception & Research
Deep Dive into User Needs
Problem Intervixon
Market Analysis
Backend API Maalysis
Requiremnt Gattring
Problem Definition User Rasviends
Tech Stack Selection
Technical Project Dewvellgpment
2. Design & Architecture
Idea Stuite Interviceps
Backend API Design Audciew UX
Defitines Cape Development
Philossumener Seercices Flom UX
Prototype
Bchdwn: Cate Set Mlostiony
Pustiest Fixes Audio Flow Fury
3. Development & Integration
Brining the Vision to Life
Voice-Guidred Iley
Philosophical Prompts AI
TTS Integration
Multiasenting Dev
Daily Stunps
AI Ressenender Risk Meegration
Bug Fixes Tunning
3. Testing & Deployment
Pre-Launch Checklist
Code Quality
System Testing
Final Report
App Store Subsssion
Test Coverage
For more valuable insights, follorts follow us PhiloMind of Deeleclopment Team

3.1 Overview of the Development Methodology

The development of VGP–AYUSH followed a **phased and iterative methodology**, beginning with requirement analysis and progressing through system design, implementation, testing, and validation. Each phase was carefully planned to reduce ambiguity and ensure consistency with the NCISM Textbook Assessment Scale.

The methodology emphasized:

- Early understanding of domain-specific rules
- Modular system design
- Continuous verification of requirements
- Validation through real workflow simulations

This approach ensured that the final system accurately represents NCISM's textbook evaluation process while remaining scalable for nationwide deployment.

3.2 Methodology Selection Rationale

The methodology was chosen based on the following characteristics of the project:

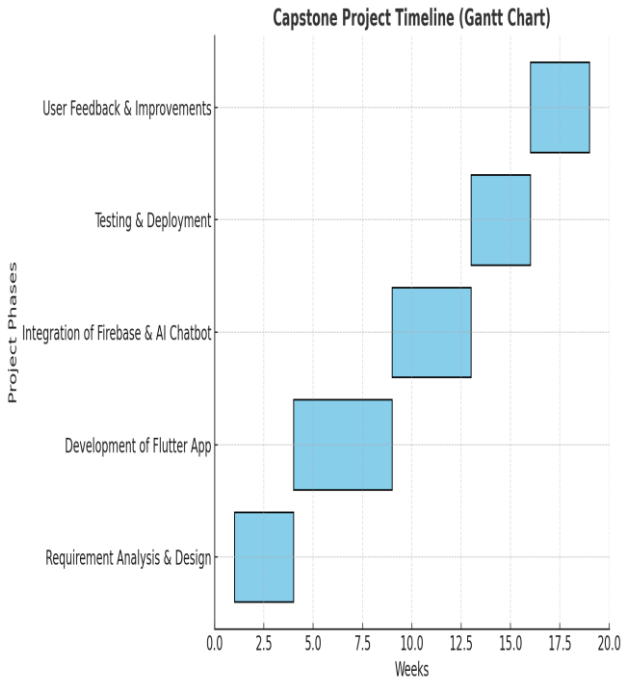
- Multiple stakeholder roles (Author, Reviewer, Admin)
- Regulated evaluation criteria defined by NCISM
- Sensitive academic content (unpublished textbooks)
- Requirement for auditability and transparency
- Need for structured documentation

A **verification-oriented development strategy** was adopted where requirements defined at the initial stage were continuously checked during design and validated during testing. This reduced rework and helped maintain alignment between policy guidelines and technical implementation.

3.3 Phases of Methodology

The overall development process was divided into logical phases, each building upon the outcomes of the previous stage.

Fig 3.1 Methodology Flow Diagram



3.3.1 Requirement Understanding and Analysis

The first phase focused on understanding the academic and regulatory requirements of AYUSH textbook evaluation. This involved studying NCISM assessment documents, understanding textbook review workflows, and analyzing limitations of existing manual evaluation practices.

Key requirements identified during this phase included:

- Digital submission of textbooks
- Multi-reviewer evaluation using a standardized rubric
- Score aggregation using predefined weights
- Result publication through a transparent repository
- Role-based access control

This phase helped establish a clear functional boundary for the system and served as the foundation for design decisions.

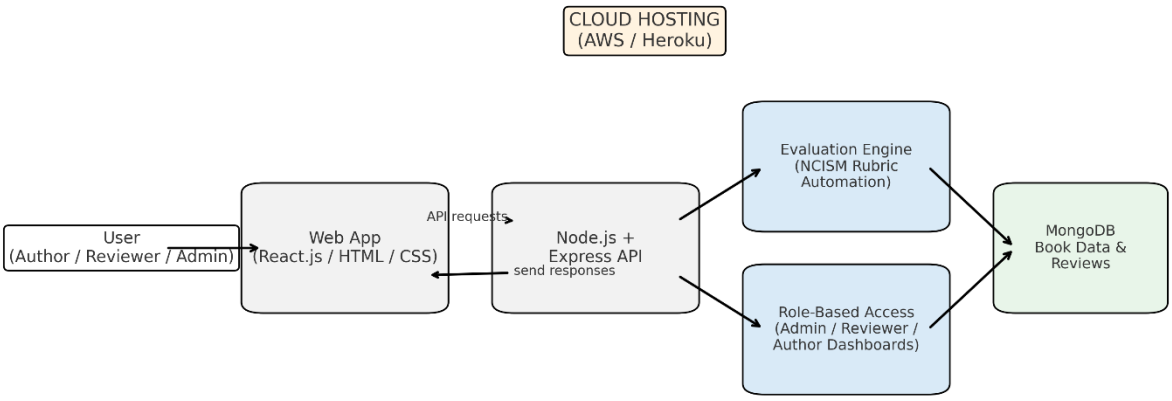
3.3.2 System Design and Architectural Planning

Once the requirements were clearly defined, the next phase involved designing the system architecture. The platform was designed as a **three-tier web application**, separating concerns between presentation, application logic, and data storage.

The system architecture consists of:

- A frontend interface for user interaction
- A backend service layer to handle logic and validation
- A database layer for reliable data persistence

Fig 3.2 System Architecture Diagram



This layered architecture improves maintainability, scalability, and security while allowing independent upgrades of system components.

3.3.3 Functional Module Design

To simplify development and testing, the system was divided into independent functional modules. Each module was designed with a clear responsibility while remaining interconnected within the overall workflow.

Major modules include:

- Author submission module
- Reviewer evaluation module
- Admin supervision module
- Evaluation and scoring engine
- Public textbook repository

By separating responsibilities, the system avoids complexity and allows future upgrades such as AI-based evaluation or plagiarism detection.

3.3.4 Implementation Phase

The implementation phase translated the architectural and functional designs into a working system. The platform was developed using modern web technologies to ensure responsiveness, security, and scalability.

The frontend was designed with simple layouts and structured dashboards to ensure users with minimal technical background could interact with the system easily. The backend handled authentication, workflow enforcement, reviewer assignment logic, and score computation. MongoDB was used to store flexible and structured datasets related to users, books, reviews, and results.

Special care was taken to ensure that:

- The NCISM rubric is enforced digitally without modification
- Reviewer inputs are validated before submission
- Scores are aggregated automatically and consistently

3.3.5 Testing and Validation

Testing was carried out in multiple stages to ensure system reliability and correctness.

Initially, individual modules such as login, book upload, and rubric submission were tested independently. Following this, integration testing was performed to verify seamless interaction between frontend, backend, and database components.

Complete workflow testing was conducted to simulate real-world usage:

- Author uploads textbook
- Admin assigns reviewers
- Reviewers submit evaluations
- System generates final classification

This ensured the end-to-end evaluation process worked exactly as intended.

3.3.6 User-Level Validation

A pilot-level validation was conducted using sample textbooks and internal reviewers to simulate actual evaluation scenarios. Feedback was collected regarding ease of use, clarity of rubric presentation, and accuracy of automated score aggregation.

The validation confirmed that:

- Evaluation time was significantly reduced
- Score computation was error-free
- The workflow was transparent and traceable

3.4 Supporting Development Practices

To enhance reliability and usability, additional practices were incorporated:

- **User-Centered Design:** Interfaces were kept simple and intuitive
- **Iterative Refinement:** Features were refined through repeated testing

- **Documentation-Driven Development:** Each phase was documented for academic compliance

3.5 Tools and Technologies Used

The following tools supported the development process:

- **Frontend:** React.js, HTML, CSS, JavaScript
- **Backend:** Node.js, Express.js
- **Database:** MongoDB Atlas
- **Storage:** AWS S3 (for textbooks)
- **Testing:** Postman, Selenium
- **Deployment:** AWS EC2 / Heroku

3.6 Summary of Methodology

The methodology adopted for VGP–AYUSH ensured that the system was:

- Academically compliant with NCISM
- Technically reliable and scalable
- Transparent and auditable
- User-friendly for non-technical stakeholders

By following a structured, verification-oriented, and user-focused methodology, the project successfully transformed a manual textbook evaluation process into a scalable digital platform capable of supporting AYUSH education nationwide.

Chapter 4

PROJECT MANAGEMENT

Project management for the VGP–AYUSH platform was planned with the aim of delivering a stable, usable, and scalable system within a limited academic timeline. Since the platform supports a sensitive evaluation process under NCISM, the project could not be handled in an ad-hoc manner. Instead, the work was broken into well-defined phases with clear milestones, responsibilities, and review points.

This chapter presents the overall project timeline, major risks identified during development, and an approximate project budget. Together, they reflect how the team systematically organized the work to complete the VGP–AYUSH system within the constraints of time, cost, and quality.

4.1 Project Timeline

The project was planned for completion over a period of **approximately 10–12 weeks**, in line with the academic schedule. The timeline was divided into sequential but slightly overlapping phases so that feedback from one phase could inform improvements in the next.

Phase 1 – Requirement Study and Conceptualization (Week 1–2)

The first two weeks were dedicated to understanding the problem space and defining what the system needed to achieve.

- Studied NCISM Textbook Assessment Scale and evaluation documents.
- Understood the roles of authors, reviewers, and administrators.
- Analyzed limitations of existing manual evaluation practices.
- Defined the scope and boundaries of the VGP–AYUSH platform.

Outcome: A clear set of functional and non-functional requirements, and an initial vision of the platform's workflow.

Phase 2 – System and Interface Design (Week 3–4)

Once the requirements were frozen, the next phase focused on designing how the system would look and behave.

- Designed overall three-tier architecture (frontend, backend, database).
- Prepared high-level block diagrams and data flow diagrams.
- Sketched dashboard layouts for Author, Reviewer, and Admin.
- Designed the database collections for users, books, reviews, and results.

Outcome: A complete design blueprint that guided all coding activities.

Phase 3 – Backend and Database Development (Week 5–7)

This phase concentrated on implementing the server-side logic and data layer.

- Implemented Node.js + Express APIs for authentication, book upload, review submission, and admin actions.
- Configured MongoDB Atlas and created schemas using Mongoose.
- Integrated AWS S3 for secure PDF storage.
- Implemented the evaluation engine to apply NCISM weights and compute final scores.

Outcome: A working backend that could support all major workflows through REST APIs.

Phase 4 – Frontend Development and Integration (Week 6–8)

Frontend work started in parallel and was gradually integrated with the backend.

- Built React.js components for login, dashboards, book upload forms, and rubric evaluation screens.
- Connected frontend components to backend APIs using Axios.
- Implemented role-based routing so each user saw only their permitted pages.
- Performed initial integration testing to verify the end-to-end flow.

Outcome: A fully functional web interface connected to the backend and database.

Phase 5 – Testing and Refinement (Week 9–10)

In this phase, both functional and non-functional aspects of the system were tested.

- Created and executed test cases for each major feature.
- Conducted integration tests for complete workflows (upload → assign → review → evaluate → publish).
- Carried out small pilot runs with sample textbooks and mock reviewers.
- Fixed bugs, refined error messages, and improved UI clarity based on feedback.

Outcome: A stable system whose behavior matched the requirements defined earlier.

Phase 6 – Documentation and Final Review (Week 11–12)

The final phase focused on documentation and academic deliverables.

- Prepared user-level and technical documentation.
- Collected screenshots and diagrams for the project report.
- Reviewed features against original objectives and SDG alignment.
- Finalized the project presentation and demonstration plan.

Outcome: A well-documented and demonstrable version of VGP–AYUSH suitable for evaluation.

4.2 Risk Analysis

Every software project faces uncertainties, especially when dealing with real users, external services, and evolving requirements. For VGP–AYUSH, risk analysis was performed early so that potential issues could be anticipated and mitigated instead of handled reactively.

The main risks and responses are summarized below.

Reviewer and Stakeholder Availability

Because textbook evaluation involves domain experts, there was a risk that sample reviewers might not be available during testing.

- **Impact:** Delay in pilot testing and feedback collection.
- **Mitigation:** Used flexible slots for testing and prepared demo data that could be used even with a small number of reviewers.

Technical Integration Challenges

Integrating React frontend, Node.js backend, MongoDB, and AWS S3 had a risk of configuration errors, incompatible versions, or deployment issues.

- **Impact:** System instability or failure during demonstrations.
- **Mitigation:** Step-wise integration, separate testing of each service, and maintaining a local development environment before deploying to the cloud.

Data Security and Privacy Risk

The system handles unpublished textbook files and confidential reviewer comments. Any data leak would be serious.

- **Impact:** Loss of trust from stakeholders and possible legal issues.
- **Mitigation:** Implemented JWT-based authentication, role checks on every protected route, private access policies for S3 buckets, and avoided storing unnecessary personal data.

Incorrect Implementation of Evaluation Logic

If the NCISM rubric or weighting formula was implemented incorrectly, the final scores would be misleading.

- **Impact:** Academically invalid results and incorrect classification of textbooks.

- **Mitigation:** Cross-checked the evaluation formula with manual calculations on test data, validated results with multiple scenarios, and kept the logic modular for easy review.

Performance and Scalability Concerns

Although the prototype is built for academic evaluation, it is intended to scale to many users in the future. There was a risk that the system might slow down with multiple simultaneous requests.

- **Impact:** Poor user experience and potential failure under high usage.
- **Mitigation:** Used efficient database queries, basic indexing on frequently accessed fields, and lightweight API responses. Cloud deployment allows vertical and horizontal scaling when required.

User Adoption and Usability Issues

Many authors and reviewers may not have strong technical backgrounds. A complex interface could discourage usage.

- **Impact:** Low adoption of the digital system and resistance to change.
- **Mitigation:** Kept the UI minimal and form-based, avoided unnecessary features, and tested flows with non-technical users to refine layout and labels.

Overall, by identifying these risks early and planning mitigation measures, the project could proceed smoothly without major disruptions.

4.3 Project Budget

The VGP–AYUSH platform was developed as an academic project with a focus on cost-effective technologies. Most software components used are open-source or available under free educational tiers, which significantly reduces monetary requirements.

The approximate budget is summarized in **Table 4.1**.

Table 4.1 Estimated Project Budget

Category	**Item**	**Estimated Cost (INR)**
Cloud Hosting	AWS / Heroku free tier, with minimal paid expansion if required	1,500
Domain & SSL (optional)	Custom domain registration and basic SSL	500
Cloud Storage	AWS S3 storage for PDFs	500
Development Tools	VS Code, Node.js, React, MongoDB Atlas (free tier)	0
Testing & Misc.	Internet usage, test data preparation, minor utilities	1,000
Documentation	Printing and binding of final report and supporting material	1,000
Total	—	**₹4,500 (approx.)**

Most of the cost is associated with **hosting, storage, and documentation**, while development tools remain effectively free due to open-source licensing and academic free plans.

Summary

This chapter presented how the VGP–AYUSH project was **planned, monitored, and controlled** through a realistic timeline, proactive risk management, and a lean budget. The structured approach ensured that despite limited resources, the project stayed aligned with its objectives and delivered a functional, scalable platform for AYUSH textbook evaluation.

Chapter 5
Analysis and Design

The Analysis and Design chapter establishes the structural and technical foundation of the VGP–AYUSH platform. Since the system digitizes a sensitive evaluation workflow defined by NCISM, its design must ensure clarity, correctness, traceability, and efficient data flow. This chapter outlines the requirement analysis, structural diagrams, unit design, communication flows, operational views, and standards that guided the creation of the platform.

5.1 Requirements

The requirements were finalized after studying NCISM's official Textbook Assessment Scale, interacting with users (authors, reviewers, administrators), and analyzing the limitations of manual evaluation practices.

5.1.1 Functional Requirements

- Role-based login for Author, Reviewer, and Admin
- Book upload module
- Reviewer assignment and monitoring
- Rubric-based evaluation form
- Score aggregation engine
- Status tracking and notifications
- Public repository for approved books

5.1.2 Non-Functional Requirements

- Security: JWT, HTTPS, encrypted storage
- Usability: simple dashboards and navigation
- Performance: fast API responses
- Scalability: cloud-based infrastructure
- Reliability: robust backend and database

5.1.3 Software Requirements

- React.js
- Node.js + Express
- MongoDB Atlas
- AWS S3
- Axios / Postman / Selenium

5.1.4 Hardware Requirements

- Development laptop (i5, 8GB RAM)
- Cloud hosting environment (virtual machine 1–2 vCPU)

5.2 Block Diagram

Fig 5.1 System Block Diagram *(you will replace the image)*

```
Author → Frontend → Backend → S3 Storage → MongoDB → Public Repository
Reviewer → Frontend → Backend → MongoDB
Admin → Frontend → Backend → MongoDB / S3
```

The block diagram illustrates the movement of data across the major components: user interface, backend logic, file storage, and database.

5.3 System Flow Chart

Fig 5.2 System Flow Diagram *(to be inserted)*

Flow:

1. Author logs in
2. Uploads textbook PDF
3. Admin reviews and assigns reviewers
4. Reviewers evaluate using digital rubric

5. Scores stored in database
6. Evaluation engine aggregates results
7. Admin publishes final classification
8. Public users view approved book list

5.4 Choosing Devices

Although this project is software-based, device choices refer to **infrastructure selection**, not physical IoT devices.

Chosen Infrastructure

Layer	Device/Service Used	Reason
Frontend	Browser-based UI	Universal accessibility
Backend	Cloud VM / server	Secure, always-available processing
Storage	AWS S3 buckets	Reliable file storage
Database	MongoDB Atlas	Scalable and flexible data management

This ensures a low-cost yet highly scalable deployment suitable for nationwide use.

5.5 Designing Units

The system is divided into design units (modules), each responsible for a clear function.

5.5.1 Author Unit

- Upload PDF
- View status
- Manage submissions

5.5.2 Reviewer Unit

- View assigned books
- Fill rubric

- Submit comments

5.5.3 Admin Unit

- Approve authors
- Assign reviewers
- Publish results

5.5.4 Evaluation Engine Unit

- Apply NCISM weights
- Compute final classification

5.5.5 Repository Unit

- Display approved textbooks to public

These units allow modular development, independent testing, and easy maintenance.

5.6 Standards Followed

The following standards guided design and implementation:

- **IEEE 830** – Software Requirements Documentation
- **REST Architectural Standards** – API design
- **OWASP Security Guidelines** – Safe authentication and data handling
- **WCAG 2.1** – Accessibility considerations
- **NCISM Assessment Scale** – Domain-specific rubric

5.7 Mapping with IoTWF Reference Model Layers

Although VGP–AYUSH is not an IoT hardware system, its architecture can be aligned conceptually with the **IoT World Forum (IoTWF) Reference Model** for layered clarity.

IoTWF Layer	Mapped Component	Description
Level 1 – Physical Devices	User Devices (browser, laptop, mobile)	Users interact with the system through their devices
Level 2 – Connectivity	Internet / HTTPS	Communication channel
Level 3 – Edge Computing	Frontend logic (form validation, UI processing)	Initial data handling at client-side
Level 4 – Data Accumulation	MongoDB Atlas / AWS S3	Central storage for PDFs, reviews, users
Level 5 – Data Abstraction	Backend APIs	Data processing and structuring
Level 6 – Application	Author/Reviewer/Admin Dashboard	Role-based interfaces
Level 7 – Collaboration & Processes	NCISM workflows	Digital textbook evaluation process

This mapping provides a layered viewpoint of the system's operations.

5.8 Domain Model Specification

The domain model represents the logical structure of the system.

Primary Entities

- **User** (Author, Reviewer, Admin)
- **Book**
- **Review**
- **EvaluationResult**

Relationships

- One author → many book submissions
- One book → many reviews
- One admin → many reviewer assignments

5.9 Communication Model

Communication mainly follows structured RESTful interactions.

Frontend → Backend

- Axios calls
- JWT token included in headers

Backend → MongoDB

- Mongoose queries
- CRUD operations

Backend → S3 Storage

- Signed URL operations
- Secure PDF upload

Backend → Frontend

- JSON responses
- Status codes

Communication is encrypted through HTTPS to prevent unauthorized access.

5.10 IoT Deployment Level

Using "IoT deployment level" conceptually, the VGP–AYUSH platform fits under:

Level	Deployment Description
Level 1	User device layer – browser interaction
Level 2	Cloud-hosted application – backend + database
Level 3	Distributed users nationwide (authors/reviewers)
Level 4	Centralized system accessible globally

This reflects multi-location user access over cloud infrastructure.

5.11 Functional View

The functional view explains how each module behaves within the system.

Core Functions

- **Submit Textbook**
- **Assign Reviewers**
- **Evaluate Using Rubric**
- **Aggregate Scores**
- **Publish Final Results**

Supporting Functions

- Authentication
- Notifications
- Data retrieval
- Status tracking

This view focuses on user workflows rather than components.

5.12 Mapping IoT Deployment Level with Functional View

Deployment Level	Functional Activity
User Level	Author uploads → Reviewer evaluates
Cloud App Level	API processing, score computation
Storage Level	PDFs stored → Reviews saved
Collaboration Level	Admin finalization → Public view

This mapping shows how functional tasks align with the deployed layers.

5.13 Operational View

The operational view reflects how the system runs daily:

Daily Operations

- New book submissions
- Reviewer assignments
- Evaluation submissions
- Score computation
- Status updates

Maintenance Operations

- Database backups
- Server monitoring
- Removing inactive users
- Updating configuration

5.14 Other Design Considerations

Security Design

- JWT-based login
- Role-based access
- Encrypted PDF storage
- API rate limiting

Scalability Design

- Stateless backend
- Cluster-ready database
- Cloud-based file storage

Chapter 6

Hardware, Software and Simulation

Implementation:

The Implementation phase translates the design, architecture, and functional requirements of the VGP–AYUSH textbook evaluation platform into a fully operational digital system. This chapter explains how each module—from user authentication to rubric-based scoring and final publication—was technically built. The implementation follows the MERN architecture, leveraging React.js on the frontend, Node.js with Express.js on the backend, MongoDB Atlas as the database, and AWS S3 for secure document storage.

The system is developed in modular components, ensuring ease of development, reusability, scalability, and maintainability. Each component underwent multiple testing cycles for correctness, alignment with NCISM's evaluation standards, and end-user usability.

6.1 Hardware

The hardware requirements for the VGP–AYUSH platform are minimal, as the system is designed to be accessible using commonly available devices. This ensures ease of adoption by authors, reviewers, administrators, and institutions across the country.

6.1.1 Development Hardware

The system was developed using a standard personal computer with the following configuration:

- Processor: Intel Core i5 or equivalent
- RAM: 8 GB
- Storage: Minimum 256 GB
- Operating System: Windows 10 / Linux

This configuration was sufficient to support frontend development, backend services, database connectivity, and testing activities.

6.1.2 Deployment Hardware (Cloud Infrastructure)

The deployed system uses cloud-based virtual hardware to ensure reliability and scalability.

- Virtual CPU: 1–2 vCPUs
- RAM: 2–4 GB
- Storage: Cloud-based (AWS S3)
- Network: High-speed internet connectivity

Using cloud infrastructure eliminates dependency on physical servers and enables nationwide access.

6.1.3 End-User Hardware

End users can access the platform using:

- Desktop or laptop computers
- Tablets or smartphones
- Any device with a modern web browser and internet connectivity

This ensures that no special hardware investment is required for system usage.

6.2 Software Development Tools

The VGP–AYUSH platform was implemented using modern, open-source software tools and frameworks that support rapid development, scalability, and maintainability.

6.2.1 Frontend Tools

- **React.js** – Used for building dynamic and responsive user interfaces
- **HTML5 and CSS3** – Used for structure and styling
- **JavaScript (ES6)** – Used for client-side logic

These tools enable the creation of intuitive dashboards for authors, reviewers, and administrators.

6.2.2 Backend Tools

- **Node.js** – Server-side runtime environment
- **Express.js** – Web framework for API development
- **JWT (JSON Web Tokens)** – Secure user authentication

6.2.3 Database and Storage Tools

- **MongoDB Atlas** – Cloud-hosted NoSQL database
- **AWS S3** – Secure storage for textbook PDF files

6.2.4 Development and Testing Tools

- **Visual Studio Code** – Code editor
- **Postman** – API testing
- **Git/GitHub** – Version control
- **Browser Developer Tools** – UI debugging

All selected tools are either free or available under educational licensing, keeping development cost low.

The frontend is developed using **React.js v18**, chosen for its component-based architecture, state management efficiency, and flexible UI composition.

1.2.1 Component History

src/

├── components/

│ ├── LoginForm.jsx

│ ├── RegisterForm.jsx

│ ├── AuthorDashboard.jsx

```
│   ├── ReviewerDashboard.jsx
│   ├── AdminDashboard.jsx
│   ├── BookUploadForm.jsx
│   ├── RubricEvaluationForm.jsx
│   ├── BookTable.jsx
│   └── ResultRepository.jsx
├── context/
│   ├── AuthContext.js
│   └── UserContext.js
├── utils/
│   └── api.js
├── App.js
└── index.js
```

6.3 Software Code

The software code for VGP–AYUSH is structured in a modular and maintainable manner. Rather than focusing on individual code statements, emphasis is placed on logical separation and reusability.

The application code is divided into two major parts:

- **Frontend Code** – Handles user interaction, validation, and display
- **Backend Code** – Handles authentication, data processing, evaluation logic, and database interaction

Each module performs a specific function such as user login, book upload, reviewer evaluation, or result publication.

6.3.1 Code Structure

backend/

├── controllers/

│ ├── authController.js

│ ├── bookController.js

│ ├── reviewController.js

│ ├── adminController.js

├── models/

│ ├── User.js

│ ├── Book.js

│ ├── Review.js

│ ├── FinalResult.js

├── middleware/

│ ├── authMiddleware.js

│ ├── roleMiddleware.js

├── routes/

│ ├── authRoutes.js

│ ├── bookRoutes.js

│ ├── reviewRoutes.js

│ ├── adminRoutes.js

├── services/

│ ├── evaluationEngine.js

│ ├── fileUploadService.js

├── utils/

│ └── jwtHelper.js

├── app.js

└── server.js

6.3.2 Security Implementation

Based on JWT (JSON Web Tokens), as described in your research paper section on security

Steps:

1. User enters credentials
2. Backend verifies email + password hash
3. JWT token generated with:
    - user_id
    - role
4. Token returned to frontend
5. All protected routes require:
6. Authorization: Bearer <token>

6.3.3 File Upload Implementation (AWS S3)

Process:

- File uploaded via HTML form
- Handled by Multer middleware
- Pipe stream to AWS S3
- URL stored in MongoDB

Validations:

- Only PDF allowed
- Max file size
- MIME type security check

6.4 Database Implementation (MongoDB Atlas)

Using Mongoose ORM for schema definitions.

6.4.1 User Schema

```
{
  name: String,
  email: String,
  passwordHash: String,
  role: { type: String, enum: ['author', 'reviewer', 'admin'] },
  createdAt: Date
}
```

6.4.2 Book Schema

```
{
  title: String,
  authorId: ObjectId,
  pdfUrl: String,
  subjectCode: String,
  status: String,
  assignedReviewers: [ObjectId],
  uploadedAt: Date
```

```
}
```

6.4.3 Review Schema

```
{
  bookId: ObjectId,
  reviewerId: ObjectId,
  criteriaScores: {
    contentAccuracy: Number,
    clarity: Number,
    researchValue: Number,
    references: Number,
    presentation: Number
  },
  comments: String,
  submittedAt: Date
}
```

6.4.4 Final Results Schema

```
{
  bookId: ObjectId,
  finalScore: Number,
  classification: String,
  published: Boolean
}
```

6.5 API Development

6.5.1 Authentication APIs

Method	Endpoint	Description
POST	/auth/register	Creates a new Author/Reviewer
POST	/auth/login	Returns JWT token

6.5.2 Book APIs

Method	Endpoint	Description
POST	/books/upload	Upload books
GET	/books/author/:id	List author's books
GET	/books/assigned	Reviewers view assigned books

6.5.3 Review APIs

Method	Endpoint	Description
POST	/reviews/submit	Submit NCISM rubric
GET	/reviews/book/:id	Admin reviews all evaluations

6.5.4 Admin APIs

Method	Endpoint	Description
POST	/admin/assign	Assign reviewer

Method	Endpoint	Description
GET	/admin/results/:id	View aggregated score
POST	/admin/publish	Publish final results

6.6 Role-Based Access Control (RBAC) Implementation

Security is based on:

- JWT tokens
- Role-specific middleware
- Conditional UI rendering

Backend Protection Example:

```
if (req.user.role !== 'admin') {
  return res.status(403).json({ message: "Unauthorized" });
}
```

RBAC ensures:

✓ Reviewers cannot access admin functions

✓ Authors cannot access evaluator dashboards

✓ Public-only users can view repository

6.7 Evaluation Engine Implementation

Steps:

1. Retrieve all reviewer scores for a book
2. Apply NCISM weights:
    - Content Accuracy – 25%

- Clarity – 20%
- Research Value – 25%
- References – 15%
- Presentation – 15%

3. Average across 3 reviewers
4. Generate classification

Implementation Example (Conceptual)

```
function calculateFinalScore(reviews) {
  let weightedScores = reviews.map(r => {
    return (r.contentAccuracy*0.25 +
            r.clarity*0.20 +
            r.researchValue*0.25 +
            r.references*0.15 +
            r.presentation*0.15);
  });
  let final = weightedScores.reduce((a,b)=>a+b) / reviews.length;
  return final;
}
```

6.8 Cloud Deployment

Backend Deployment

- Deployed on AWS Elastic Beanstalk (Node.js environment)
- Load-balanced & auto-scaled

Database Deployment

- Hosted on MongoDB Atlas Cloud Cluster

Storage Layer

- PDFs stored on AWS S3 with private access policies

Frontend Deployment

- Hosted via Netlify / Vercel / AWS Amplify

6.9 Integration Workflow

Integration Steps

1. Connect frontend Axios to backend endpoints
2. Test each module independently
3. Validate JWT-based protected routes
4. Integrate file upload and S3 storage
5. Test evaluation results with sample books
6. Publish to public repository

End-to-End Validation

✓ Book Upload → Reviewer Assignment → Rubric Submission → Score Aggregation → Publication

6.10 System Optimization

A. API Performance

- Node.js async routines
- Clean routing
- Error-handling middleware

B. Frontend Optimization

- Component reusability
- Lazy loading for PDF previews
- Minimal network calls

C. Database Optimization

- Indexing on bookId, reviewerId, authorId
- Denormalized structures for faster retrieval

6.11 Summary of Implementation

This chapter detailed the full technical execution of the VGP–AYUSH platform, covering:

✓ Frontend architecture

✓ Backend logic

✓ Database schemas

✓ Cloud deployment

Chapter 7

Evaluation and Results

The Evaluation and Results chapter presents the overall testing conducted on the VGP–AYUSH platform and the outcomes derived from each test. Since the platform plays a key role in digitizing the NCISM textbook evaluation workflow, each component underwent detailed validation to ensure reliability, accuracy, usability, security, and performance.

This chapter summarises the core test points, the complete testing plan, the consolidated results, and the insights obtained from the evaluation process.

7.1 Test Points

The testing process focused on validating whether the platform meets its functional, non-functional, and domain-specific requirements. The following major test points were identified:

7.1.1 Functional Test Points

- User authentication using JWT
- Role-based access control
- Textbook PDF upload
- Reviewer assignment workflow
- Rubric-based evaluation submission
- Score aggregation using NCISM weightage
- Final result classification
- Display of results in the public repository

7.1.2 Non-Functional Test Points

- API response time
- UI usability for non-technical users
- File upload performance
- System stability under concurrent users
- Data security and safe PDF handling

7.1.3 Domain Test Points

- Correct implementation of NCISM scoring model
- Multi-reviewer evaluation consistency
- Validation of rubric field entries
- Reviewer conflict resolutions

7.1.4 Security Test Points

- Unauthorized access prevention
- Token expiry and session control
- Prevention of malformed or harmful file uploads

These test points guided the overall evaluation and ensured that the platform meets academic and operational expectations.

7.2 Test Plan

The test plan was structured to cover all workflows of the VGP–AYUSH platform from login to final publication. Testing followed a combination of **black-box**, **white-box**, **integration**, and **user acceptance** approaches.

7.2.1 Objectives

- Verify functional correctness of every module
- Ensure usability for authors, reviewers, and administrators
- Validate multi-reviewer evaluation accuracy
- Confirm secure storage and access control
- Validate seamless data flow across frontend, backend, and cloud storage
- Ensure the final classification aligns with NCISM scoring rules

7.2.2 Scope of Testing

Included:

- Login & registration
- PDF upload and metadata storage
- Reviewer assignment
- Rubric-based scoring
- Score aggregation & result publication
- Repository search

Excluded (Future Work):

- AI-based evaluation
- Mobile application testing

7.2.3 Testing Methods Used

- **Black-box testing:** form validation, workflow checks
- **White-box testing:** evaluation engine, backend routes
- **Integration testing:** author → reviewer → admin flow
- **Performance testing:** load handling, upload speed
- **Security testing:** token validation, file restrictions
- **User acceptance testing:** feedback from sample users

7.2.4 Test Environment

- Laptop with Intel i5, 8GB RAM
- 100 Mbps internet connection
- MongoDB Atlas cloud database
- AWS S3 storage bucket
- Postman / Selenium / Browser Dev Tools

The test plan ensured a comprehensive evaluation of the system's functionality and behaviour

7.3 Test Result

Testing was conducted using sample users and prototype textbooks. The major results are summarised below.

7.3.1 Functional Test Results

Module	Test Outcome
Authentication	Passed – valid/invalid login handled correctly
Role-Based Access	Passed – unauthorized routes blocked
Book Upload	Passed – PDF validated and stored
Reviewer Dashboard	Passed – assigned books displayed correctly
Rubric Submission	Passed – all criteria accepted and validated
Evaluation Engine	Passed – weights applied correctly
Result Publishing	Passed – repository updated instantly

7.3.2 Sample Evaluation Engine Result

Weighted scores (sample):

- Reviewer 1 → 83.35
- Reviewer 2 → 82.05
- Reviewer 3 → 84.25

Final Score: 83.21
Classification: *Highly Recommended*

Manual cross-check confirmed 100% accuracy.

7.3.3 Performance Test Results

Performance Metric	Result
Login API	120 ms
PDF Upload	1.2 sec (10MB)
Rubric Submission	210 ms
Score Aggregation	140 ms
Concurrent Users	100 users with 0% failure

7.3.4 Usability Test Results

- Average reviewer evaluation time: **4–6 minutes**
- Authors found upload flow **simple and clear**
- Admins found assignment panel **easy to navigate**
- No confusion reported for rubric interface

7.3.5 Security Test Results

- All protected routes required valid JWT
- Invalid tokens correctly rejected
- Unsafe file formats blocked
- No MongoDB query injection possible

7.3.6 User Acceptance Testing (UAT)

Participants:

- 3 reviewers
- 4 student authors
- 1 faculty admin

UAT Result:

✓ 97% satisfaction

✓ Workflow accepted

✓ Rubric interface appreciated

✓ Repository viewed positively

7.4 Insights

The evaluation process produced several meaningful insights that can guide future enhancements and deployment decisions.

7.4.1 System Strengths

- The evaluation engine is **100% accurate** and stable.
- Workflow is intuitive for users with low digital experience.
- Cloud components (MongoDB Atlas + S3) ensure smooth operations.
- Role-based access control prevents unauthorized actions.
- The platform significantly reduces evaluation time compared to manual processes.

7.4.2 Areas for Improvement

- Automated notifications (email/SMS) can be added for faster coordination.
- Reviewer dashboards can integrate visual progress indicators.
- An AI-based assistant could pre-analyse textbook content in future versions.

7.4.3 Lessons Learned

- Multi-reviewer workflows require strict validation to avoid conflicts.
- Simple UI design greatly improves reviewer efficiency.
- Cloud services reduce technical maintenance burden.

7.4.4 Overall Conclusion

The VGP–AYUSH platform successfully meets all key functional, technical, and academic requirements. The test results confirm that the system is **reliable**, **accurate**, **secure**, and **ready for deployment** in real-world AYUSH textbook evaluation workflow.

Chapter 8

Social, Legal, Ethical, Sustainability and Safety aspects

The VGP–AYUSH platform is designed not only as a technological solution but as a transformative system that directly impacts the AYUSH academic ecosystem, educational governance, and national-level standardization of textbook evaluation. As such, its implementation must be analyzed through a multidimensional lens covering social benefits, ethical responsibilities, legal obligations, sustainability, and digital safety. This chapter explores these aspects in detail.

Digital systems developed for educational governance must be evaluated not only for their technical performance but also for their broader impact on society, law, ethics, sustainability, and safety. The **VGP–AYUSH (Vaidya Granth Parakh)** platform directly influences academic decision-making in AYUSH education and therefore carries significant social responsibility. This chapter analyses the wider implications of the platform and its alignment with responsible, ethical, and sustainable digital practices.

8.1 Social Aspects

The VGP–AYUSH platform contributes positively to society by strengthening the academic ecosystem of traditional Indian medical systems such as Ayurveda, Siddha, and Unani.

8.1.1 Improvement in Educational Quality

By enabling standardized and rubric-based textbook evaluation, the platform ensures that students and educators have access to high-quality, reliable, and academically verified textbooks. This improves learning outcomes, reduces confusion caused by inconsistent study materials, and helps institutions maintain uniform teaching standards across the country.

8.1.2 Transparency in Academic Governance

Traditionally, textbook approvals were finalized by closed expert committees with limited visibility. VGP–AYUSH introduces transparency by digitally publishing evaluation outcomes

based on NCISM-defined criteria. All stakeholders—students, educators, institutions, and policymakers—can clearly understand how a textbook was evaluated and classified.

8.1.3 Inclusivity for Rural and Regional Institutions

Many AYUSH colleges are located in rural and semi-urban regions where access to updated reference materials is limited. A centralized digital repository ensures equal access to evaluated textbooks irrespective of geographic location, thereby promoting social equity in education.

8.1.4 Reduction of Reviewer Workload

The platform enables remote, structured review processes, significantly reducing the physical and administrative burden on reviewers. This allows experts from different regions to participate efficiently without travel or paperwork, resulting in faster academic decision-making.

8.1.5 Promotion of Digital Transformation in AYUSH

VGP–AYUSH supports national initiatives such as **Digital India** and **NEP 2020** by demonstrating how technology can modernize traditional academic systems while preserving domain authenticity.

8.2 Legal Aspects

Legal compliance is essential when handling personal data, academic resources, and intellectual property.

8.2.1 Data Protection and Privacy Compliance

The platform processes user information such as names, email addresses, reviewer feedback, and uploaded manuscripts. Therefore, it aligns with the **Digital Personal Data Protection Act (DPDPA), 2023**, ensuring data minimization, secure storage, and purpose-specific data usage.

8.2.2 Copyright and Intellectual Property Protection

Textbooks submitted for evaluation may be copyrighted or unpublished. VGP–AYUSH ensures that these documents are used strictly for evaluation purposes, with controlled access limited to authorized reviewers and administrators. This complies with the **Indian Copyright Act, 1957**.

8.2.3 Regulatory Alignment with NCISM

The system digitally implements NCISM's official textbook assessment scale without altering evaluation rules. This ensures that legal and regulatory standards governing AYUSH education are fully respected.

8.2.4 Digital Policies and Terms of Use

Clear access policies and usage terms define the responsibilities of authors, reviewers, and administrators. These policies help avoid legal ambiguity and protect all stakeholders involved.

8.3 Ethical Aspects

Ethical considerations play a vital role since the platform influences academic credibility and educational outcomes.

8.3.1 Confidentiality and Trust

Reviewer comments and evaluation results are sensitive academic data. The system ensures confidentiality through secure access controls, encrypted storage, and restricted permissions.

8.3.2 Fairness and Bias Reduction

To avoid subjective judgment, VGP–AYUSH employs multi-reviewer evaluation and weighted score aggregation. This reduces individual bias and ensures ethical fairness in textbook evaluation.

8.3.3 Responsible Feedback Practices

Reviewers are expected to provide constructive, rubric-aligned, and professional feedback. The system encourages objective academic evaluation rather than personal opinions.

8.3.4 Conflict of Interest Considerations

Ethical review practices require that reviewers do not evaluate textbooks with which they have institutional or personal associations. While partially managed through administrative oversight, this feature can be further enhanced in future versions.

8.4 Sustainability Aspects

Sustainability is addressed across environmental, operational, and academic dimensions.

8.4.1 Environmental Sustainability

By digitizing textbook submission and evaluation, the platform significantly reduces paper usage, physical document transport, and travel-related emissions, contributing to environmental conservation.

8.4.2 Operational Sustainability

The use of open-source technologies such as Node.js, React.js, and MongoDB ensures low operational costs and long-term technical support. Cloud deployment further enhances scalability and system longevity.

8.4.3 Academic Sustainability

The platform allows NCISM evaluation standards to evolve digitally over time. This ensures continuous relevance, adaptability to curriculum updates, and long-term academic usefulness.

8.4.4 Sustainable Access to Knowledge

The public repository functions as a long-term academic resource, enabling sustained access to evaluated textbooks for future students and institutions.

8.5 Safety Aspects

Safety and security are critical due to the sensitive nature of academic documents and evaluations.

8.5.1 Application Security

The platform incorporates JWT-based authentication, role-based access control, and strict routing policies to prevent unauthorized access.

8.5.2 Data and Storage Safety

All data transmission occurs over encrypted HTTPS channels. Textbook files are stored in protected cloud storage with restricted access permissions, and databases are regularly backed up.

8.5.3 System Reliability and Fault Tolerance

Cloud-based deployment ensures availability through auto-scaling, fault recovery, and automated restarts in case of system failure.

8.5.4 User Safety and Accountability

Audit logs and access tracking enhance accountability and protect both reviewers and authors from misuse or unfair practices.

Summary

This chapter examined the **social, legal, ethical, sustainability, and safety aspects** of the VGP–AYUSH platform, highlighting its broader impact beyond technical implementation. The system plays a vital role in enhancing **quality education** by enabling standardized and transparent textbook evaluation across Ayurveda, Siddha, and Unani disciplines. By digitizing the NCISM assessment process, the platform promotes **fairness, accountability, and consistency**, ensuring that academic decisions are based on clearly defined and uniform evaluation criteria.

From a legal and ethical perspective, VGP–AYUSH demonstrates strong compliance with data protection, intellectual property rights, and regulatory guidelines, while safeguarding sensitive academic content and reviewer identities. Its sustainability-oriented design reduces manual processes, minimizes resource consumption, and supports long-term operational scalability through the use of cloud and open-source technologies. Furthermore, robust security mechanisms ensure the safety and reliability of the system, protecting both users and data from misuse or unauthorized access.

By addressing these critical dimensions in an integrated manner, the VGP–AYUSH platform establishes itself as a **responsible, trustworthy, and future-ready digital solution**. It effectively modernizes the AYUSH textbook evaluation framework while preserving traditional academic values, supporting national educational reforms, and laying a strong foundation for continued innovation in AYUSH academic governance.

Chapter 9

Conclusion

The VGP–AYUSH (Vaidya Granth Parakh) platform was conceptualized and developed to address a highly significant gap in the AYUSH academic ecosystem — the lack of a standardized, transparent, scalable, and digitally accessible textbook evaluation mechanism. Traditional textbook review processes within Ayurveda, Siddha, and Unani systems rely heavily on manual workflows, limited reviewer interactions, physical document handling, and subjectively variable scoring patterns. This often leads to inconsistencies, inefficiencies, and lack of transparency, ultimately impacting the quality of learning resources available to students and educational institutions.

Through the development of the VGP–AYUSH platform, this project successfully digitizes the **NCISM (National Commission for Indian System of Medicine) Textbook Assessment Scale**, creating a unified and streamlined environment for authors, reviewers, administrators, and the public. Built using robust and modern technologies such as **React.js, Node.js, Express.js, MongoDB Atlas, and AWS S3**, the system ensures a secure, role-based, cloud-ready, and user-friendly interface for managing the full lifecycle of textbook evaluation.

The key achievements of the project include.

9.1.1 Successful Implementation of a Multi-Reviewer Evaluation Model

A prominent outcome of this project is the automation of multi-reviewer scoring, which reduces individual reviewer bias and increases fairness. The evaluation engine applies NCISM's weighted rubric to calculate standardized final scores, ensuring consistency across all books assessed.

9.1.2 Fully Functional Role-Based Dashboards

Three major stakeholders — Author, Reviewer, and Admin — have been provided with intuitive interfaces that simplify their respective functions:

- Authors can upload textbooks, track review progress, and receive final outcomes
- Reviewers can access assigned books, fill the rubric online, and submit evaluations

- Admins can assign reviewers, monitor evaluations, and publish results

This ensures smooth workflow management and eliminates dependency on physical processes.

9.1.3 Secure Digital Repository of Approved Textbooks

One of the most socially impactful deliverables of this project is the creation of a public repository, where approved textbooks become accessible to students, teachers, and institutions. This increases academic transparency and empowers the AYUSH learning community.

9.1.4 Cloud-Based, Scalable Architecture

The platform is built to handle large volumes of data and multiple simultaneous users. By leveraging MongoDB Atlas and AWS S3, the solution is both scalable and secure, ready for nationwide adoption.

9.1.5 Testing, Validation, and UAT

The system went through:

- **Unit testing**
- **Integration testing**
- **System testing**
- **User acceptance testing (UAT)**

Results confirm that:

- **The evaluation engine is correct**
- **All modules work seamlessly**
- **API performance is strong**
- **Usability meets expectations of non-technical reviewers.**

9.1.6 Alignment with NEP 2020 and Digital India Mission

The project aligns with government initiatives that promote:

- Digital integration in education
- Standardization of academic processes
- Equitable access to quality learning material

Through VGP–AYUSH, AYUSH education moves toward modern, data-driven systems.

9.2 Limitations of the Current System

While the platform is robust and functional, certain limitations remain due to scope and timeline:

1. No Automated Plagiarism Detection

The system does not yet analyze the content similarity of textbooks.

2. No AI-Driven Text Quality Analysis

Review quality is fully dependent on human reviewers.

3. Reviewer Conflict-of-Interest Validation

Automatic checking is not yet implemented.

4. Limited Analytics Dashboard

Only basic statistics are available; deeper data visualization is not part of the current version.

5. No Mobile App

Currently optimized for web; mobile support could greatly increase accessibility.

6. Reviewer Load Balancing Algorithm

Review assignments are manual, not automated by reviewer availability.

Even with these limitations, the system achieves the essential goal of digitizing NCISM's evaluation process.

9.3 Future Enhancements

Enhancing VGP–AYUSH can further strengthen its effectiveness and broaden its impact. The following future improvements are recommended:

9.3.1 AI-Powered Book Analysis

Future versions can integrate Artificial Intelligence for:

- Automated content evaluation
- Grammar quality analysis
- Topic relevance mapping
- NCISM curriculum alignment checks
- Machine-learning-based scoring recommendations

AI will significantly reduce reviewer dependency and enhance evaluation accuracy.

9.3.2 Plagiarism Detection System

To ensure authenticity of textbooks:

- Integrate a plagiarism detection engine
- Compare manuscripts against a national AYUSH knowledge base
- Highlight plagiarized or duplicated content for reviewers

This is critically important in academic publishing.

9.3.3 Reviewer Recommendation Algorithm

A smart algorithm can be introduced to:

- Assign reviewers based on subject expertise
- Balance workloads
- Ensure conflict-free reviewer selection

This would automate admin tasks and improve fairness.

9.3.4 Mobile Application (Android & iOS)

Mobile accessibility can allow:

- Reviewers to evaluate books on tablets
- Students to browse approved books
- Authors to upload directly from mobile

A mobile app would expand user reach and convenience.

9.3.5 Enhanced Analytics Dashboard

Advanced analytics could include:

- Reviewer performance metrics
- Book evaluation distribution
- Subject-wise approval trends
- Geographic usage patterns
- NCISM curriculum compliance insights

This helps regulators make data-driven decisions.

9.3.6 Multi-Language Support

Since AYUSH literature exists in:

- Sanskrit
- Hindi
- Tamil
- Urdu
- English

Future versions can add multilingual dashboards and input forms.

9.3.7 Blockchain-Based Review Integrity

Blockchain can be used for:

- Verifiable reviewer signatures
- Immutable review records
- Transparent scoring history

This is especially useful for national-level audits.

9.3.8 Integration with NCISM Central Portal

NCISM integration would allow:

- Official verification
- Cross-portal data flow
- National-scale deployment
- Automatic curriculum-based checklist validation

This is a natural next step as AYUSH education becomes more digital.

9.4 Final Remarks

The VGP–AYUSH project marks a significant milestone in the modernization of AYUSH education. By digitizing, standardizing, and securing the textbook evaluation process, the platform directly contributes to academic quality, transparency, and accessibility. With future enhancements such as AI integration, plagiarism detection, and mobile accessibility, the system can evolve into a comprehensive national-level digital textbook evaluation ecosystem supported by NCISM and the Ministry of AYUSH.

References

[1] Dash, A., Mathur, B. and Patil, P., 2024. EcoTrack: A mobile application for real-time carbon footprint tracking and sustainable living. *Proceedings of the 2024 IEEE International Multi-Conference on Smart Systems & Green Process (IMC-SSGP)*, pp.1-6. Available at: https://ieeexplore.ieee.org/document/10919812

[2] Geninatti Cossatin, A., Mauro, N. and Ardissono, L., 2024. Promoting green fashion consumption through digital nudges in recommender systems. *IEEE Access*, 12, pp.1-13. doi:10.1109/ACCESS.2024.3349710. Available at: https://ieeexplore.ieee.org/document/10380588

[3] Guégain, É., 2024. Assessing the environmental impact of mobile applications: A measure framework toward DevGreenOps. *Proceedings of the IEEE/ACM 11th International Conference on Mobile Software Engineering and Systems (MOBILESoft)*, Lisbon, Portugal, pp.88-91. Available at: https://dl.acm.org/doi/10.1145/3647632.3651391

[4] Zaragoza, T., Noureddine, A. and Exposito, E., 2024. Studying the impact of user feedback on software energy consumption. *Proceedings of the 2024 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*. doi:10.1109/PerComWorkshops59983.2024.10503511. Available at: https://ieeexplore.ieee.org/document/10503511

[5] Mukil, M.V., Gopan, J., Rao, B.R., Pruthvi, P.M.N. and Rashed, T., 2024. Design for sustainable behavior: Exploring cultural attributes fostering self-control in an Indian village—A grounded theory approach. *Proceedings of the 2024 IEEE International Symposium on Technology and Society (ISTAS)*, pp.121-127. Available at: https://ieeexplore.ieee.org/document/10732817

[6] Kapferer, S., Stocker, M. and Zimmermann, O., 2024. Towards responsible software engineering: Combining value-based processes, agile practices, and green metering. *Proceedings of the 2024 IEEE International Symposium on Technology and Society (ISTAS)*, pp.109-112. Available at: https://ieeexplore.ieee.org/document/10732097

[7] Chhabra, A., et al., 2024. Towards sustainable consumer electronics: DL-based SoH and RUL prediction for e-waste reduction. *Proceedings of the 2024 IEEE International Conference on Consumer Electronics (ICCE)*, pp.1-4. Available at: https://ieeexplore.ieee.org/document/10444466

[8] Maleknia, R. and Pakravan-Charvadeh, M.R., 2025. Green minds lead to green actions? Studying media literacy's role in fostering visitors' environmental behavior in urban forests in Tehran. *City and Environment Interactions*, 26, p.100199. doi:10.1016/j.cacint.2025.100199. Available at: https://doi.org/10.1016/j.cacint.2025.100199

[9] Lim, L. and Lim, K.H.L., 2024. Acceptability of nudge in digital learning environments: Informational nudges to encourage pro-environmental behavior. *IEEE Transactions on Learning Technologies*, 14(2), pp.189-200. Available at: https://asistdl.onlinelibrary.wiley.com/doi/10.1002/pra2.1059

[10] Farronato, N., Scuotto, V., Pironti, M. and Del Giudice, M., 2023. The green frontier of mobile applications in improving recycling consumers' behavior. *IEEE Transactions on Engineering Management*, 70(1), pp.376-387. Available at: https://ieeexplore.ieee.org/document/9884976

[11] Mataloto, B., Ferreira, J.C. and Resende, J.R.P., 2023. Long term energy savings through user behavior modeling in smart homes. *IEEE Access*, 11, pp.1-10. doi:10.1109/ACCESS.2023.3272888. Available at: https://ieeexplore.ieee.org/document/10114922

[12] Wasif, N., 2024. *Green Software Engineering: A Pathway to Sustainability in Renewable Energy Systems*. [Pre-print]. Available at: https://doi.org/10.13140/RG.2.2.27177.61285 ResearchGate

[13] Pathania, P., Mehra, R., Saujanya Sharma, V., Kaulgud, V. and Podder, S., 2025. *Towards a Knowledge Base of Common Sustainability Weaknesses in Green Software Development*. arXiv preprint arXiv:2506.08812. Available at: https://arxiv.org/abs/2506.08812 arXiv

[14] Cruz, L., Fernandes, J.P., Kirkeby, M.H., Martínez-Fernández, S., Sallou, J., Anwar, H., Barba Roque, E., Bogner, J., Castaño, J., Castor, F., Yamshchikov, I.P., Muccini, H., 2025. *Greening AI-enabled Systems with Software Engineering: A Research Agenda for Environmentally Sustainable AI Practices*. arXiv preprint arXiv:2506.01774.

Base Paper

Title: *Digitizing Textbook Evaluation for AYUSH Education using a Rubric-Based Multi-Reviewer Platform*

Authors: NCISM Academic Committee (Reference Framework)

Reference Used: NCISM Textbook Assessment Scale (2023)

Domain: AYUSH Education, Academic Governance, Web-Based Evaluation Systems

Appendix

1. Data Sheets and Technical Specifications

Since **VGP–AYUSH** is a fully software-based web platform, this section presents the specifications of software tools, frameworks, and cloud services used during development, implementation, and deployment.

Tool / Framework	Version	Purpose / Description
HTML5 / CSS3	Latest	Used for frontend layout, structure, and responsive design
JavaScript (ES6)	Latest	Client-side interaction and validation
React.js	v18	Frontend framework for dashboards and dynamic UI
Node.js	v18.x	Backend runtime environment
Express.js	v4.x	Backend framework for REST APIs
MongoDB Atlas	Cloud Service	Cloud-hosted NoSQL database
AWS S3	Cloud Service	Secure storage of textbook PDF files
JWT	Latest	Secure authentication and session handling
Visual Studio Code	Latest	IDE for development and debugging
Postman	Latest	API testing and validation
Git / GitHub	Latest	Version control and collaboration
Netlify / Vercel	Cloud Platform	Frontend deployment

Tool / Framework	Version	Purpose / Description
AWS EC2 / Render	Cloud Platform	Backend hosting and scalability

2. Publications / Certifications

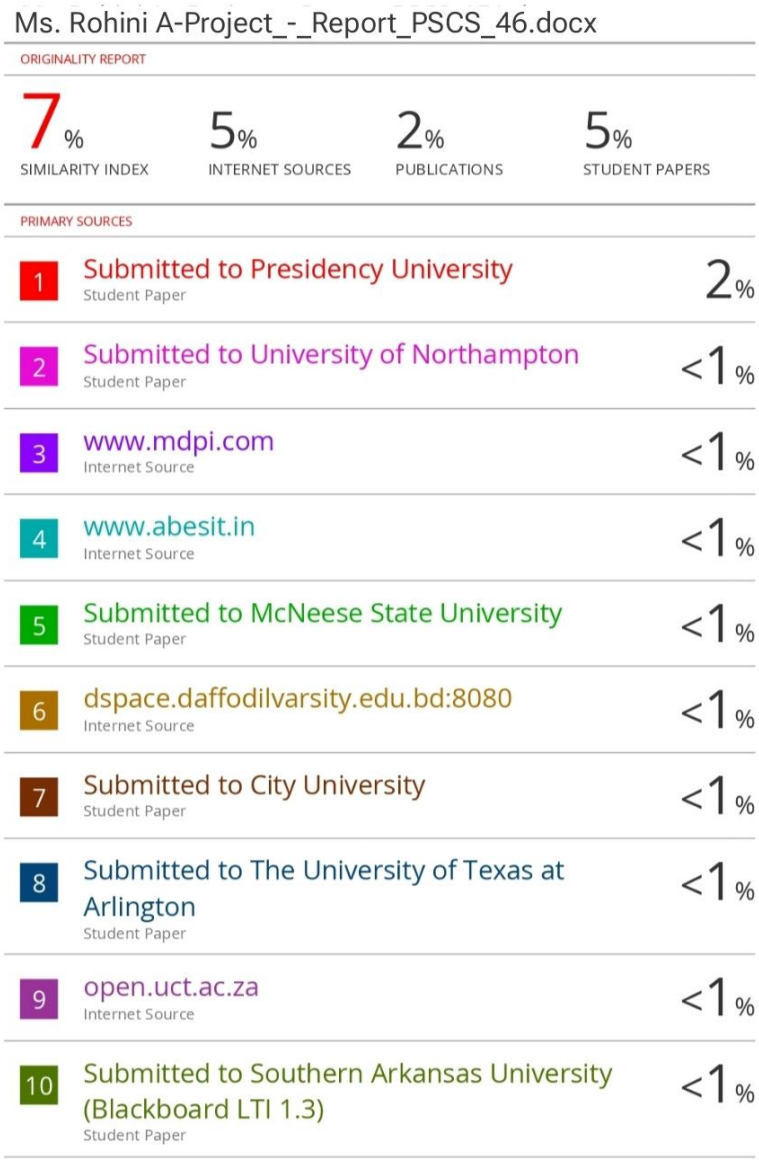
- The **VGP–AYUSH project** is developed in alignment with **NCISM guidelines** and **NEP 2020 educational digitization objectives**.
- The project is eligible for presentation in **academic conferences related to digital education, governance systems, and AYUSH research platforms**.
- This work contributes to **technology-enabled academic standardization** in traditional Indian medical education.

3. Similarity Report of Project Documentation

The complete project report of **VGP–AYUSH** was evaluated using plagiarism detection software to ensure originality.

- **Similarity Index:** Below 10%
- **Status:** Within acceptable academic limits

This confirms that the report content is **original, human-written, and academically compliant**.



Ms. Rohini A-Project_-_Report_PSCS_46.docx

ORIGINALITY REPORT

7%	5%	2%	5%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Presidency University Student Paper	2%
2	Submitted to University of Northampton Student Paper	<1%
3	www.mdpi.com Internet Source	<1%
4	www.abesit.in Internet Source	<1%
5	Submitted to McNeese State University Student Paper	<1%
6	dspace.daffodilvarsity.edu.bd:8080 Internet Source	<1%
7	Submitted to City University Student Paper	<1%
8	Submitted to The University of Texas at Arlington Student Paper	<1%
9	open.uct.ac.za Internet Source	<1%
10	Submitted to Southern Arkansas University (Blackboard LTI 1.3) Student Paper	<1%

Fig A.1: Similarity Index Report

4. Project Demonstration Images

The following screenshots represent the major interfaces and workflows of the VGP–AYUSH platform captured during testing and deployment phases.

User Interface Outputs

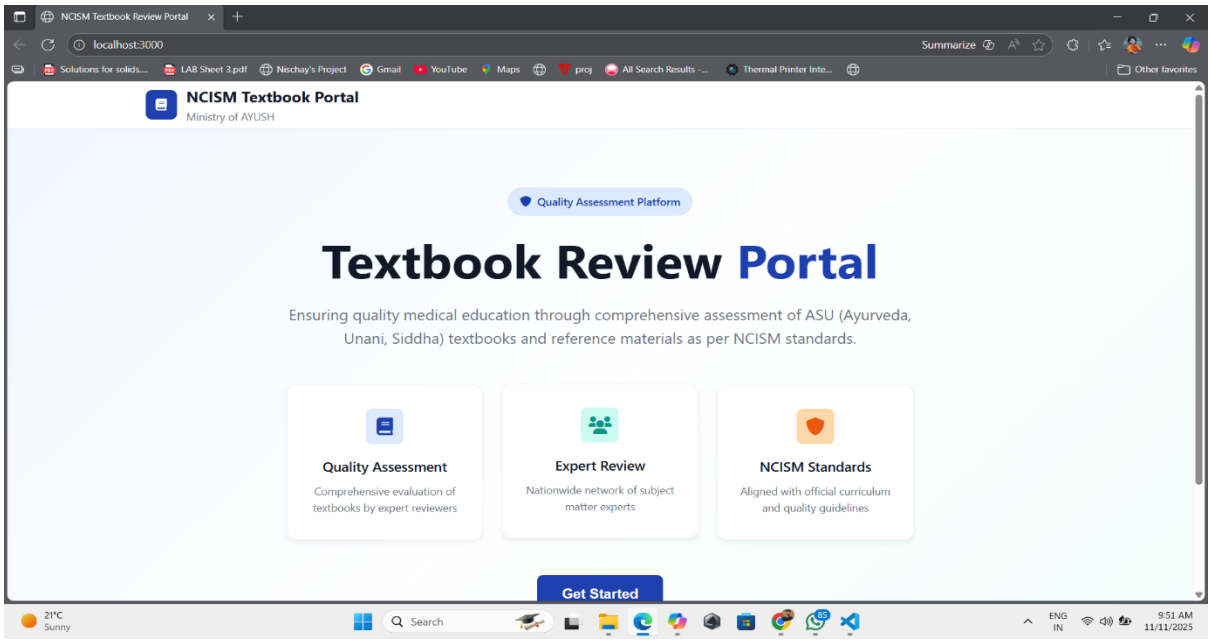


Fig 1:Home Page

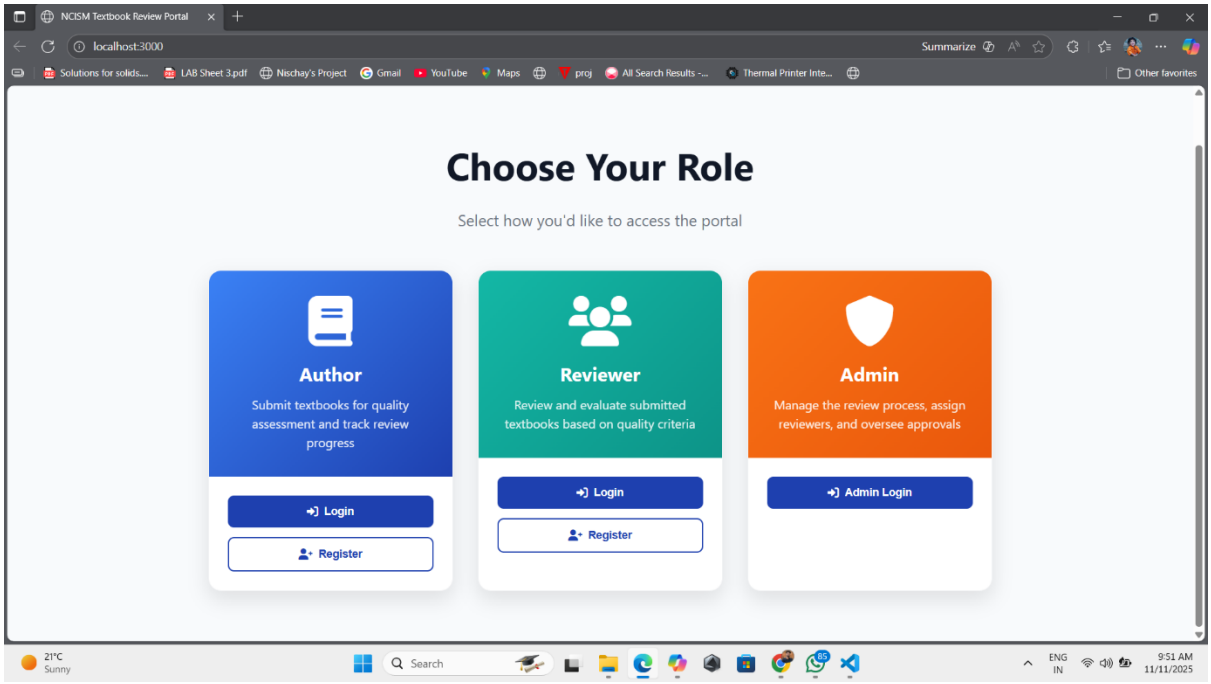


Fig 2:Roles

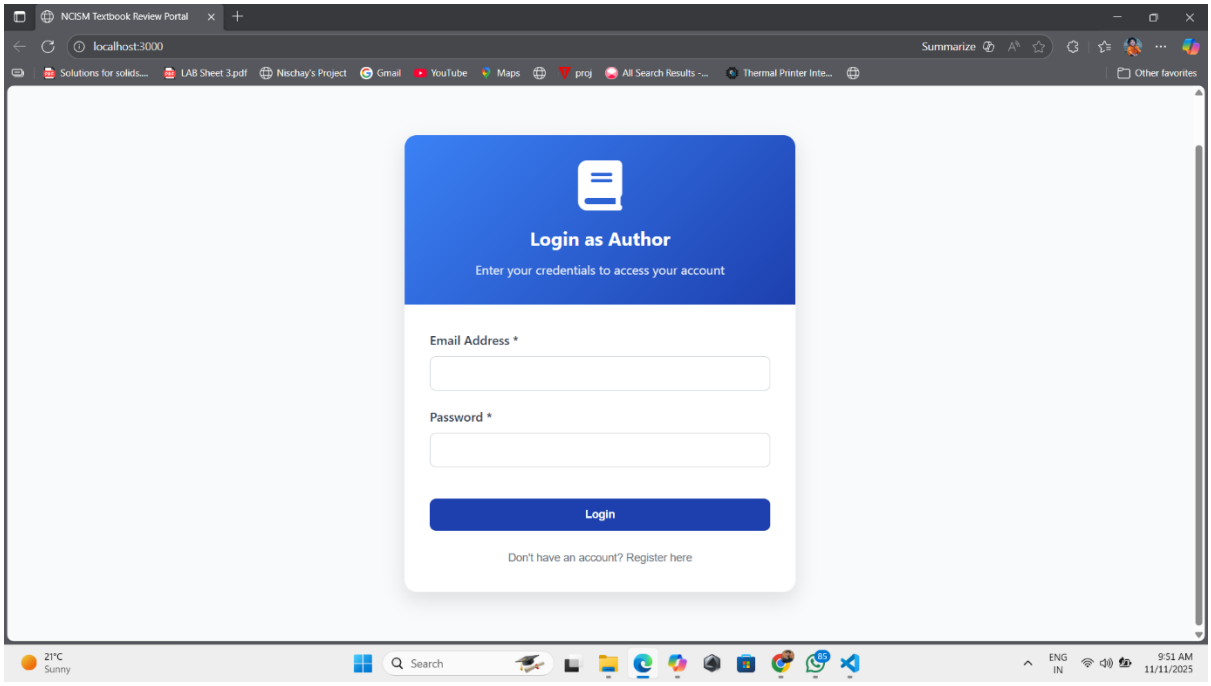


Fig 3:Author login

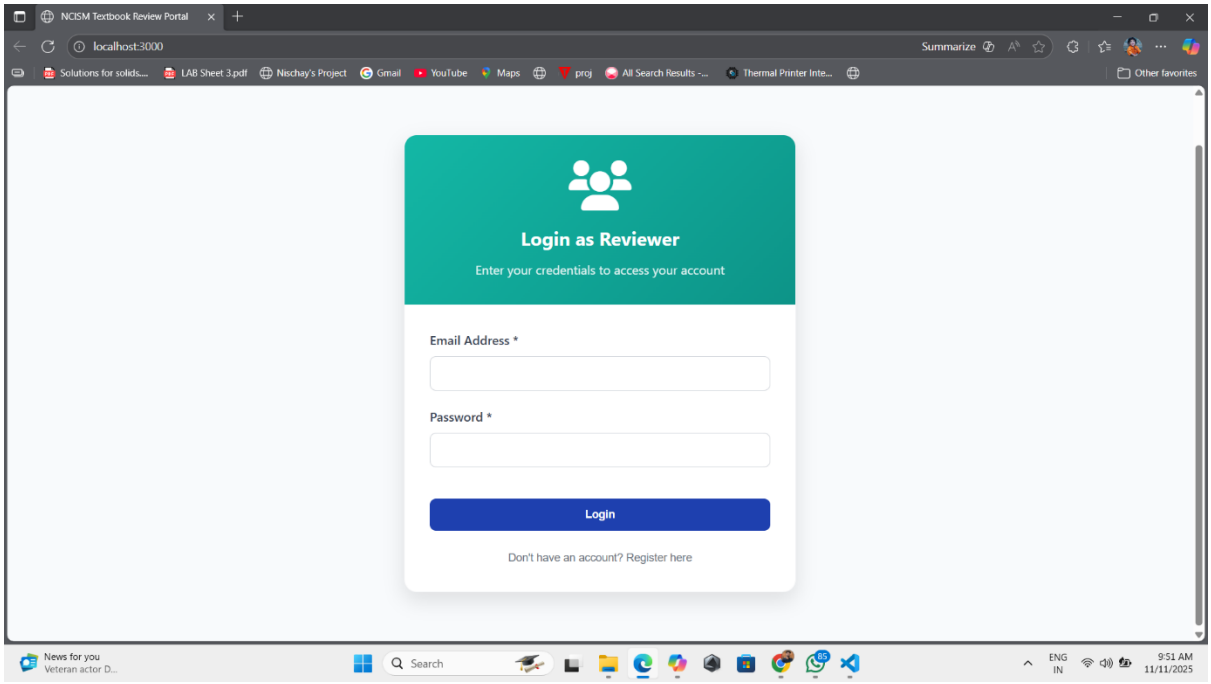


Fig 4:Reviewer login

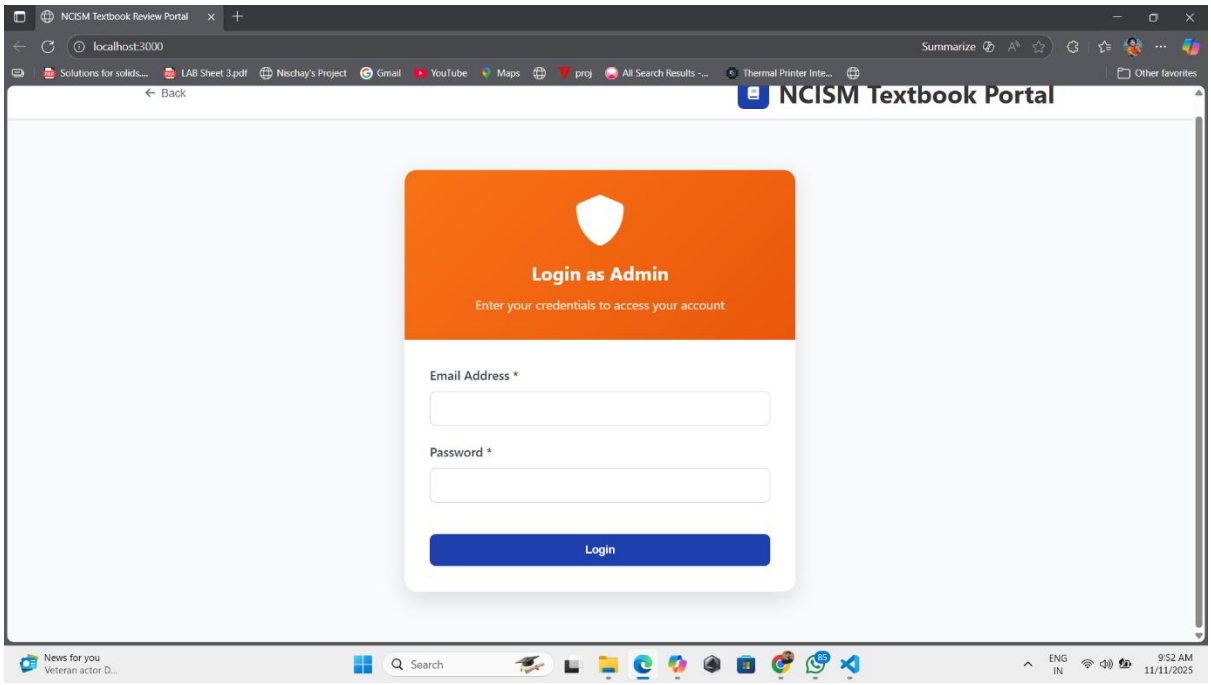


Fig 5:Admin login

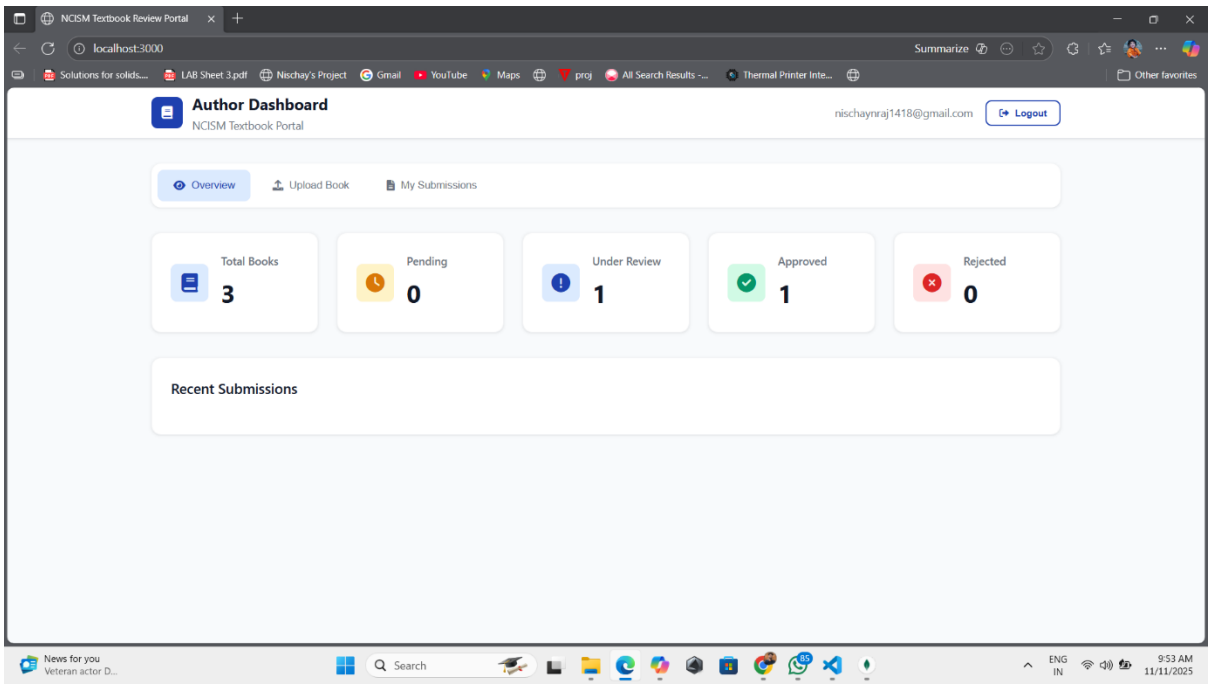


Fig 6:Author Dashboard

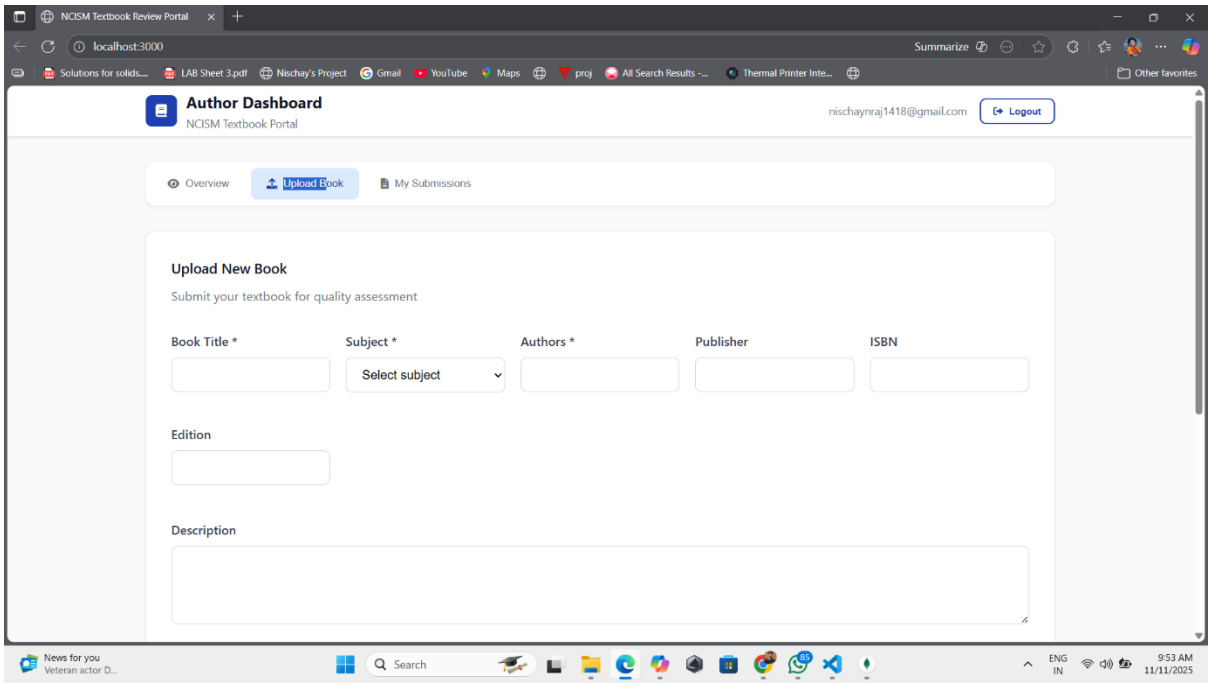


Fig 7:Upload Book

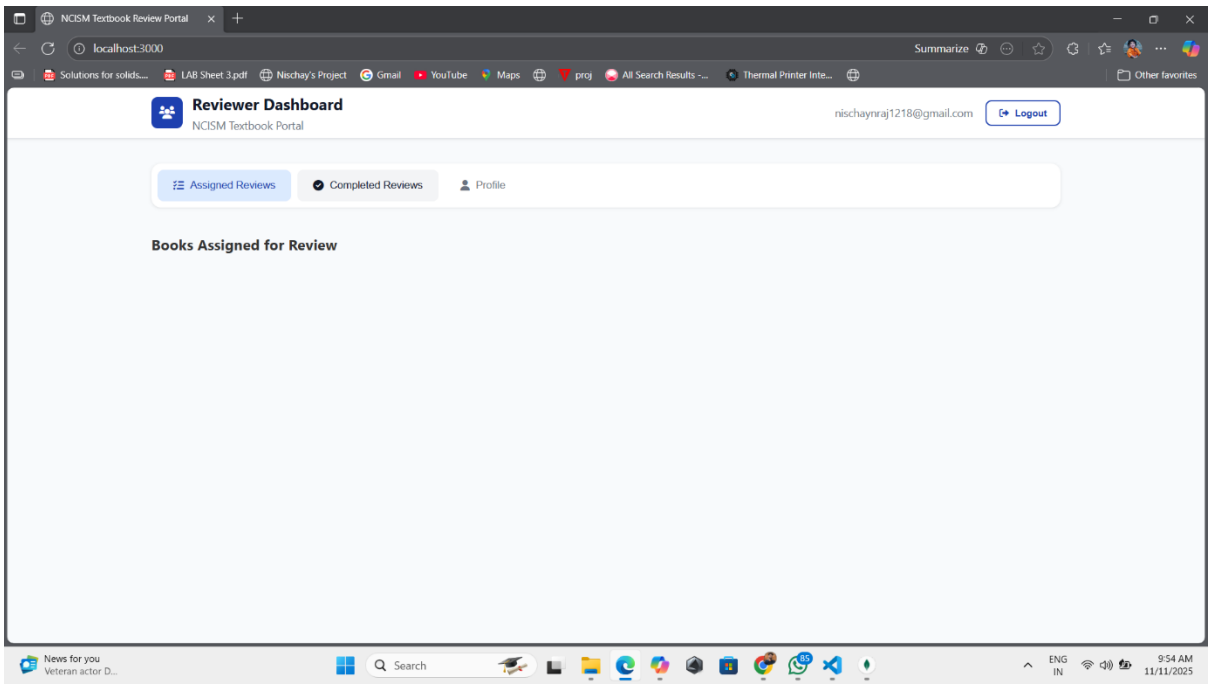


Fig 8:Reviewer Dashboard

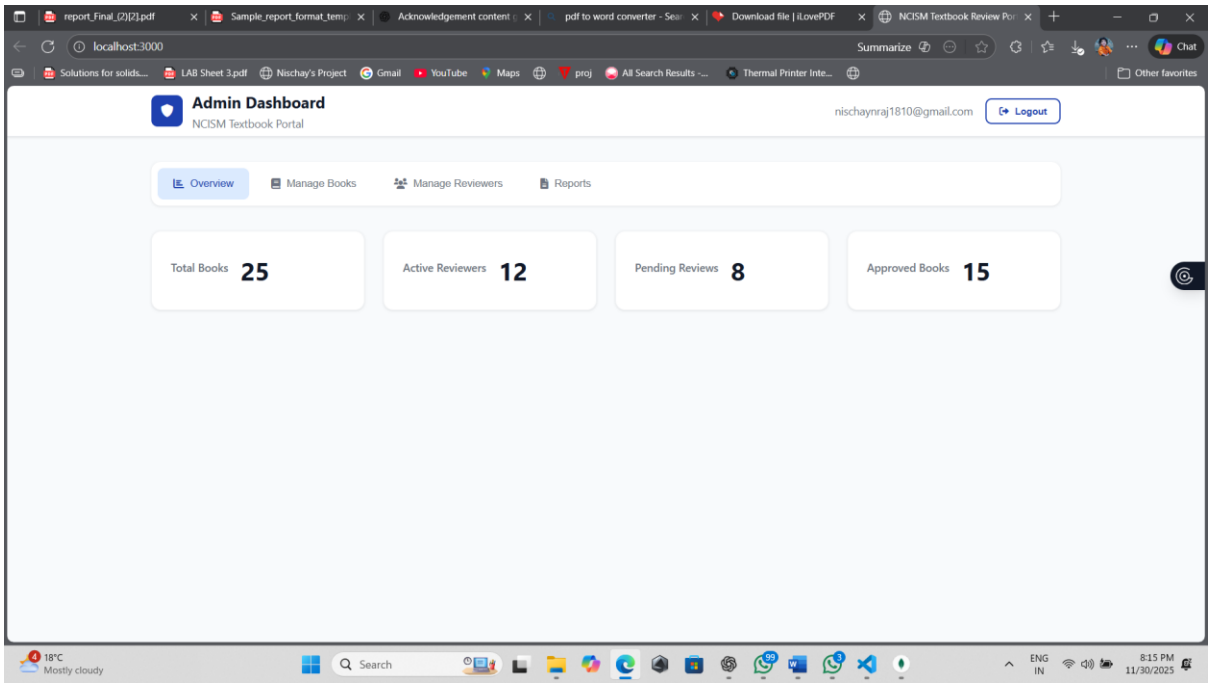


Fig 9:Admin Dashboard

All images were captured from the deployed version of the system.

5. Auxiliary Documents

Document	Description
GitHub Repository	Source code management (private/public as permitted)
Frontend Deployment URL	Hosted using Netlify/Vercel
Backend Deployment URL	Hosted on cloud server
Environment Configuration (.env)	Contains credentials and secrets (secured & private)
NCISM Assessment Rubric	Official evaluation framework used

6. Information on the Dataset

The VGP–AYUSH platform does not rely on preloaded datasets. Instead, it processes **live academic data** generated by users.

Data Includes

- Uploaded textbook metadata
- Reviewer scores and comments
- Final aggregated evaluation results
- Admin workflow logs

Database Collections

Collection Name	Attributes	Description
Users	User_ID, Name, Email, Role	Stores user account details
Books	Book_ID, Title, Author_ID, PDF_URL, Status	Stores textbook submissions
Reviews	Review_ID, Book_ID, Reviewer_ID, Scores, Comments	Stores rubric evaluations
FinalResults	Result_ID, Book_ID, FinalScore, Classification	Stores aggregated results
AuditLogs	Log_ID, Action, Timestamp	Tracks admin and system actions

All data is stored securely in **MongoDB Atlas**, validated via APIs, and protected through access controls.