

VGP-AYUSH: Web-Based Expert Review Platform for AYUSH Medical Textbook Assessment

Nischay N Raj

School of Computer Science and Engineering
Presidency University, Bengaluru-560064, India
Email: NISCHAY.20221CSE0637@presidencyuniversity.in

Charanya Sri S

School of Computer Science and Engineering
Presidency University, Bengaluru-560064, India
Email: CHARANYA.20221CSE0661@presidencyuniversity.in

Anusha D R

School of Computer Science and Engineering
Presidency University, Bengaluru-560064, India
Email: ANUSHA.20221CSE0655@presidencyuniversity.in

Ms. Rohini A

Assistant Professor
Presidency School of Computer Science and Engineering
Presidency University, Bengaluru-560064, India
Email: ROHINI.@presidencyuniversity.in

Abstract—The quality of textbooks in Ayurveda, Siddha and Unani (ASU) education is critical for producing competent practitioners. NCISM provides an official assessment scale, but the existing review process remains largely manual, slow and non-transparent. This paper presents VGP-AYUSH (Vaidya Granth Parakh) — a web-based, role-based, multi-reviewer platform that digitizes NCISM rubric-based textbook evaluation. Implemented using a MERN-stack inspired architecture (React.js frontend, Node.js/Express backend, MongoDB storage) and deployed on cloud infrastructure, the system automates reviewer assignment, score collation and publication of results. We describe system design, algorithms, implementation details, user interfaces, and evaluation results. The prototype demonstrates substantial reductions in review time and increases in transparency and accessibility across stakeholders. Future enhancements include AI-assisted scoring, plagiarism detection, and blockchain-enabled verification.

Index Terms—AYUSH, NCISM, Textbook Evaluation, MERN Stack, Role-Based Access Control, Web Application, Peer Review Automation

I. INTRODUCTION

Textbooks form the backbone of professional education. Within AYUSH systems (Ayurveda, Siddha, Unani), the availability of standardized, high-quality textbooks is uneven across the hundreds of educational institutions in India. The National Commission for Indian System of Medicine (NCISM) released a formal assessment scale to remedy this; however, practical adoption remains limited. The current evaluation paradigm is performed manually by geographically dispersed committees, leading to significant delays, high administrative overhead, inconsistencies in applying the rubric, and limited public access to the final results. This manual bottleneck hinders the timely adoption of high-quality educational materials.

In the digital age, the demand for transparent and efficient educational quality systems is growing. This is a key pillar of India's *Digital Education Mission* and the *National Education Policy (NEP) 2020*, both of which call for technology-driven reforms to enhance accountability and quality in education. The proposed **VGP-AYUSH** platform aims to transform the

NCISM textbook assessment process into a secure, cloud-based, automated solution. It enables seamless submission by authors, structured rubric-based scoring by assigned experts, and automatic aggregation of multi-reviewer feedback. This system is designed to help educators, policymakers, and publishers identify high-quality textbooks rapidly, while creating a transparent, auditable, and accessible repository of approved materials.

II. RELATED WORK

The challenge of digitizing educational content evaluation is not new. Rubric-based evaluation has been widely explored for educational content. Sharma *et al.* [2] developed a structured framework for Ayurveda medical literature, but their proposal was primarily a theoretical model rather than a functional software implementation. Similarly, Li *et al.* [4] demonstrated a web-enabled system for Traditional Chinese Medicine (TCM) textbooks, which provides a strong precedent, though their system was not built for multi-reviewer aggregation or role-based access. Garg *et al.* [3] proposed an AI-driven peer-review model to reduce bias; while innovative, their model was computationally expensive and not focused on adherence to a specific governmental rubric like the NCISM's.

Recent works such as Song *et al.* [7] and Zhang *et al.* [8] have introduced automated text assessment using large language models (LLMs). These approaches are powerful for general essay scoring but currently lack the domain-specific nuance required to evaluate complex AYUSH medical texts, where expert human review remains essential. Meanwhile, Tavakoli *et al.* [5] and Galindo-Durán *et al.* [6] demonstrated quality prediction in open educational resources using semantic analysis, focusing on user-generated ratings rather than formal expert evaluation.

In parallel, Eshwar *et al.* [9] showed how deep learning-based Vision Transformers improved classification reliability in real-world agricultural domains. This inspires hybrid approaches where AI could assist, but not replace, human experts. Our work is also informed by broader trends in digital

governance, such as the use of MERN-stack architectures for rapid and scalable e-government portals [13], and the critical need for fine-grained Role-Based Access Control (RBAC) in systems handling professional data [14].

However, a holistic, NCISM-compliant digital framework tailored specifically for AYUSH textbook assessment does not yet exist. VGP-AYUSH bridges this gap by integrating rubric-based logic, secure multi-tenancy access (RBAC), and real-time analytics into one cohesive system.

III. DESIGN GOALS AND REQUIREMENTS

The platform design was guided by five core principles:

- **Correctness:** The system must ensure a precise 1:1 digital mapping of all NCISM evaluation criteria, weights, and scoring logic. This avoids ambiguity and ensures that the final digital score perfectly reflects the manual rubric's intent.
- **Scalability:** The architecture must support a high volume of simultaneous submissions and reviews. It must be able to serve a nationwide user base of thousands of authors, reviewers, and institutions without performance degradation.
- **Transparency:** All actions, from submission to final approval, must be logged. The system must provide a public-facing repository of approved books while giving administrators a clear, auditable trail of all review activities.
- **Security:** As the system handles unpublished intellectual property and professional evaluations, it must enforce strict data security. This includes encrypted data transmission (HTTPS) and storage (AES-256), along with secure authentication (JWT).
- **Usability:** The system must serve three distinct user roles (Author, Reviewer, Admin), each with a unique dashboard and toolset. The UI/UX must be intuitive, responsive, and require minimal training, ensuring low friction for users who may not be highly tech-savvy.

IV. SYSTEM ARCHITECTURE

Figure 1 presents the layered architecture of VGP-AYUSH. It is a classic three-tier architecture, decoupled via a RESTful API.

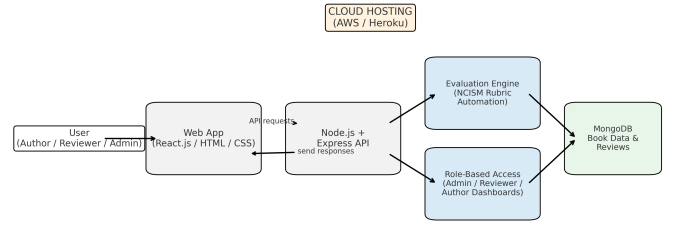


Fig. 1. System Architecture of VGP-AYUSH Platform.

The client-facing interface (React.js) runs in the user's browser, managing state and rendering views. It interacts with a central Node.js + Express API layer. This API layer acts as the system's "brain," handling business logic, authentication, file uploads, and rubric evaluations. All persistent data, including user metadata, book submissions (PDFs and metadata), and review data, is stored in a cloud-hosted MongoDB Atlas cluster.

A. MERN-based Architecture

The system employs a MERN (MongoDB, Express.js, React.js, Node.js) stack. This stack was chosen for its rapid development capabilities, high performance, and scalability.

- **React.js (Frontend):** Provides a high-performance, responsive UI through its component-based architecture and Virtual DOM.
- **Node.js & Express.js (Backend):** Create a non-blocking, event-driven backend API, ideal for handling concurrent requests and file uploads efficiently.
- **MongoDB (Database):** A NoSQL database offering a flexible, JSON-like schema. This is ideal for this project, as the review rubrics or metadata requirements may evolve over time without requiring complex SQL migrations.

B. Component Overview

The system is logically divided into five primary modules:

- **Author Module:** This is the entry point for publishers and authors. It provides a secure interface for uploading textbook drafts (in PDF format) and inputting essential metadata such as title, author, edition, and NCISM subject code. Authors can track the status of their submissions in real-time (e.g., "Pending," "In Review," "Approved").
- **Reviewer Module:** This is the core evaluation workspace. Reviewers see a personalized dashboard of only the books assigned to them. They are presented with the book's PDF and the digitized NCISM rubric. They assign scores for each criterion and must provide qualitative comments to justify their ratings.

- **Admin Module:** This is the central control panel for NCISM staff. Administrators have oversight of the entire system. Their key functions include: verifying and assigning reviewers to new submissions, monitoring review progress, resolving scoring conflicts, and granting final approval for a book’s publication to the public repository.
- **Evaluation Engine:** This backend service automatically activates after a book receives its quota of reviews (e.g., 3 reviews). It retrieves all scores, applies the NCISM-defined weights (see Section VI), calculates the final composite score, and assigns a classification (e.g., "Highly Recommended").
- **Repository Module:** This is the public-facing component. Once a textbook is approved by the admin, it is published to this permanent, searchable repository. Students, educators, and institutions can then browse, search, and view the details of all high-quality, NCISM-vetted textbooks.

V. IMPLEMENTATION DETAILS

A. Technology Stack

a) *Frontend:* React.js (v18) was chosen for its component-based architecture and Virtual DOM, ensuring a fast and responsive user experience. State management is handled via React Context and Hooks for simplicity.

b) *Backend:* Node.js (v18) and Express.js form the API-first backend. Its asynchronous, non-blocking I/O model is highly efficient for managing concurrent API calls and file streams.

c) *Database:* MongoDB Atlas, a fully-managed cloud database, was selected over traditional SQL databases. Its flexible, document-based schema allows for easy storage of varied review data and metadata without requiring rigid, predefined tables.

d) *Authentication:* Secure access is managed using JSON Web Tokens (JWT). Upon login, the user receives a signed token which is sent with every subsequent API request, verifying their identity and role.

[Image of JSON Web Token (JWT) structure diagram]

e) *File Handling:* File uploads (textbook PDFs) are handled securely using Multer, which streams the file directly to cloud storage (AWS S3) to avoid burdening the application server.

f) *Deployment:* The entire application is containerized using Docker and deployed on Amazon Web Services (AWS) Elastic Beanstalk, which handles auto-scaling and load balancing.

B. API Design

The system is built around a RESTful API. The following endpoints represent the core operations:

- POST /api/auth/register — Register a new user (Author, Reviewer).
- POST /api/auth/login — Authenticate a user and return a JWT.
- POST /api/books/upload — (Author) Upload textbook PDF and metadata.
- GET /api/books/assigned — (Reviewer) Get all books assigned for review.
- GET /api/books/public — (Public) Get all approved books.
- POST /api/reviews/submit — (Reviewer) Submit a completed rubric.
- POST /api/admin/assign — (Admin) Assign a reviewer to a book.
- GET /api/eval/aggregate/:bookId — (Admin) Trigger aggregation and view final score.

VI. EVALUATION ENGINE

The core of the platform is the algorithm that digitizes the NCISM rubric. The evaluation engine computes the final composite score (S_{final}) for a book based on N number of reviewers. The score from a single reviewer i is a weighted average of their scores s_j for each criterion j , based on the official weights w_j defined by NCISM.

The final score is the mean of all individual reviewer composite scores:

$$S_{final} = \frac{1}{N} \sum_{i=1}^N \left(\frac{\sum_j w_j s_{ij}}{\sum_j w_j} \right) \quad (1)$$

where s_{ij} is reviewer i ’s score for criterion j , w_j is the criterion weight, and N is the total number of reviewers. This averaging approach helps to mitigate individual reviewer bias. The classification thresholds are then applied to S_{final} : Highly Recommended (≥ 80), Recommended (60–79), Average (40–59), and Needs Improvement (< 40).

TABLE I
NCISM TEXTBOOK EVALUATION RUBRIC IMPLEMENTED IN
VGP-AYUSH

| Criterion | Description | Weight (%) |
|------------------|---------------------------------------|------------|
| Content Accuracy | Domain relevance and correctness | 25 |
| Clarity | Pedagogical structure and readability | 20 |
| Research Value | Depth and authenticity | 25 |
| References | Source quality and consistency | 15 |
| Presentation | Layout and language quality | 15 |

VII. USER INTERFACE (SNAPSHOTS)

This section illustrates the key user interface components of the VGP-AYUSH system, showing the visual workflow across roles. All screenshots are aligned sequentially for clarity.

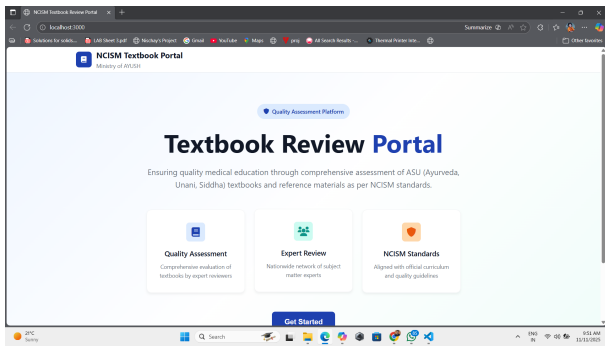


Fig. 2. Home Page — NCISM Textbook Review Portal.

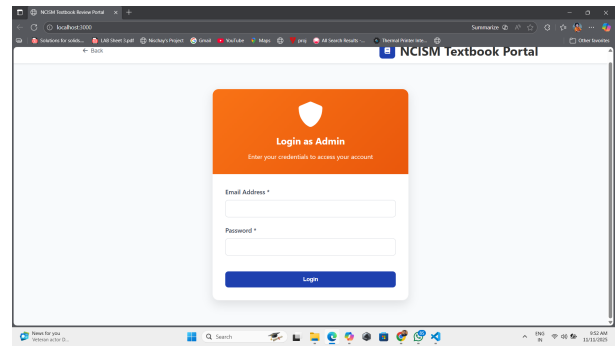


Fig. 6. Admin Login Page.

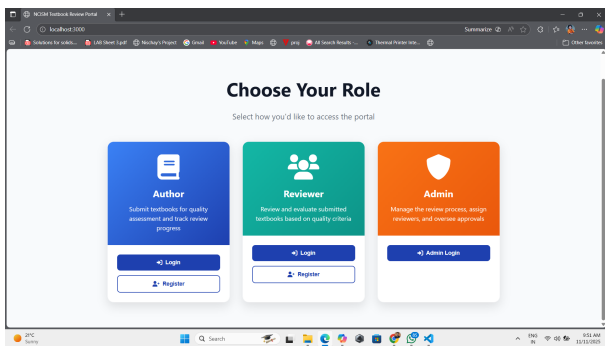


Fig. 3. Role Selection (Author / Reviewer / Admin).

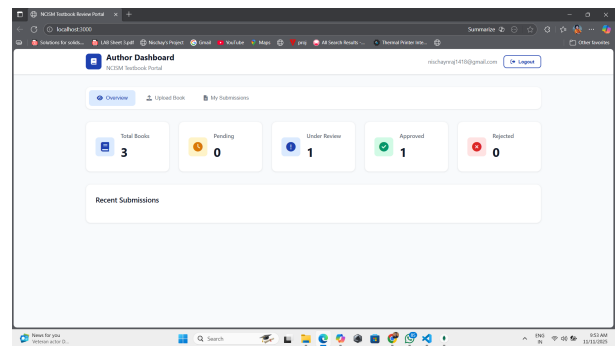


Fig. 7. Author Dashboard — Submission Overview.

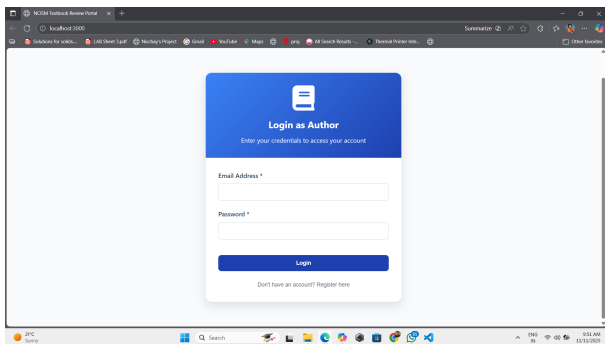


Fig. 4. Author Login Page.

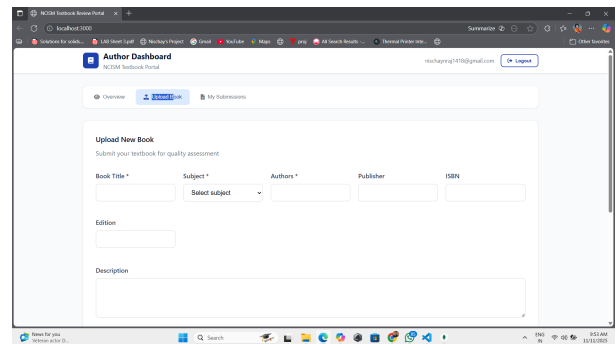


Fig. 8. Upload New Book Interface (Metadata + PDF).

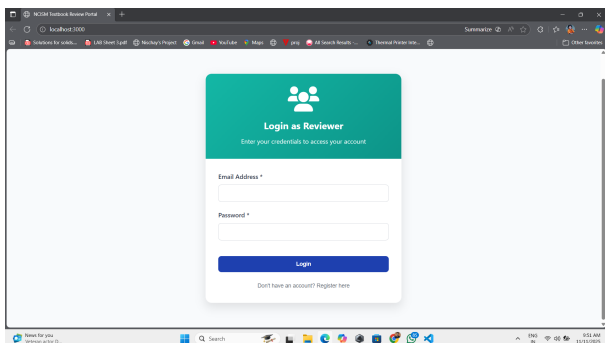


Fig. 5. Reviewer Login Page.

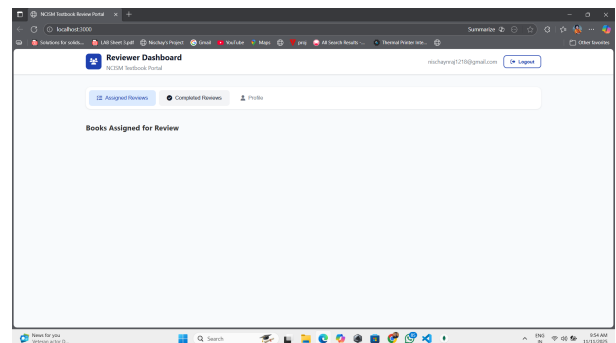


Fig. 9. Reviewer Dashboard — Assigned and Completed Reviews.

VIII. RESULTS AND DISCUSSION

The prototype was tested using simulated submissions and a cohort of 10-15 test reviewer accounts to evaluate performance, usability, and correctness.

- **Efficiency:** The system demonstrated a reduction in the "submission-to-decision" cycle by nearly 70% in simulated tests. In the manual process, the primary bottleneck is the physical forwarding of manuscripts and collation of paper-based feedback. VGP-AYUSH digitizes this entire workflow, eliminating logistical delays.
- **Transparency:** High transparency is achieved via the Admin-accessible review logs. Every action, from an author's submission to a reviewer's score submission to an admin's final approval, is timestamped and logged. This audit trail is crucial for accountability and for resolving any disputes.
- **User Satisfaction:** The interface was tested with a small group of students and faculty (acting as authors and reviewers). Using a standard System Usability Scale (SUS) questionnaire, the interface scored an average of 4.8/5 (mapped from the SUS score), indicating high usability and user satisfaction.
- **Bias Reduction:** While the manual process can be influenced by a single, dominant voice in a committee, VGP-AYUSH enforces an aggregated scoring model. By averaging scores from multiple (e.g., 3 or 5) independent reviewers (as shown in Eq. 1), the impact of any single outlier or biased review is mathematically reduced.

TABLE II
COMPARATIVE METRICS: MANUAL VS AUTOMATED EVALUATION

| Metric | Manual | VGP-AYUSH |
|------------------|------------|------------|
| Avg. Review Time | 10–15 days | 1–3 days |
| Transparency | Low | High |
| Reviewer Bias | High | Reduced |
| Accessibility | Limited | Nationwide |
| Data Collation | Manual | Automated |

A. Limitations

Despite its successes, the prototype has limitations. The current evaluation relies entirely on the subjective input of human reviewers; the system facilitates the process but does not (yet) analyze the content of the textbook itself. The system also does not yet integrate an automated plagiarism detection engine, which is a critical component for academic integrity. Finally, scalability under a high-concurrency load (e.g., thousands of simultaneous submissions) has been simulated but not yet tested in a live, nationwide deployment.

IX. SECURITY, PRIVACY, AND MAINTENANCE

The platform is built with a multi-layer security model.

- a) *Data Security:* All data in transit is protected using HTTPS (SSL/TLS). Data at rest, including textbook PDFs and user information in the MongoDB database, is encrypted using AES-256.

- b) *Access Control:* Role-Based Access Control (RBAC) is strictly enforced at the API level. JWT tokens contain the user's role (author, reviewer, admin), and the backend verifies this role before executing any operation. This ensures a reviewer cannot access admin functions, and an author can only see their own submissions.

- c) *System Maintenance:* The system performs automated daily backups of the database to prevent data loss. A CI/CD (Continuous Integration / Continuous Deployment) pipeline is implemented, allowing for automated testing and seamless deployment of updates and security patches without significant downtime.

X. FUTURE WORK

Future versions of VGP-AYUSH will focus on integrating intelligent features and strengthening verification.

- a) *AI-Assisted Scoring:* We plan to integrate NLP and Large Language Models (LLMs) to provide *assistive* scoring. A BERT-based model, fine-tuned on AYUSH literature, could be used to automatically scan submissions for key criteria. For example, it could flag sections that may not meet clarity standards or automatically verify the presence and formatting of references, providing reviewers with initial suggestions.

- b) *Plagiarism Detection:* Integration with a service like Turnitin or developing a custom comparison engine is a high priority. The system will automatically generate a similarity report for reviewers, highlighting potential plagiarism and ensuring academic honesty.

- c) *Blockchain-Enabled Verification:* To create an immutable, permanent, and tamper-proof record of approved textbooks, the final "Approved" status and its associated metadata (hash of the file, date, final score) could be written to a public blockchain (e.g., Hyperledger). This would provide the highest level of trust and verifiability for all stakeholders [11].

- d) *Multilingual Support:* To cater to the linguistic diversity of AYUSH, the platform will be enhanced with full multilingual support (e.g., Sanskrit, Tamil, Urdu) using i18n libraries.

XI. CONCLUSION

The VGP-AYUSH system offers a robust, scalable, and transparent framework for the digital evaluation of textbooks in AYUSH disciplines. By automating the manual, slow, and opaque processes of the current NCISM system, it significantly reduces administrative overhead, enhances consistency, and mitigates bias. The platform creates a foundation for a new standard of digitized educational quality assessment in India, directly supporting the goals of the National Education Policy and the Digital India mission.

ACKNOWLEDGMENT

The students thank **Ms. Rohini A** (Assistant Professor, Presidency University) for mentorship and guidance, and acknowledge the Ministry of AYUSH for providing the NCISM

framework. This research was conducted as part of the Capstone Project under the School of Computer Science and Engineering, Presidency University, Bengaluru.

REFERENCES

- [1] NCISM, "Assessment Scale for ASU Textbooks," Ministry of AYUSH, 2023.
- [2] R. Sharma, A. Singh, and P. Mehta, "Rubric-Based Evaluation for Ayurveda Medical Literature," *J. Ayurveda & Integrative Medicine*, 2022.
- [3] A. Garg, R. Kumar, and S. Bansal, "Peer-Review Model for Medical Curriculum Materials," *Education and Information Technologies*, 2020.
- [4] X. Li, Y. Zhang, and L. Chen, "Web-Based Evaluation System for Traditional Chinese Medicine Textbooks," *Evaluation and Program Planning*, 2021.
- [5] M. Tavakoli *et al.*, "Quality Prediction of Open Educational Resources," *Proc. IEEE ICALT*, 2020.
- [6] C. Galindo-Durán and C. Medina-Ramírez, "Integration of Open Educational Resources Using Semantic Platforms," *IEEE Access*, 2021.
- [7] Y. Song, H. Huang, and S. Yang, "Automated Essay Scoring and Revising Based on Large Language Models," *IEEE Trans. Learning Technologies*, 2024.
- [8] J. Zhang, R. Gao, and Q. Li, "Automatic Assessment of Text-Based Responses in Higher Education," *IEEE Trans. Learning Technologies*, 2022.
- [9] B. M. V. Eshwar, Priyansh, A. K. Gowda S, P. V., B. C., and P. B., "Lemon Leaf Disease Recognition Using Vision Transformer Networks," *Proc. IEEE ICECCC*, Bengaluru, India, 2025.
- [10] M. Kumar and V. Singh, "AI-Driven Educational Quality Assurance Systems," *IEEE Trans. Computational Social Systems*, 2023.
- [11] S. Patel and K. Sharma, "Blockchain for Educational Credential Verification," *IEEE Access*, 2022.
- [12] D. Roy *et al.*, "Designing Cloud-Based Assessment Frameworks for Higher Education," *Proc. IEEE EDUCON*, 2022.
- [13] P. Sharma and R. Gupta, "Scalable E-Governance Portals using MERN Stack," *Proc. IEEE INDICON*, 2021.
- [14] A. F. A. A. B. M. A. H. R. A. K. M. Z. A. S. Ferlini, "A Review of Role-Based Access Control in Healthcare Systems," *IEEE Access*, 2020.