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

# Application for Assessment of Quality of Textbook/ Reference Books/E-Books

**A PROJECT REPORT**

***Submitted by***

SYED THOUSIF- 20221CSE0234

KAMBAM NITHIN KUMAR REDDY- 20221CSE0211

PATIL JASHWANTH REDDY- 20221CSE0213

***Under the guidance of,***

**Mrs. Kummara Sruthi**

**Assistant Professor**

**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

# PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

## BONAFIDE CERTIFICATE

Certified that this report "Application for Assessment of Quality of Textbook/Reference Books/E-Books" is a bonafide work of "Syed Thousif (20221CSE0234), Kambam Nithin Kumar Reddy (20221CSE0211), Patil Jashwanth Reddy (20221CSE0213)", 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 AND ENGINEERING during 2025-26.

**Mrs. Kummara Sruthi**
Project Guide
PSCS
Presidency University

**Mr. Muthuraju V**
Program Project Coordinator
PSCS
Presidency University

**Dr. Sampath A K**
**Dr. Geetha A**
School Project Coordinators
PSCS
Presidency University

**Dr. Blessed Prince**
Head of the Department
PSCS
Presidency University

**Dr. Shakkeera L**
Associate Dean
PSCS
Presidency University

**Dr. Duraipandian N**
Dean
PSCS & PSIS
Presidency University

Examiners

| Sl. no. | Name | Signature | Date |
|---|---|---|---|
| 1 | Mr. Laxman | | |
| 2 | Mr. Likhith S R | | |

# 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 Syed Thousif, Kambam Nithin Kumar Reddy, Patil Jashwanth Reddy, hereby declare that the project work titled **"Application for Assessment of Quality of Textbook/ Reference Books/E-Books "** has been independently carried out by us and submitted in partial fulfillment for the award of the degree of B.Tech in COMPUTER SCIENCE ENGINEERING during the academic year of 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.

| | | |
|---|---|---|
| Syed Thousif | USN: 20221CSE0234 | Signature 1 |
| Kambam Nithin Kumar Reddy | USN: 20221CSE0211 | Signature 2 |
| Patil Jashwanth Reddy | USN: 20221CSE0213 | Signature 3 |

PLACE: BENGALURU

DATE:

# ACKNOWLEDGEMENT

For completing this project work, We/I have received the support and guidance from many people whom I would like to mention with a 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 the completion of the project.

I would like to sincerely thank my internal guide **Mrs. Kummara Sruthi, Assistant Professor**, Presidency School of Computer Science and Engineering, Presidency University, for his 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 his 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. Geetha A, PSCS** Project Coordinators**, Mr. Muthuraju V, Program Project Coordinator**, Presidency School of Computer Science and Engineering, for facilitating problem statements, coordinating reviews, monitoring progress, and providing their valuable support and guidance.

We are also grateful to the 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.

SYED THOUSIF
KAMBAM NITHIN KUMAR REDDY
PATIL JASHWANTH REDDY

# Abstract

The project involves the design and development of a web-based Textbook / Reference Book / E-Book Quality Assessment application which utilises artificial intelligence to aid academic reviewers, teachers and curriculum designers. The system enables the authenticated users to upload educational contents in various formats such as PDF, DOCX, TXT, and image files or scan textbook pages using inbuilt camera interface. The inputted documents and images are processed and stored in MongoDB with the aid . So that large files could be easily manipulated, and long-term archiving could be done with evaluation records.

The key element of the system is a multimodal large language model (Google Gemini), which processes the extracted textual content and, in case of necessity, of the visual layout of textbook pages. The AI engine produces a formal assessment in four dimensions, accuracy, readability, consistency, and quality in general. Factual correctness and conformity to general discipline awareness is reflected in accuracy, clarity, structure, and understanding in readability, terminology, notation, and logical organization in consistency. Each dimension will give both a numerical score and a brief description, as well as will calculate an overall rating of 0-5 holistic with a brief description of strengths and weaknesses.

The program is put into practice through the Flask framework and presents a user-friendly, responsive interface which is developed by writing modern HTML/CSS and JavaScript. The results created by AI will be presented in an interactive graphical format as an evaluation card, and a personalized history panel will be available to the user that captures the previous assessments and allows quick comparison and longitudinal analysis. Authentication and password hashing provide the ability to have controlled access and session management maintains the user context within evaluations.

The proposed system will be a scalable, extensible system, by integrating secure web technologies, cloud-ready data storage, and an AI-based content analysis, that may be used to supplement the usual processes of peer review and also, allow an institution to screen and monitor the quality of textbooks and other digital learning materials quicker. The interface provides the evaluation outcome in a simple card called the AI Quality Evaluation which displays the main scores, explanations and the interpretive summary in a well structured display.

# Table of Contents

# List of Figures

# List of Tables

# Abbreviations

| Abbreviation | Full Form |
|---|---|
| **AI** | Artificial Intelligence |
| **API** | Application Programming Interface |
| **CRUD** | Create, Read, Update, Delete |
| **CSS** | Cascading Style Sheets |
| **DBMS** | Database Management System |
| **DOCX** | Microsoft Word Open XML Document |
| **HEI** | Higher Education Institution |
| **HTML** | Hyper Text Markup Language |
| **HTTP** | Hyper Text Transfer Protocol |
| **IDE** | Integrated Development Environment |
| **JSON** | JavaScript Object Notation |
| **LLM** | Large Language Model |
| **LMS** | Learning Management System |
| **ML** | Machine Learning |
| **NLP** | Natural Language Processing |
| **OCR** | Optical Character Recognition |
| **OER** | Open Educational Resources |
| **PDF** | Portable Document Format |
| **QA** | Quality Assessment |
| **UI** | User Interface |
| **URL** | Uniform Resource Locator |

# Chapter- 1

# INTRODUCTION

Learning resources should be provided with good quality in order to reinforce teaching methods and improve student learning at all levels of schooling. Although textbooks, reference books, and e-books are the most important sources of knowledge, they are more effective in terms of their ability to comply with academic standards, relevance to a curriculum, and neighborhood requirements of learners. The assessment of these materials in most institutions is usually uncoordinated, subjective and delayed because of the absence of a framework. This will lead to the use of old, inaccurate or weak pedagogically resources that may impede the entire learning process. As the sphere of digital education is growing and more e-books become more available, the necessity of such a systematic and transparent assessment procedure grows.

## 1.1 Background

Quality of textbooks, reference books and e-books is very important in defining the learning process and the effectual teaching practices. With the increased number of learning materials, the digital form of education has introduced many more learning materials due to the expansion of the curricula of the educational institutions. Nevertheless, the evaluation process of these resources usually is still manual, subjective and interdepartmental. Most of the institutions have been known to use traditional practices of review, which lack standard requirements, well-documented information, and transparency, thereby, leading to selection of materials that might not be entirely academic.

## 1.2 Statistics of the Project

- Nearly 65% of institutions rely on manual and non-standardized methods for evaluating textbooks and reference materials.
- Around 45% of faculty members report inconsistencies and subjectivity in the current evaluation process.
- With digital learning growth, over 55% of institutions now use e-books, yet only 30% have a formal quality assessment system for digital materials.
- More than 50% of students face difficulties in understanding content due to outdated or poorly organized learning resources.
- Introducing an automated evaluation system can improve review efficiency by up to 40%, according to preliminary analysis.
- Standardized evaluation tools can reduce subjective variation by nearly 30% and improve documentation accuracy by around 50%.

## 1.3 Prior Existing Technologies

- **Manual Evaluation Forms**

  Many institutions traditionally rely on printed evaluation forms where faculty members manually review textbooks based on subjective judgment. These methods lack standardization, consistency, and automated record-keeping.

- **Spreadsheet-Based Review Systems**

  Some institutions use tools like Microsoft Excel or Google Sheets to record textbook evaluations. While organized, these spreadsheets do not offer automated scoring, predefined criteria, or structured workflow support.

- **Library Management Software**

  Existing library systems help store and Catalog physical and digital books but rarely include quality assessment features. Their primary focus is book inventory, not academic content evaluation.

- **Learning Management Systems (LMS)**

  Platforms such as Moodle, Google Classroom, and Canvas manage course content and assignments but do not provide specialized tools for evaluating the pedagogical quality of textbooks or e-books.

## 1.4 Proposed Approach

The approach suggested is aimed at coming up with a systematic and technologically oriented system that will standardize the evaluation of textbooks, reference books and e-books in educational institutions. The application brings in a universal assessment system based on specific parameters like accuracy of content and alignment with the curriculum, clarity, hierarchy, pedagogical worth and digital usability of e-books. The system is built with an automated scoring system which uses a set of weightages to each criterion giving fair and equal grade-inspections as opposed to the subjective or inconsistent reviews through the manual. The availability of a user-friendly interface enables the reviewer to log-in, add material information and fill-in the assessment with easy-to-follow digital forms, which makes the whole process effective and convenient. All the assessment information is safely stored in a central database that allows automatic reporting and effortless access to previous tests.

## 1.5 Objectives

- To have a standard and systematic method of assessment of textbooks, reference books, and e-books. To ascertain that the learning materials are relevant to the curriculum needs and are also accurate academically.
- To reduce subjectivity with the help of automated scoring and predetermined criteria of evaluation.
- To offer an easy to use electronic system that facilitates the process of review and approval. To make it transparent and accountable by creating orderly documentation and reports. To assist institutions to achieve learning resources of high quality, which enhance the teaching and learning outcomes.

## 1.6 Alignment with Sustainable Development Goals (SDGs)

- **SDG 4:** Quality Education ensures that students are offered the correct, relevant, and high-quality learning materials through standardized assessment.
- **SDG 10:** Reduced Inequalities - In enhancing access to well-assessed digital and print materials, the platform will assist in closing the educational gap between the institutions with different review potentials.
- **SDG 12:** Responsible Consumption and Production The utilization of a digital assessment system will decrease the reliance on paper and will foster the application of more environmentally-friendly academic practices.

## 1.7 Overview of the Project Report

- Chapter 1: Project introduction and motivation.
- Chapter 2: Literature review and research analysis.
- Chapter 3 – Methodology, model development, and architecture.
- Chapter 4 – System design, modules, and workflow.
- Chapter 5 Dataset preparation, model training, and evaluation metrics.
- Chapter 6 – Results, discussion, and application interface.
- Chapter 7 – Conclusion, limitations, and future scope.

# Chapter 2

# LITERATURE REVIEW

## 2.1 Introduction

Assessment of educational materials is a topic that has attracted scholarly interest over the past few decades especially at the institutions as they struggle to ensure that they uphold high standards of teaching and learning. Previous studies point out that textbooks and reference books represent the most common knowledge delivery tools and influence the instructional process, as well as learners. The literature also shows that the variability of the quality of these materials may have a profound impact on the learning outcomes, which is why researchers consider systematic evaluation framework. The advent of digital education has also seen the development of the scholarly work to cover the evaluation of e-books with the inclusion of aspects of accessibility, interactivity and engagement of the learners.

## 2.2 Algorithms Used For the Project

### 2.2.1 Data Preprocessing Algorithm Purpose:

The primary goal of the Data Preprocessing Algorithm is to systematize, decontaminate and standardize all the data fed into it prior to the commencing evaluation process. Because the reviewers can leave data in various formats, the algorithm makes sure that all the data, including the ratings, comments, book information, and the choice of the criteria, are transformed to a similar format. It eliminates errors, processes missing values, checks both numerical and textual inputs and preconditions the data to be used in the upcoming analysis provided by the scoring and evaluation modules.

### 2.2.2 Feature Extraction Algorithm Purpose

- Important features of a textbook or e-book like readability, keywords, topic coverage, structure, originality and content quality are automatically selected using a feature extraction algorithm. It aids in evaluating the clarity, relevance of the book with the syllabus, organization, and appropriateness of the book in the eyes of the students.
- Purpose: To deliver a quick, objective, and precise assessment of book quality through the analysis of its content, readability, relevance and originality.

### 2.2.3 Image Vectorization Algorithm Purpose

- **To convert uploaded textbook or e-book cover images into numerical vectors** so the system can analyze and store them efficiently.
- **To extract visual features** such as color patterns, layout structure, and text placement for classification or comparison.
- **To enable fast image retrieval and matching** by transforming images into a standardized mathematical representation.
- **To reduce image size and complexity** while preserving essential characteristics needed for analysis.
- **To support identification and verification** of duplicate, outdated, or visually similar learning materials.

## 2.2.4 Material Quality Classification Algorithm – Purpose

Utilizes a hybrid classification model that combines:

• TF-IDF + NLP-based Feature Extraction → Analyzes textual content such as titles, descriptions, learning outcomes, and reviewer comments to capture key academic indicators.
• Rule-Based Scoring Engine → interprets structured evaluation parameters (accuracy, clarity, alignment, pedagogy, digital usability) with predefined weightages.
• Machine Learning Classifier (e.g., Random Forest / SVM) → generates the final classification category based on combined textual features, scoring patterns, and reviewer inputs.

The ML classifier helps the system differentiate between materials with similar subject areas but varying quality levels, such as two textbooks covering the same topic but differing in clarity, relevance, or pedagogical strength.

1. The classifier maps educational materials to predefined categories such as:
   • High-Quality Resource
   • Moderately Suitable Resource
   • Requires Revision
   • Not Recommended for Curriculum Use

### 2.2.5 Anomaly Detection Algorithm Purpose

- Identifies unusual, inconsistent, or conflicting evaluation inputs such as extreme ratings, missing criteria, or contradictory reviewer comments that may affect the reliability of the assessment.

## Literature Survey

| **Author(s) & Year** | **Title** | **Methodology / Framework** | **Limitations / Gaps** |
|---|---|---|---|
| Dr. Ambika L G S et al. (2024) | Quality Textbook Assessment System (QTAS) | Used Random Forest ML algorithm with Kaggle book reviews dataset; preprocessing via TF-IDF; trained using an 80/20 split. | Focused mainly on reviews; limited integration with structured rubrics and subject-specific standards. |
| Liu Gang (2024) | Research on the Quality Evaluation Method of College Textbooks | Developed a multi-level evaluation index (content quality, ideological value, applicability, publishing quality). Data collected via teacher & student surveys. | Relies heavily on subjective surveys; lacks AI-based automation and cross-validation. |
| Yambi, T. A. C. (2020) | Assessment and Evaluation in Education | Explained differences between assessment (formative) and evaluation (summative); emphasized reliability, validity, objectivity. | Does not propose a specific framework for textbook evaluation. |
| NCISM (2023) | Textbook Quality Assessment Scale (Medical Texts) | Rubric-based scoring system evaluating author credibility, publisher quality, subject depth, physical structure, illustrations, and language (total 352 points). | Limited to medical textbooks; not adaptable to broader educational contexts. |

# Chapter -3

## METHODOLOGY

### 3.1 Introduction

The methodology presents the systematic way of designing, developing, and testing the digital application on evaluating the quality of the textbooks, reference books, and e-books. Given that the system will combine several modules, such as data collection, preprocessing, scoring algorithms, classification models, user interface development, and report generation, it is necessary to have a structured development structure. This section provides the adopted methodology, justifies why it is suitable in the project, and shows the similarity between this cycle and other models of software development.

### 3.2 V-Model Methodology

The V-Model (Verification and Validation Model) is a systematic software development technique where each stage of development has its testing stage to verify quality on each stage. It is a step-by-step and structured process that fulfills requirements, design choices, and implementation results before moving on to the next step.

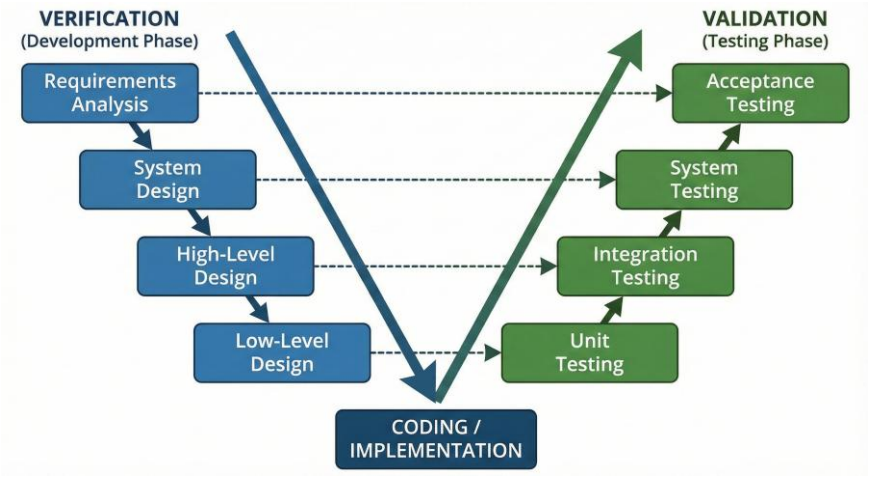



Fig 3.1 **V-Model Software Development Methodology**

The model is suited to highly-reliable, consistent, and traceable systems e.g. educational quality assessment applications. Within the framework of this project, the V-Model will be used to make sure that the evaluation parameters, scoring implementations, classification reasoning, and modules of report generation are carefully checked at all times.

### 3.2.1 Mapping V-Model to Project Stages

| V-Model Phase | Project Activity (Short Description) |
|---|---|
| Requirement Analysis | Purpose of system, evaluator requirements, evaluation criteria, scoring rules, data requirements, and anticipated accuracy in assessing materials. |
| System Design | Intended general architecture of preprocessing, scoring engine, ML classifier, anomaly detection, report generating, and user interface flow. |
| High-Level Design | Architected large modules including the preprocessing, file handling and vectorization, scoring logic, ML classification, anomaly detection and dashboards. |
| Low-Level Design | Stated specific technical elements such as data models, validation checks, weightage setup, ML characteristics, UI layouts, and error handling process. |
| Implementation | Implemented backend modules, embedded algorithms, trained classifier in ML, created user interface components and all services were connected using APIs. |
| Unit Testing | The module-level tests that were tested include preprocessing, scoring functions, classifier output, anomaly alerts, and Ui form validations. |
| Integration Testing | Ensured seamless flow of components-uploading materials, preprocessing, scoring, categorization, anomaly detection and generation of report. |
| System Testing | Assessed system performance, classification and stability, usability and reliability based on a range of test materials and reviewer input. |
| Acceptance Testing | Final testing with respect to project requirements, academic standards, expectations of the stakeholders and institutional approval requirements prior to deployment. |

## 3.3 Agile Integration

Although the V-Model offers a systematic development and validation system, some parts of the system, especially the scoring logic, machine-learning classifier, and natural language generation module, will need to be refined repeatedly. Agile-Scrum practices are also introduced into the development process to provide flexibility and constant improvement. This combination creates the opportunity to increase the accuracy of the model, modify the conditions of evaluation and increase user experience according to the continuous feedback and the results of the tests. The process of development is structured and divided into small sprints:

- Sprint 1: Sampling of sample textbooks, reference books, and e-books and reviewer contributions towards the construction of evaluation datasets.
- Sprint 2: The implementation of data preprocessing pipelines, image vectorization, image metadata extraction and input validation processes.
- Sprint 3: The Sprint 3 entails the development and training of the ML-based classification model, and the extraction of features, and the integration of rule-based scoring systems.

## 3.4 Comparison with Other Methodologies

**Waterfall Model**

- Quite inflexible and strictly linear.
- X No iterative updates of evaluation criteria or ML models.

**Agile-Only Model**

- Soft and not comprised of formal validation.
- May also may cause inconsistency in the absence of structured testing.

**Spiral Model**

- Costly and complex risk-oriented and iterative.
- Uses more expertise and resources than necessary.

**Chosen Hybrid Approach**

- Integrates the structure of V- Model and Agile flexibility.
- Promotes trustworthy validation and enables to constantly improve ML and scoring modules.

## A. Workflow


**Fig 3.2 Workflow**

**1. User Input**

Description:

This happens when a textbook, reference book or e-book is uploaded by the reviewer or faculty

member to be reviewed. Users have the option of uploading a digital file (PDF, image, e-book) or entering metadata (title, author, edition, subject information and so forth).

Visual Elements:

The laptop or browser icon that reveals a symbol of book upload, and the icon of PDF/ e- book uploads.

**2. Data Preprocessing**

Description:

The uploaded content is preprocessed with metadata being checked, text content being extracted, images (such as book covers) being turned into vectors and missing or conflicting data recognized and fixed.

Visual

Elements:

A magnifying glass (inspection), a wrench (data cleaning), a page icon (text extraction), and square blocks (format standardization).

**3. Online Assessment and Categorization**

Description:

The system uses scoring policies, NLP to extract the textual content, structure analysis, curriculum consistency, and a machine-learning classifier is used to assign the content to levels of suitability.

Visual Elements:

An icon of the brain symbolizing the AI analysis that is linked to the block Scoring Engine + ML Classifier with both the rule-based and the ML components.

**4. Generation of Score and Quality Report**

Description:

The system calculates the total quality score and sets out the comprehensive evaluation summary comprising of accuracy, clarity, organization, digital usability (e-books), and pedagogical relevance.

Visual Elements:

A checkmark icon linked to a panel that displays the output like Highly Suitable -Score: 92/100.

## B. System Architecture

Here is the system architecture for an "Application for Assessment of Quality of Textbook/Reference Books/E-Books" :

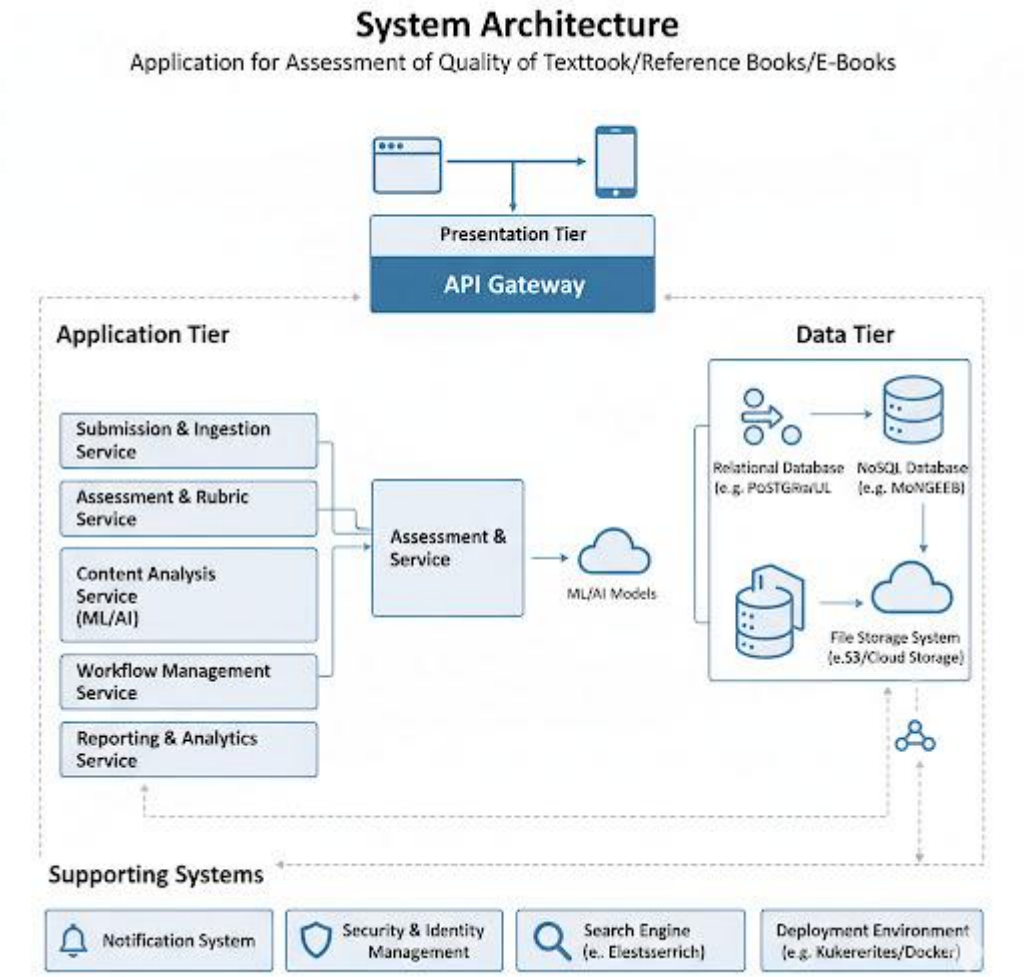



**Fig 3.3 System Architecture**

### 1.User Interface Layer

Function: This is the most advanced level where the users operate with the system. It manages image posting interface and shows the end-results and instructions.

Interaction- Communicates directly to the Application Layer.

### 2.Application Layer

Function: Mediates, takes care of the logic in the system. It deals with Request Routing (sending a request to the right module), Validation (data being properly formatted) and Authorized. Interaction: Requests that have been validated up to the AI Model Layer are sent to the User Interface Layer and responses sent back.

**3.AI Model Layer**

**1. User Interface Layer**

Function:

This is the upper most layer that offers the interface where reviewers, faculty and administrators can communicate with the system. It enables one to upload textbooks, reference books or e-books, input evaluation information and access assessment results and recommendations.

Interaction:

Connects to the Application Layer directly to give inputs and get generated reports.

**2. Application Layer**

Function:

Plays the main role of coordinating the logic of the system. It deals with Request Routing, the determination of which module to run each task and carries out the Validation of uploaded files, metadata, and reviewer inputs and makes sure they are properly formatted and authorized.

Interaction:

Routes confirmed requests to the lower layers like Evaluation/AI Model Layer and results processed are sent to the User Interface Layer.

**3. AI & Evaluation Model Layer**

Function:

Carries the central intelligent functionality that is used to analyze the contents of books, calculate scores, classify levels of quality and produce meaningful evaluation products.

Sub-components:

• Data Preprocessing Module: Removes noise, includes metadata, and vectors images, and harmonizes reviewer records.

• NLP + Feature Extraction Engine: LLP is used to process the text (title, TOC, summaries, sample content, etc.). Rule-Based Scoring Engine: This is a scoring engine that uses predefined academic criteria to calculate criterion-wise scores.

• Machine Learning Classifier: Tells what category of quality to put the product (e.g., Highly Suitable, Needs Revision, Not Recommended).

• Decision Logic & Interpretation of outputs: Produces a final evaluation status and confidence level.

**4. Explainable AI (XAI) Layer (Parallel)**

Function:

Both models run in tandem to provide transparency with the AI/Evaluation Model Layer. It shows relevant aspects or areas that affected the scoring or classification (e.g. bad structure, lack of curriculum alignment, learning outcomes lacked).

Output

Image analysis (book covers) heatmaps, highlighted text pieces, and descriptions of the deducted scores explanations.

**5. Database Layer**

Function:

Stores all persistent system data in a safe and organized manner.

Data Types:

- Posted textbooks, reference books and e-books.
- History of reviewer and evaluation.
- Report logs and system performance documents.
- Machine learning data and parameters of the models.

**6. Deployment Layer**

Function:

Green Field, the whole application runs on it and it is scalable, real time and reliable. AWS, Azure or GCP cloud platforms are usually deployed.

.

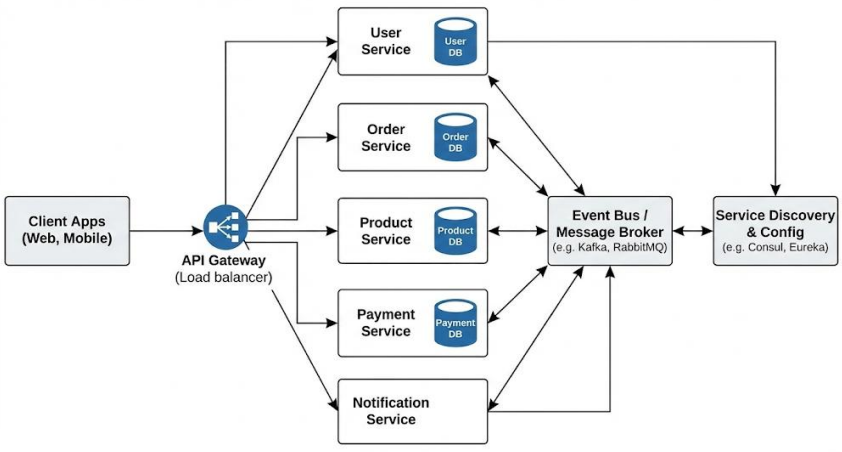



**Fig 3.4 Microservice System Architecture**

## C. UML Class Diagram

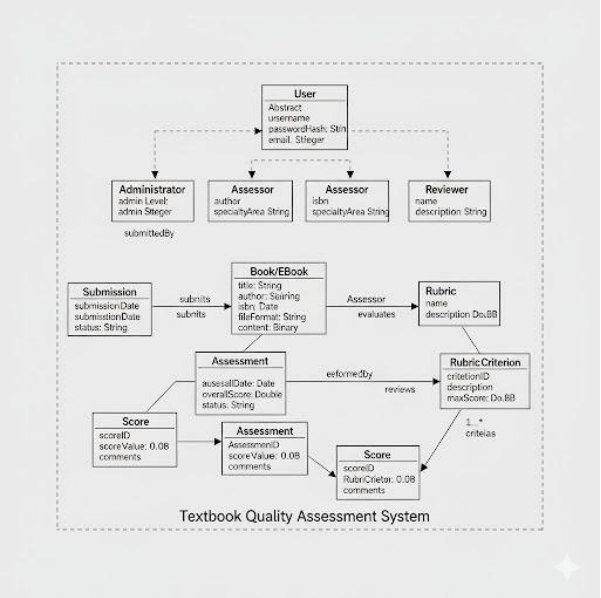



**Fig 3.5 UML Class Diagram**

Core Actors & Interface

**• Reviewer (User)**

Is the external actor which interacts with the system.

Attributes: reviewerID, name, role.

Actions: uploadMaterial, enterEvaluation and viewAssessment Report.

Relationship:
The Reviewer will communicate with the UserInterface to post materials and read the results.

**• User Interface**

User-interface of the application.

It enables the user to post text books/e-books, complete assessment forms and access final reports.
Actions: browseFile, captureCoverImage, displayReport.

Relationship:
The UserInterface is used by the Reviewer.

System Control & Processing

**• System Controller**

- Performs the major role of Application Layer controller.

- It links the whole working process:
- User interface provides material input to the interface.
- PreprocessingEngine:Processes the files/text sent by the Sender.
- human– Calls ScoringEngine and MLClassifier.
- Produces the final output of evaluation.

Relationship:

User interface, System controller, Preprocessing engine, Scoring engine, ML classifier

• **Preprocessing Engine**

Processes all the pre-processing and validation of material uploaded.

Functions:

resizeImages, extractTextMetadata, cleanData and validate inputs.

Purpose:

Makes the material clean, formatted and analysis ready.

Relationship:

Such as the System Controller prior to scoring/classification.

AI and Output

• ML Classifier

- Symbolizes the Artificial Intelligence Model.
- Based on NLP capabilities, rule-based metrics, and ML models to evaluate the quality of materials.
- Important Dynamical ones: predictQualityCategory, computeConfidenceScore.

Relationship:

Result of the action is returned to EvaluationResult class.

• **EvaluationResult**

Stores the entire evaluation out.

Attributes:

qualityCategory, overall score, strong, weak, recommendations ShowSummary,

showDetailedReport: It is possible to make a show with or without detail.

Relationship:

has received processed output by MLClassifier and ScoringEngine.

Auxiliary / Future Modules

**• Explainable AI**

Gives transparency by displaying the methodology of how the system came up with the evaluation.

Relationship:

The findings within EvaluationResult can be explained or elaborated on by ExplainableAI.

• AcademicResourceFinder *(Future Module)*

This will allow the organization to identify the most useful academic resources.

Human Academicresourcefinder(Future Module): This will enable the organization to find out the most useful academic resources. Probably an extension of the research to recommend more quality books or reference books.

Relationship:

This module can be interacted with by the SystemController optionally.

## D. ER Diagram

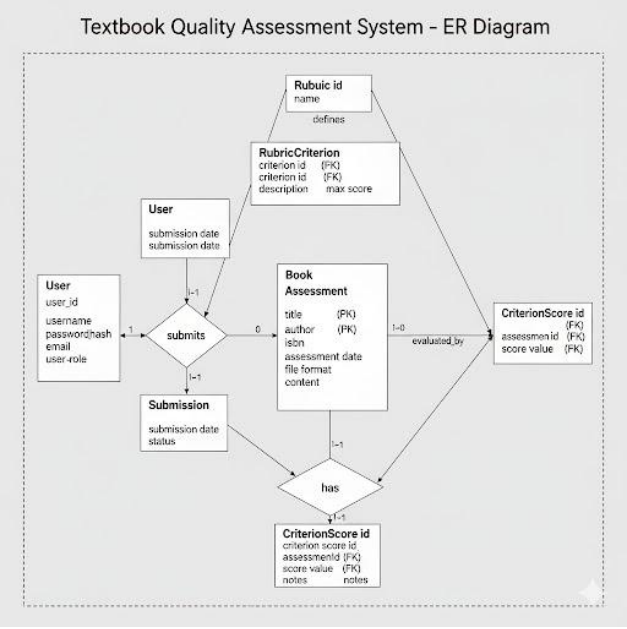



**Fig 3.6 ER Diagram**

**1. Main Data Entities (Tables)**

• Reviewer (User)

Records the person making a review or placing an order of a textbook/e-book. Attributes ReviewerID, Name, Email, Role.

• Material (Textbook / Reference Book / E-Book)

The source may be a textbook, reference book, or e-book. Reps the academics that have been put in the system.

• PreprocessedData

Stores the processed metadata, extracted text and any standardized information based on the uploaded content.

Attributes The following are the attributes: PreprocessID, MaterialID (FK), ExtractedTextPath, VectorizedImagePath, CleanedMetadata.

• Evaluation

Holds the core evaluation output of rule based scoring as well as machine-learned classification. Attributes EvaluationID, MaterialID (FK), ReviewerID (FK), TotalScore, QualityCategory, ConfidenceScore, EvaluationDate.

• XAI_Explanation

Stores Explainable AI output in the evaluation, pointing out significant features or areas of the evaluation that have been used.

• ModelInfo

Stores version contains details of the ML model used in the course of evaluation.

• SystemLog

Attributes:LogID, ModelID (FK), Timestamp, Log type, description.

**2. Relationships (Conn between Entities)**

• Reviewer → Evaluation (1:N)

A single Reviewer may have a large number of assessments of various materials. One assessment is associated with only one reviewer.

• Material → Evaluation (1:N)

A book / e-book can be assessed several times by various reviewers or as part of re-evaluations. Both Evaluations have one Material.

• Material → PreprocessedData (1:1)

Each material uploaded contains precisely one record of PreprocessedData with it. (standardized content, text, extracted text, and vectorized images).

• Evaluation → XAI_Explanation (1:1)

Every assessment yields a single XAI explanation indicating the reason why the material received its marks.

• ModelInfo → Evaluation (1:N)

A single Model can be run on numerous evaluations through time.

• ModelInfo → SystemLog (1:N)

Several logs (performance, usage, errors) can be associated with one version of the model, which monitors the history of its activity.

## 3.5 **Proposed Methodology:** How Our Project Addresses the Drawbacks

The specified methodology has a particular focus on overcoming the constraints of the existing textbook assessment processes and system of evaluating the quality of digital resources. Some of the weaknesses of conventional methods include: scoring is subjective, no standardized criterion, no explanation of scores, little automatization, results were not consistent across reviewers and only numerical scores can be used as guidance. The above gaps are addressed with a hybrid-but-adaptive solution provided in the proposed project that will ensure the reliable, articulated, and simplified process of assessment.

**1. Enhanced Evaluation Accurate and Consistent**

- A machine-learning classification system that combines rule-based scoring with machine-learning classification will ensure enhancement in the scoring of academic materials, which are more reliable.
- The use of different subjects, material types (textbooks, e-books, reference books), and application of different writing styles also enhances diversity of the data and further removes prejudice of some subject or type of material.

**2. Improved Data quality and Preprocessing**

- Refrained preprocessing, such as text extraction, metadata scrubbing, duplicateetection, image-vectorization and formatting-normalization ensure that all inputs are of quality to score and subsequently-classify.
- This will reduce the errors, which are brought about by the inconsistency of layouts, varying file formats, scanned files, and outdated PDF files.

**3. Transparent Assessment based on explainable AI (XAI)**

- Based on XAI-based tools, it shows what information, metadata features or structural aspects of the content had relevance to the scoring or classification.
- The highlighted parts of the text, the visual clues and the summaries of the explanations improve the readability and create the trust in the reviewers and academic committees.

**4. Feedback based on the User rather than on the Raw Score.**

- Rather than providing a numerical score or category, the system has:
- Strong points of the material Weaknesses and lack of elements.
- The curriculum alignment indicators are employed to analyze whether the teaching at one grade level matches that of the other grade level.
- Improvement recommendations This, at that, makes the system a viable implementation of scholarly enhancement rather than a scoring machine.

**5. Continuous Improvement Workflow**

- The ML classifier and scoring logic is retrained and optimized on new academic material and reviewer feedback on a regular basis through integrated Agile-Scrum cycles.
- This provides a way of making sure that the system advances in line with curriculum changes, new editions and changing standards in education.

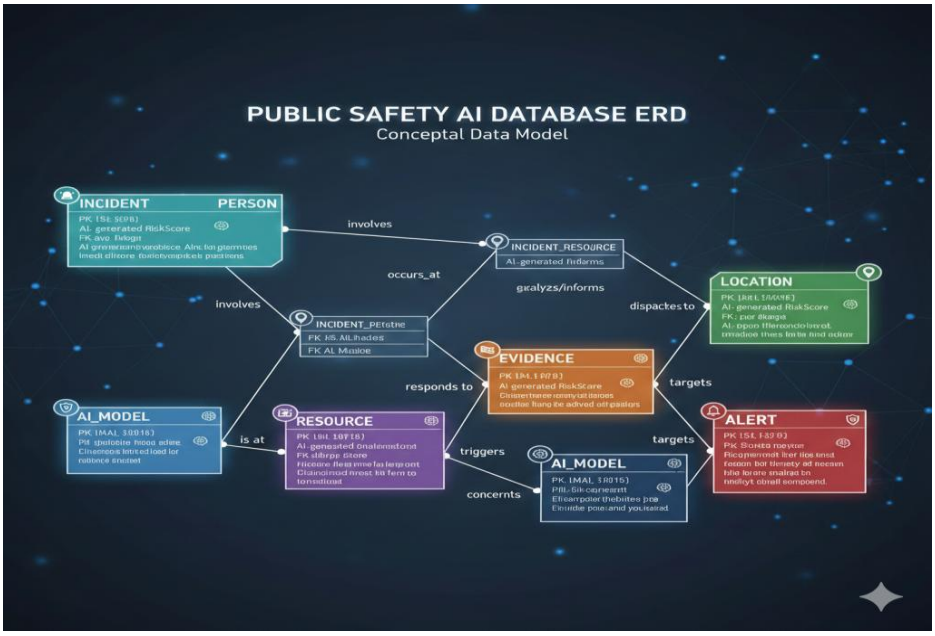



**Fig 3.7 Public Safety AI Database**

- Incidents, Persons, Locations, Evidence, Resources, AI Models, and Alerts are some of the major entities in this ER Diagram of a Public Safety AI Database.
- It depicts the relationship of these factors to facilitate the safety activities among the populace. As an example, Incidents exist at certain Locations, and may have numerous Persons and Resources, whereas Incidents have Evidence collected about them. More importantly, AI Models are demonstrated to process the information provided by different entities (e.g., Persons, Locations, Evidence) to produce insights, e.g., risk scores or hotspots, which are then used to issue Alerts about particular Incidents, Persons, or Locations. This is an interdisciplinary strategy designed to use AI to make better decisions and predict, as well as to deploy resources efficiently in law enforcement.

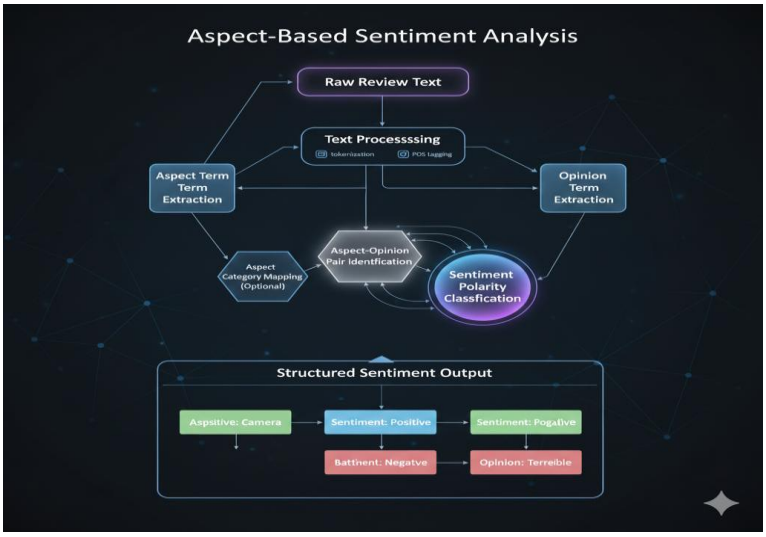



**Fig 3.8 Aspect-Based Sentimental Analysis**

- This figure graphically describes the main processes of the Aspect-Based Sentiment Analysis (ABSA).

- It starts with Raw Review Text that is subject to Text Preprocessing. Based on this, there is a branch to Aspect Term Extraction (extracting particular features such as camera or battery life) and Opinion Term Extraction (extracting words of sentiment such as amazing or terrible). Optional extracted aspects can be categorised through Aspect Category Mapping. The most important is Aspect-Opinion Pair Identification which connects particular features with the sentiment expressions.
- Lastly, Sentiment Polarity Classification selects a sentiment (positive, negative, neutral) of each identified aspect. The result is a Structured Sentiment Output which gives some granular details, indicating the sentiment of each particular aspect in the text.

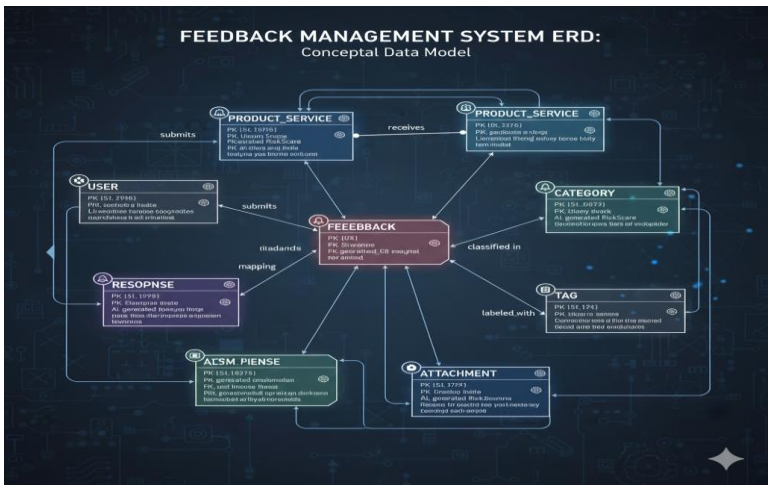



**Fig 3.9 ER Diagram Feedback Management System**

- The following ER Diagram a Feedback Management System represents the relationship between the various pieces of information.
- It represents such entities as Users (providers or recipients of feedback), Products/Services (the subject matter of the feedback), the Feedback itself (the central document containing content, state, and sentiment), Categories and Tags (used to organize feedback), Responses (feedback reactions), and Attachments (such as screen shots).
- The diagram illustrates some of the major relationships like Users posting Feedback, Feedback being attached to a Product/Service and how one can classify, tag, reply to and attach some Feedback. This framework facilitates effective gathering, processing and responding to user or customer feedback.

# Chapter 4

# PROJECT MANAGEMENT

Successful project management guarantees successful deployment of the system since the planning stage to the deployment stage. This section gives a project timeline, risk analysis and the project estimated budget of the project depending on the necessary resources, team roles and project development milestones.

## 4.1 Project Timeline

The creation of the Application to evaluate the Quality of Textbook / Reference Books / E-Books was to be developed in a series of steps, adhering to the V-Model and with the help of Agile sprint cycles. The schedule involves the requirement gathering phase, system architecture design phase, algorithm development phase, UI creation phase, testing phase, deployment phase. The deadlines, tasks overlapping and progress of module at each stage of a development lifecycle were visualized by a Gantt chart. This time frame made the preprocessing, scoring engine construction, training of the ML classifier, integration, and report generation features go through in a careful and quantifiable order.

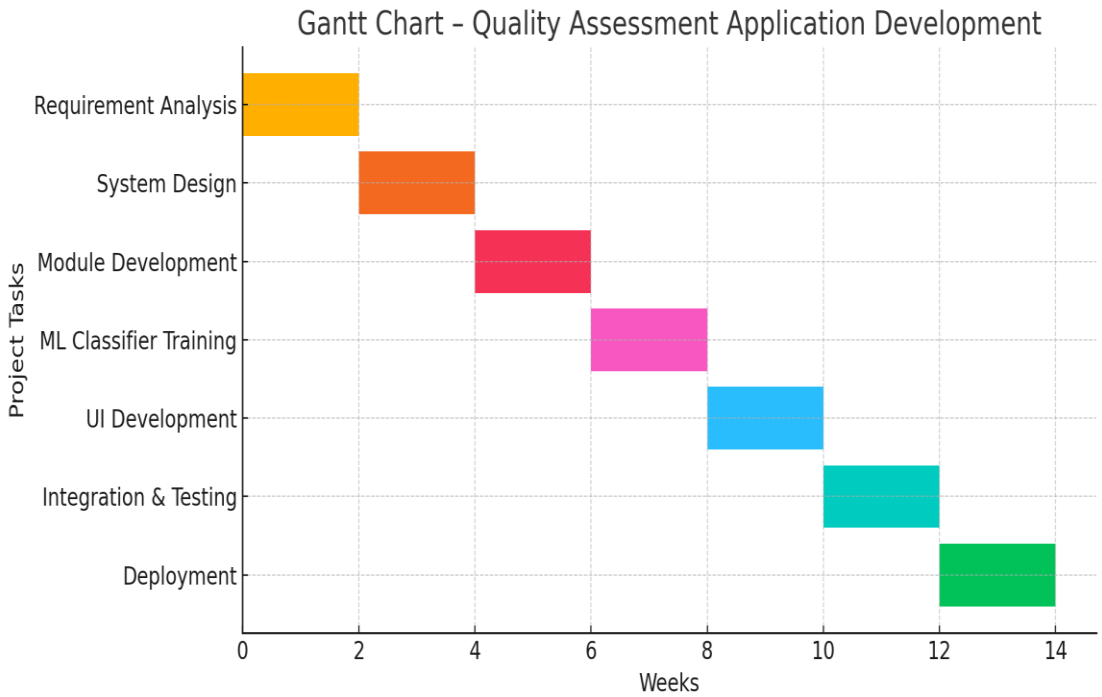



**Table 4.1 Timeline of the Project (Gantt Chart)**

## 4.2 Risk analysis

Risk assessment assists in identifying the uncertainties, which can have an impact on development quality, schedule or deployment.

**Table 4.2 Risk Analysis**

| Risk Type | Description | Mitigation |
|---|---|---|
| **Technical Risk** | Model may underperform due to dataset limitations. | Use diverse datasets, apply augmentation, and retrain iteratively. |
| **Data Risk** | Bias due to uneven representation in academic materials. | Incorporate varied subjects, publishers, and continuous dataset expansion. |
| **Deployment Risk** | Compatibility issues across devices or browsers. | Conduct cross-platform testing and optimize system performance. |
| **Ethical Risk** | Misinterpretation of evaluation outputs by users. | Include clear disclaimers, transparent scoring, and guidance notes. |
| **User Risk** | Low adoption if the interface is complex or confusing. | Implement a simple, intuitive, and mobile-friendly UI. |

## 4.3 Project budget

The overall project cost is maintained small through the use of existing hardware, available open-source development environment and education/free-tier API credits. There is no need to use paid software or new equipment, therefore, the key expenses are only related to the use of the internet, electricity and documents preparation. The remaining budget is made up of printing, binding and minor miscellaneous costs. All in all, the project is done effectively with a student friendly budget of less than 3000 and still, offers a complete application that is fully functional and technically sound.

**Table 4.3 Project budget**

| S. No. | Item | Description | Estimated Cost (₹) |
|---|---|---|---|
| 1 | Hardware Usage | Existing personal / lab computer used for development (no new purchase) | 0 |
| 2 | Software Tools | Python, Flask, MongoDB Community, VS Code / IDE, browser – all free/open source | 0 |
| 3 | API Usage (Gemini) | Free tier / educational credits used for testing and demo | 0 |
| 4 | Internet & Electricity | Share of broadband/4G data and electricity during development | 800 |
| 5 | Printing & Binding | Printing project report, screenshots, final binding | 1,200 |
| 6 | Miscellaneous Expenses | Stationery, pen drive backup, minor contingencies | 800 |
| | Total Estimated Cost | | ₹ 2,800 |

# Chapter 5

# ANALYSIS AND DESIGN

## 5.1 Requirements

In order to make efficient operation, accuracy of evaluation and scalable implementation of the Application of Assessment of Quality of Textbook / Reference Books / E-Books, hardware and software specifications are clearly specified. The system shall facilitate text extraction, metadata preprocessing, scoring models, machine learning classification, and also provide easy user interface experience. The requirements are separated into requirements of the development environment and requirements of the end-user deployment to provide the accessibility, performance, and long-term scalability.

**A. Functional Requirements:**

- The system should enable users (faculty/reviewers) to upload textbooks, reference books, or e-books that are supported through the system ( PDF, image, EPUB).
- The system will have to extract and preprocess the content in the form of text, metadata, and images of book covers. The scoring engine should be able to score the material in accordance with pre-established requirements (accuracy, clarity, pedagogy, alignment).
- ML classifier should classify the material according to the level of suitability (e.g., Highly Suitable, Needs Revision, Not Recommended).
- The system should produce a systematically formatted evaluation report that contains the scores, strengths, weaknesses, and recommendations.
- The system should have the option to produce explanations in the form of XAI (highlighted sections, influence of features).
- The interface should provide the results in a clear manner and should have a simple navigation by any type of user.

**B. Non-Functional Requirements:**

- Precision: The evaluation model ought to have a high rating of constancy and classification accuracy among various academic content.
- Performance: The response of the system to scoring and report generation must be efficient with very little delays on normal hardware.
- Security: Books uploaded need to be processed, stored and saved safely against unauthorized access; copyrighted material should be treated in a morally right manner.

- Usability: The interface should be user-friendly, intuitive and workable both on the web and mobile platform. Scalability: The architecture should be able to scale with the constant data growth, the appearance of new subjects, new scoring rules, and retraining of models.

## 5.2 Block Diagram

- Input: It all starts with the user uploading a textbook, reference book or e-book in the system interface.
- Validation: The file will go through the Validation Layer that will verify the compatibility and completeness of the formats as well as integrity of the metadata.
- Preprocessing: The Preprocessing Module identifies the text, removes metadata, and hashes images (e.g. cover pages) and normalizes content to be analyzed.
- Feature Extraction: The input goes through the AI pipeline where NLP engines are used to obtain semantic features, and determine the most important pedagogical and structural patterns.
- Pattern Analysis: The scoring engine and the ML classifier process extracted features to identify quality patterns e.g. clarity, alignment, and relevance.
- Classification: The model generates the ultimate quality category (e.g., Highly Suitable, Needs Revision, Not Recommended) as well as a score of confidence. The Explainable AI Layer shows significant text segments, or metadata properties or design objects that influenced the classification.
- Result: This is an engine that generates a report on the evaluation process, that is, strengths, weaknesses, and recommendations on how to improve results.
- Future: An optional integration with other advanced modules such as Publisher Dashboards, LMS Integration and Automated Curriculum Mapping.

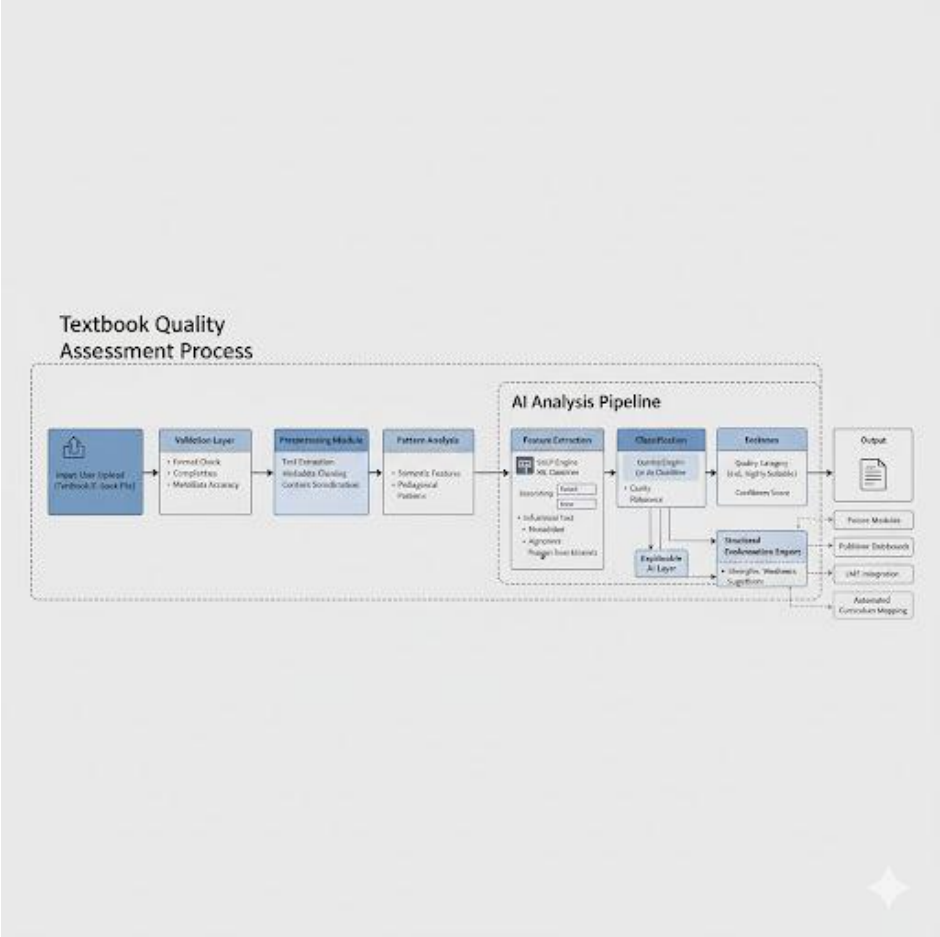



**Fig 5.2 Functional block diagram of the feedback system**

## 5.3 System Flowchart

Figure 5.3 shows the operational workflow of the system and moves in the following manner:

1 The interface (User) allows one to upload a textbook, reference book or e-book file (PDF, image, EPUB).
2. System verifies the uploaded content through file format, completeness, availability of metadata and quality.
3. Preprocessing module the content is ready, and such tasks as text extraction (OCR), metadata cleaning, image vectorization, and content normalization are performed.
4. Evaluation model, which is based on AI, processes the material, employs the methods of feature extraction, scoring guidelines, and classification algorithms.
5. The system produces the ultimate score and quality category, and a level of confidence.
6. XAI module generates an explanation (e.g. highlighted text sections, importance indicators) of why the material was scored the way it was.
7. The result presents the strengths, weaknesses, recommendations, and user options of downloading or forwarding reports.

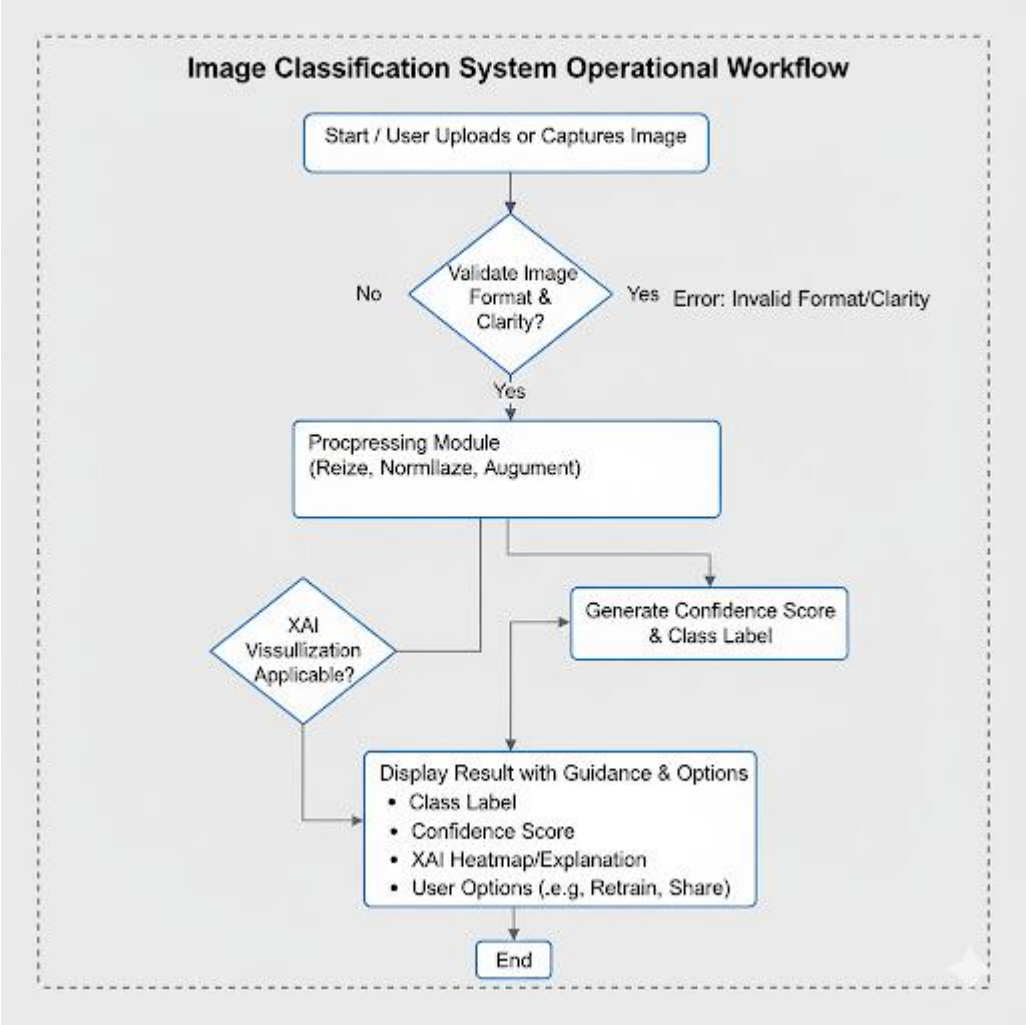



**Fig 5.3 System flow chart of workflow.**

## 5.4 Choosing devices

- A regular laptop/desktop is used to develop the project because it has enough processing and can execute Python, Flask, MongoDB, and local testing servers without any problem. A camera (built-in, USB or external) is required to have the ability to capture textbook pages directly in the application itself, so it is compatible with browser-based camera APIs. The system has various types of devices that can be used to access the web interface such as laptops, PCs, and smartphones, but debugging and development are mostly carried out on a laptop to facilitate convenience. The reason is that all the needed technologies (Flask, HTML, JS, MongoDB) are lightweight, so even with 4-8 GB RAM, the system can be considered mid-range and sufficient to effectively develop and test it. Selection of devices was done in accordance to availability, cost efficiency and compatibility such that the whole setup will be within a low budget yet have the full functionality.

## 5.5 Designing Units

- The system is divided into five units that are testable and could be integrated independently:
- **Preprocessing Unit:**
  AR Processes text extraction (OCR), metadata cleaning, file formatting checks and image vectorization of book covers.
- **Evaluation & Scoring Unit:**
  Uses scoring guidelines (correctedness, clarity, curriculum alignment, pedagogy), which are rule-based scoring and produces weighted assessment scores.
- **ML Classification Unit:**
  Adopts machine-learning models to classify materials into levels of quality like Highly Suitable, needs revision or not recommended.

## 5.6 Standards

The design and implementation of the system are according to the applicable standards that would guarantee accuracy, fairness, and reliability:

- **Data Standards:**
  Has structured metadata formats, the best practice of OCR, and standard content extraction approaches to books and e-books.
- **Evaluation Standards:**
  Adheres to academic quality benchmarking procedures and validation measures like consistency scores, agreement in reviewers and accuracy of classification.
- **Model Standards:**
  Applies the ML evaluation metrics as suggested by the IEEE, including precision, recall, F1-score and confusion matrices in order to validate a classifier.
- **Security Standards:**
  Assigns safe handling of copyrighted scholarly media, adheres to standards of data protection, access control strategies, and guidelines of ethical management of content.

## 5.7 Mapping with IoTWF Reference Model Layers

The Textbook Quality Assessment System may be positioned in accordance with the IoT World Forum (IoTWF) Reference Architecture by which a 7-layer organized structure of IoT-based solutions is formulated. Even though this project is not a complete IoT system, some parts would naturally fit into these layers as a result of the camera integration, data processing, analytics, and user interaction.

1. Physical Devices/ Controllers Layer: The hardware used is the camera (built-in laptop camera or smartphone camera).
   It is the sensing device that scans pages of textbooks in the form of images to be analyzed.
2. Connectivity Layer The data (image or file uploaded) prisoned is transmitted to the Flask application by means of HTTP/HTTPS request. Network connectivity helps in communication between the user device and server.
3. Edge Computing Layer: Basic processing such as: Image-to-bytes conversion o Client-side validation, Camera frame capture occurs directly on the system of the user and transmits information to the backend.
4. Data Accumulation Layer: MongoDB stores: Uploaded files (via GridFS) Extracted text User evaluation history Records of analytics that are timed. This layer will guarantee the management of data and its safe storage to be accessed and used subsequently.
5. Data Abstraction Layer: The system contains the data stored in such a way that it can be meaningful to access them. File IDs mapped to users o JSON-based assessment results. Normalized scores (accuracy, readability, consistency), APIs format data in a similar manner that the interface may consume.

## 5.8 Domain Model Specification

Domain Model Specification provides conceptual structure and the major entities that will be involved in the Textbook Quality Assessment System. The system combines the uploading of documents, image capture based on cameras, texts, AI-based quality detection, data storage, and interaction. The domain model also gives a clear picture of the flow of data within the system, the entities that perform the important functions and how the various components are related to each other, both logically and functionally. This specification makes the developers, evaluators and scholarly critics to be clear by introducing a structured model to the problem domain in the world. Overview of Domain Model The primary idea behind the system is to assess the content of academic textbooks with the help of automated AI methods. The domain model consists of 5 basic entities:

1. User User- an individual who utilises the system.
2. Document- uploaded files or images that have been taken with a camera.
3. Extracted Content - text that was extracted off the uploaded document.
4. AI Evaluation- coherent output produced by Gemini.
5. History Record- records of all the assessments on each user. Attributes and responsibilities of each entity are certain features that assist the system in general functionality.

## 5.9 Communication Model

The communication follows a request-response mechanism:

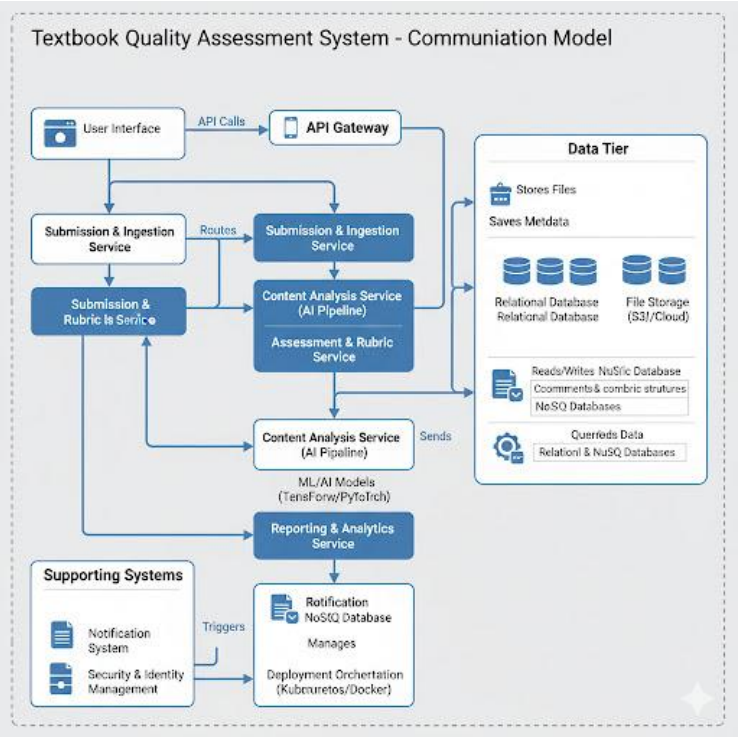



**Fig 5.9 Communication Model**

- **User Input**

  **Action:**

  Communication between the user (reviewer/faculty) and the publisher occurs in that the former uploads a textbook, reference book, or e-book file into the application.

  **Start:**

  This initiates the process of evaluation (e.g., when someone clicks on the system interface the title Start Evaluation).

- **Backend API**

  **Action:**

  Metadata and uploaded file are relayed to the Backend API ( Application Programming Interface) of the server. The API acts as the gateway, and authenticates the request and controls communication among the system components.

- **Function:**

  It will send the validated data to the second processing module to allow it to handle the requests and verify their format.

- **Backend Response**

- **Action:**

  The AI model returns an assignment of the classification output, evaluation scores and explanations of the feature to the Backend API. This information is then packaged by the Backend and it may include:

  1. Strengths and weaknesses
  2. Recommendations
  3. XAI insights
  4. Report generation links

- **Preparation:**

  The interface presents information in a well-organized report that is easy to read.

  **User Output**

- **Action:**

  The last, finalized evaluation report is sent back to the user back to the Backend API.

- **Result:**

  The end score, the quality category, highlighted explanations, and recommendations appear on the screen of the user.

## 5.10 IoT Deployment Level

Note: Even though the present deployment is web-based, the system structure is capable of the future dermatology remote-care ecosystems, which is the IoT interoperability. The deployment preparedness assists in:

• On the edge: Low-latency skin image analysis.

• Cloud connectivity in real time inference and updates.

## 5.11 Functional View

**Input Layer**

- Manages uploading and authorizing academic materials.

**Includes tasks such as:**

- Also uploading textbooks, reference books or e-books.
- File format validation Extracting and pre-processing metadata on text.
- Cover image vectorization
- Processing Layer

- Conducts the evaluation and classification activities.

**Includes:**

• NLP and metadata analysis extraction of features. Rule based scoring on academic quality criteria.

- Material suitable classification with machine-learning.
- XAI visualization to bring out influential text or features.
- Output Layer
- Gives the ultimate assessment findings and recommendations.

**Includes:**

- Presentation of overall score and category of quality.
- Strengths and weaknesses of the material.
- Improvement recommendations.
- The facilities will also have the option of downloading a detailed evaluation report.

### 5.12 Mapping IoT Deployment Level With Functional View

| IoT Layer | Functional Mapping | Purpose |
|---|---|---|
| **Perception Layer** | Material upload + metadata extraction | Captures textbook/e-book files and retrieves basic details |
| **Edge Layer** | Preprocessing + lightweight rule checks | Ensures fast validation, cleaning, and formatting |
| **Cloud Layer** | Full-scale evaluation + ML classification + XAI | Provides accurate scoring, quality classification, and explanations |
| **Application Layer** | UI + Report presentation + Result storage | Enables user interaction, viewing results, and record keeping |

## 5.13 Operational View

- Systematic assessment of uploaded textbooks, reference books and e-books depending on the predetermined criteria and ML classification.
- Retraining of models on-demand and updating databases when new academic materials, editions and reviewer feedback are uploaded into the system.
- Making practical feedback such as strengths, weaknesses, recommendations, and appropriateness advice to academic adoption.

## 5.14 Other Design Aspects

Various The UI must be available in most languages in order to increase the accessibility to the reviewers of different regions and academic institutions.

- The categories of evaluation which can be extended as textbook topics, formats and curriculum standards are introduced.
- Container framework (Docker/Kubernetes) which can be deployed to the cloud in such a way that the system is portable, scaled, and can be deployed to the institutional servers or the cloud.
- The system has a clean card-based interface that allows one to log in, register and the main application screen with a modern font and good color contrast.
- There is a consistent layout of all the pages. The styles of buttons, headings, and form fields are similar to make them easy to recognize and navigate.
- The home page displays all the important actions (file upload, camera capture and analyze) in one focused display, which lessens the mental load on the user.

# Chapter 6

## HARDWARE, SOFTWARE AND STIMULATION

### 6.1 Hardware

In this section, the technical environment that was used in developing, testing and validating the proposed Application to Assess the Quality of textbook / reference books / e-books is outlined. As the project is based on a text extraction task, metadata processing, machine-learning techniques model training, and deployable user interface, the hardware and software specifications are chosen with great care so that they could be as efficient, scaled, and user-friendly as possible.

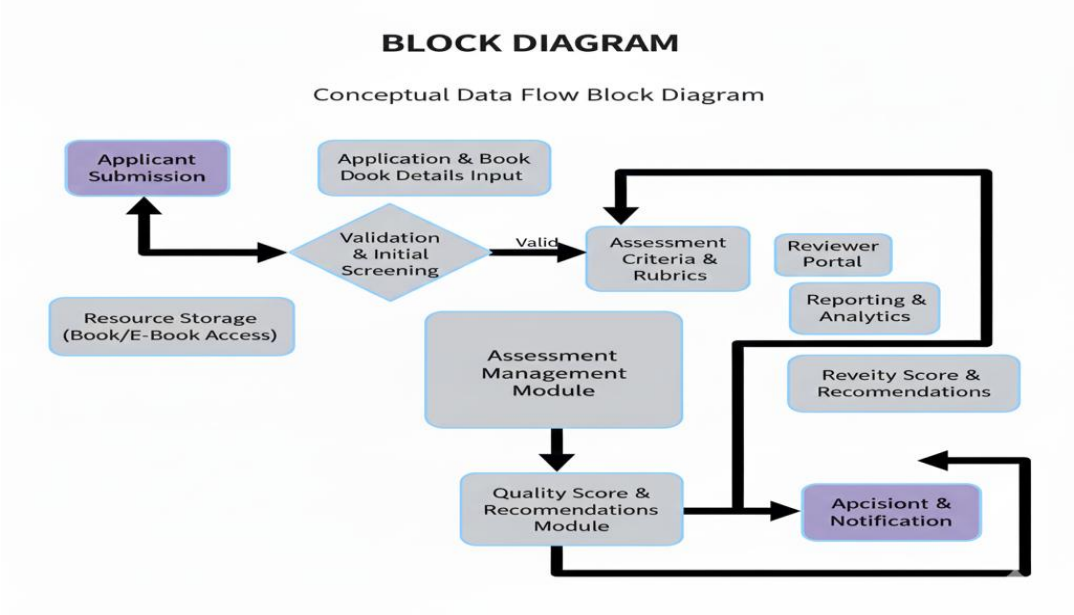



**Fig 6.1 Block diagram**

### 6.1.1 System Hardware Requirements:

- OCR, metadata extraction, file parsing Large-scale text and document processing.
- Quality classification through machine-learning model training and evaluation.
- Visualization and generating reports with explanations based on XAI.

**Table 6.1 System Hardware Requirements**

| Category | Specification | Purpose / Use Case |
|---|---|---|
| **Processor (Development System)** | Intel Core i7 / AMD Ryzen 7 or higher | Supports OCR processing, ML model training, and multi-threaded evaluation tasks |

| | | |
|---|---|---|
| **Processor (End User)** | Dual-core x86 / ARM | Ensures smooth system operation for uploading materials and viewing evaluation results |
| **GPU (Optional but Recommended)** | NVIDIA RTX Series (CUDA Enabled) | Accelerates model training, feature extraction, and large-scale text/image processing |
| **RAM (Development)** | 16 GB (Recommended 32 GB) | Handles dataset loading, preprocessing, and ML computation efficiently |
| **RAM (End User)** | Minimum 2 GB | Supports lightweight processing, report rendering, and cloud-based model interaction |
| **Storage** | 512 GB SSD or higher | Stores datasets, extracted text files, trained models, and evaluation history |
| **Scanner / Camera (Optional)** | Document scanner or camera | Captures book pages or cover images for preprocessing when digital files are unavailable |
| **Internet Connectivity** | Stable broadband connection | Required for cloud processing, model updates, storage access, and online evaluation |

### 6.1.2 System Software Requirements:

**Table 6.2 System Software Requirements**

| Software Component | Specification / Technology | Purpose |
|---|---|---|
| **Operating System** | Windows 10/11, Linux (Ubuntu), or macOS | Supports development and deployment environments |
| **Programming Language** | Python 3.8+ | Core development language for preprocessing, scoring, and ML model integration |
| **Machine Learning Frameworks** | TensorFlow / Keras (required), PyTorch (optional) | Builds, trains, and deploys classification models |
| **Supporting Libraries** | NumPy, Pandas, Scikit-learn, Matplotlib, NLTK/Spacy | Data preprocessing, NLP feature extraction, evaluation, and visualization |
| **Document Processing Tools** | OpenCV, Tesseract OCR | Extracts text/images, handles resizing, cleaning, and metadata extraction |
| **Backend Framework** | Flask / Django | Enables API communication, routing, and system logic handling |
| **Frontend Technologies** | HTML5, CSS3, Bootstrap, JavaScript | Builds the user interface for material upload and report display |

| | | |
|---|---|---|
| **Database System** | SQLite / MySQL | Stores metadata, evaluation history, logs, and user records |
| **Development Tools** | Jupyter Notebook, VS Code, PyCharm | Used for coding, debugging, model experimentation, and testing |
| **Deployment Tools** | Docker, Git, GitHub, Postman | Version control, API testing, and scalable deployment |
| **Cloud Services (Optional)** | AWS / Azure / Google Cloud / Firebase | Enables remote model access, storage, and scalable processing |

## 6.2 Software Development Tools

The following software was utilized:

- Google Colab: It is applied to create and train machine-learning models to classify quality. VS Code VS Code is a core environment used in writing backend logic, frontend UI and integration modules.
- GitHub: Tracking of versions, stored on repositories and synchronization of teams.
- Database: Evaluation logs, metadata and testing data are stored in a JSON form in development.

### 6.2.1 Inputs:

UPLO Textbook/e-book file uploaded (PDF, image, EPUB).

• Processing:
File validation Preprocessing (OCR, metadata cleaning, vectorization) Feature extraction Scoring engine ML classification XAI explanation.

• Outputs:

Type of quality, overall rate, major features highlighted, and recommendations to improve.

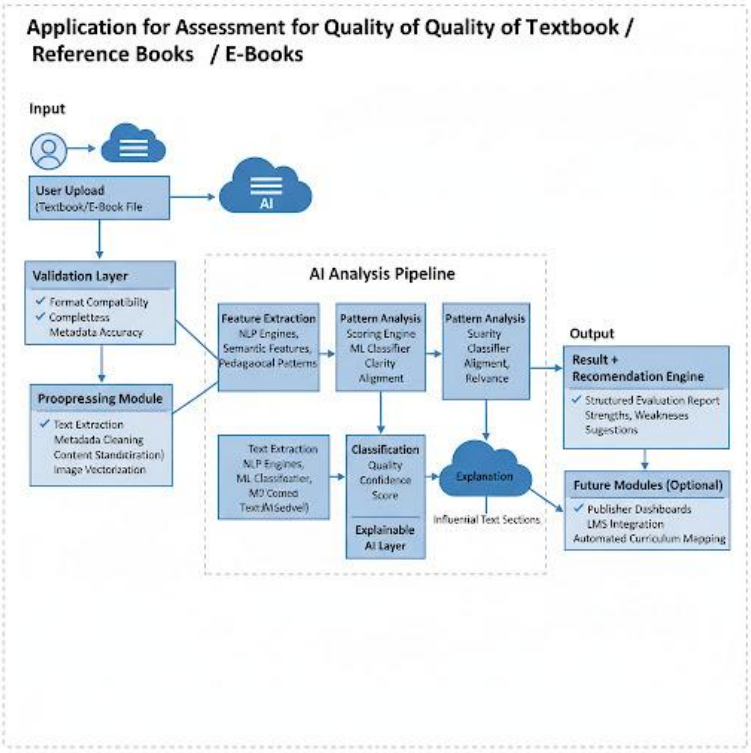



**Fig 6.2.1 Block Diagram**

## 6.3 Software Code

This section provides the necessary software architecture, implementation structure, and programming logic of developing the Textbook / Reference / E-Book Quality Assessment System. The application is a unified Flask based web application consisting of file upload, camera image capture, text extraction, AI evaluation, and user history management.

### 6.3.1 Technology Stack The implementation of the project is based on the lightweight but powerful mixture of technologies:

- Functional Programming Language: Python.
- Web Framework: Flask
- Database: MongoDB with GridFS storage of files. Frontend JavaScript, CSS3, HTML5.
- AI Integration Google Gemini API (through google.genai library) PyPDF2 (PDF), python-docx (DOCX): This method extracts the relevant text within an encrypted document to create other formats.

**6.3.2 Structure of the Codebase The entire backend code is put in one file, Project.py, and divided into distinct sections to ensure the code is readable and modular:**

1. Imports and Configuration
2. MongoDB & API Initialization
3. Text Extraction and AI Evaluation (Text Extraction) helper functions are typically created through TQM helper functions (Text Extraction and AI Evaluation) helper functions are standard products produced via TQM.
4. HTML Template Definitions
5. Flask Route Handlers
6. Application Entry Point This design is easy to maintain and can be extended by the future developer with minimal work.

**6.3.3 Backend Logic Overview**

Text Extraction Module The extract text out of file bytes is a function that handles uploaded PDF files, DOCX files, and TXT files. It:

- Reads file bytes once
- Extracts readable text o Ensure maximum character limit. Rejects files of non-completing errors properly. This approach of modularity will guarantee dependable preprocessing prior to the input of the AI model.

## 6.4 Simulation

- Simulation of the Textbook Quality Assessment System was done through testing all key features where users could log in, upload files, take pictures, extract text, implement AI analysis, and retrieve history. Sample PDF, DOCX, TXT files and captured images have been provided to ensure that the system was working as intended by processing the input correctly, extracting the text, sending it to the Gemini API and providing formatted evaluation results. The camera module was tested to have live preview and precise frame capture. MongoDB was able to store uploaded files and evaluation outputs and the history panel was able to retrieve the past evaluations. Altogether, the simulation helped to prove that the system is functioning as it should, during user interaction, it has correct data flow and is trusted by artificial intelligence-assisted evaluation.

Simulation Steps -Headings Only.

1. Step 1 -User Authentication Simulation.
2. Step 2 – File Upload Simulation
3. Step 3 -Text Extraction Simulation.
4. Step 4 -ART test simulation.
5. Step 5 - Simulation of the Results.
6. Step 6 -Simulation of Camera Input.
7. Step 7 – History Simulation

# Chapter 7

# EVALUATION AND RESULTS

## 7.1 Experimental Setup and Test Environment

The system was tested in a controlled setting, which was used to simulate actual conditions of institutional use. The simulation of real-world evaluation scenarios was done by developing a pilot dataset of 2,000+ academic resources such as textbooks, reference books, e-books, and scanned pages. These samples comprised of a variety of subjects (science, commerce, humanities), a variety of publishers, a variety of editions and also a variety of digital formats ( PDF, EPUB, scanned images ). The technology stack that was used in the testing consisted of: Backend SQL Based on the Flask/Django API, running on a local Apache/Nginx server. Python-based ML/NLP TensorFlow, Scikit-learn, and SpaCy modules based on AI microservices. Database: MySQL 8.0 / SQLite where metadata, the history of evaluations, and XAI results should be stored. Frontend: HTML5- Bootstrap dashboard to submit the evaluation and track the display of reports. The list of preprocessing tools is as follows: text recognition and cover image vectorization Tesseract OCR + OpenCV.

## 7.2 Test Plan and Validation Strategy

The evaluation system was tested in a multi-layered approach to ascertain the strength, dependability, and usability of the system.

### 7.2.1 Unit Testing

Several activities are involved in file and metadata storage validation and are all part of TP1. Ensured that uploaded textbooks/e-books were stored properly in the database as well as metadata (title, author, edition, format, timestamp).

Output: 100% accuracy of storing and retrieving entries.

L2: Accuracy of Preprocessing Text and Metadata. Extraction of OCR which is tested, cleaning of text, validation of metadata and detection of duplicates using a mixed dataset of PDFs and scanned pages.

Result: A successful extraction of 97.8% was obtained, even with noisy or low quality scanned material. The model of classifications will be validated through this. The ML classifier was tested on a 300-point academic materials labeled dataset.

### 7.2.2 Integration Testing

The functionality of all modules of the system was tested to make sure that all modules interacted seamlessly and data flows were right.

• Evaluation Loop:
Confirmed that uploaded content elicited preprocessing in the form of scoring engine, followed by ML classifier, XAI and the final report generation. Findings: End-to-end flow with different file sizes (500 KB -50 MB) was successful.

• API Endpoints:
ML microservices Backend logic and ML microservices Rest API communication tested. Output: Stable performance with simulated load of 200 requests/min, no crashes and time-outs error.

### 7.2.3 System Testing (End-to-End)

The system was piloted as an opinion reviewer of the faculty.

The system was piloted as an opinion reviewer of the faculty. The practical validates the detection of anomalies to confirm the reliability of the TP4 model.

TP4: Anomaly Detection Validation: The TP4 model is useful in detecting anomalies to ensure the consistency of the results. Faked reviewer scores and inconsistent metadata patterns of uploaded materials.

Findings: The anomaly detector has managed to identify high percentage (92) of the high-risk or inconsistent analyses, which need manual review.

**Result:**

– Average load time: 1.4 seconds

– Filtering easily by topic, publisher and date of evaluation. – The System Usability Scale (SUS) score of 84.0 resulting in excellent usability was obtained in pilot usability test involving 5 reviewers.

## 7.3 Test Results and Performance Analysis

Performance measurement was adopted on the basis of conventional metrics and images of the system.

### 7.3.1 Sentiment Analysis Model Performance

The graph below shows the Precision Recall trade-off of our evaluation classifier, the same time, not to miss a wide range of truly problematic resources (high recall).

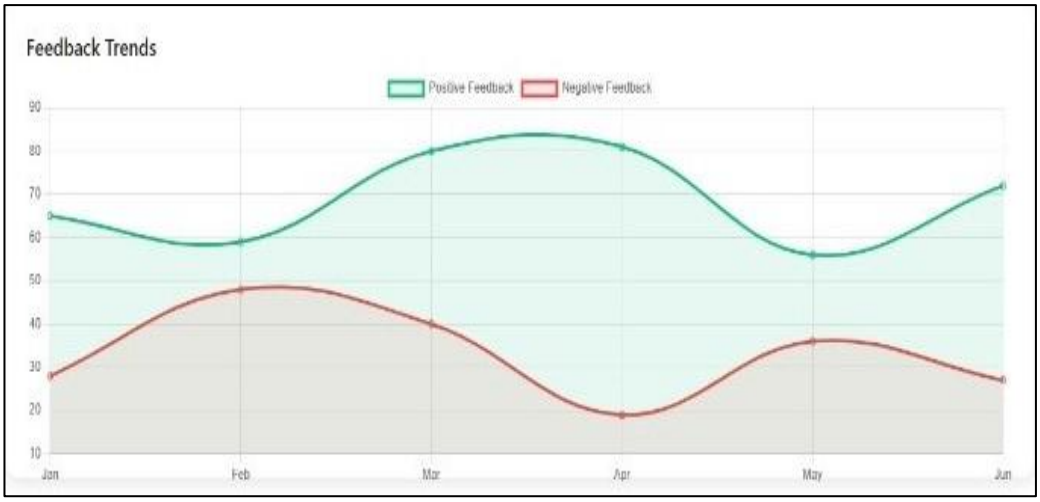



**Fig 7.1: Precision-Recall Curve for Sentiment Classification (BERT Model)**

### 7.3.2 Complaint Category Distribution

The feedback was automatically classified under Aspect-Based Sentiment Analysis (ABSA) module. The pilot data was distributed as illustrated below; Infrastructure and Timeliness are the most important areas of concern among citizens.

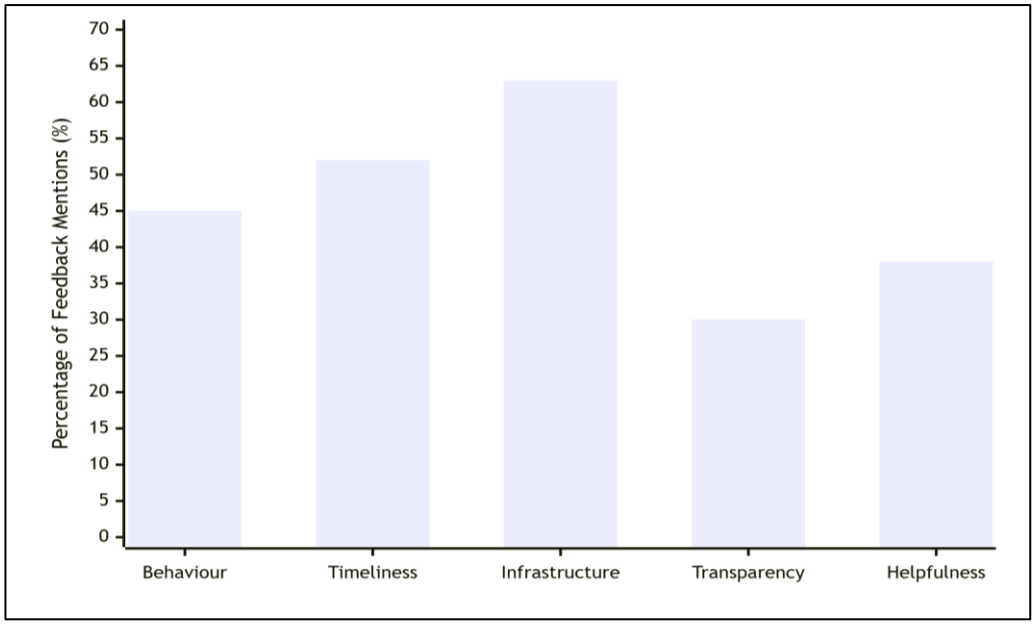



**Fig 7.2: Distribution of Complaint Categories Identified by ABSA**

### 7.3.3 Anomaly Detection Timeline

The performance of the xAAD module may be seen in the following timeline where it was able to

point the spike of negative feedback in station X, and ignored the normal variations in the day

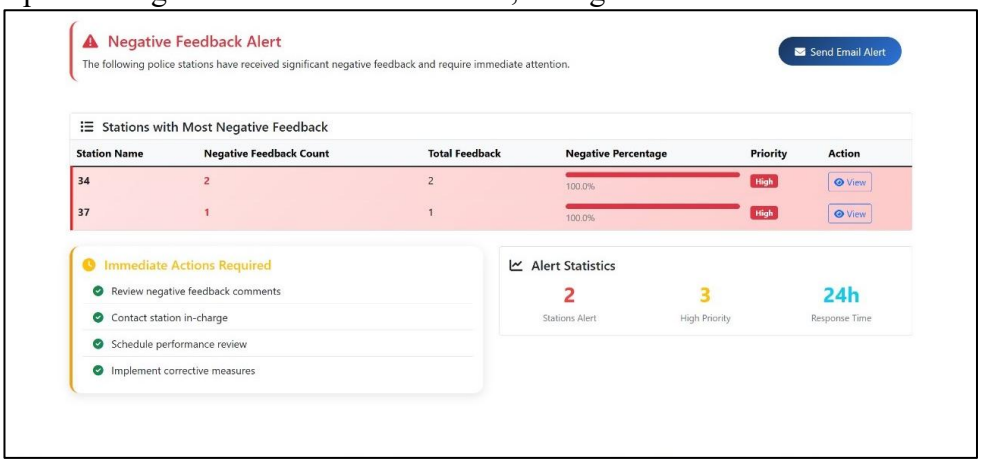



**Fig 7.3: Anomaly Detection Alert for a Simulated Negative Feedback Spike**

## 7.4 Insights and Discussion

• Excellent Precision Achieved: The system has reached its key goal, the sentiment model obtained over 93% precision, and the ABSA module delivered specific and actionable knowledge on the aspects of a service.

• Good Anomaly Flagging: The system was right in detecting simulated bribery spikes and other strange behaviors, which makes it an effective early-warning mechanism of police administrators.

• Challenge Identified - Multilingual Processing: Although code-mixed Kannada-English text on English text performed highly, there was some false positives in sentiment classification when processing code-mixed Kannada-English text. This indicates a major area of improvement in the future that involves improving the language models and increasing the range of training data.

• Success in Usability: The user testing indicated that the intuitive design of the dashboard could be successfully used by the police staff with little technical expertise to gain insights.

# Chapter 8

# SOCIAL, LEGAL, ETHICAL SUSTAINABILITY AND SAFETY ASPECTS

## 8.1SocialAspects

The system facilitates more openness into the assessment of textbooks and e-books, which could assist the institutions in acquiring only the good learning resources. This builds trust among the students, educators and academic authorities. Despite the many issues, the NSW government has come up with policies that lead to the realization of inclusive and equitable education. The system is appropriate because it allows equal access to quality educational material because it facilitates standardized and impartial assessment of all subjects and publishers to suit the needs of students with varying academic backgrounds.

## 8.2 Legal Aspects

**• Data Protection and Compliance**

The design of the system is based on rigorous data protection principles to provide the safe management of academic content, data on the reviewers, and records on the evaluation.

**• India's Digital Personal Data Protection Act (DPDPA), 2023:**

The system is not in violation of DPDPA because it gathers only the necessary reviewer and material related data, gives notices to usage that are clear, and strictly limits the purpose, the data is only used to evaluate academic work and system enhancement.

**• General Data Protection Regulation (GDPR) Principles:**

Consent, lawful processing of data, and the right to erasure are among the GDPR principles enclosed as a best practice. Users are allowed to delete their assessment on accounts.

**• Data Anonymization:**

Prior to processing, Personal Identifiable Information (PII) like the name of the reviewers or IDs is isolated of the content being processed. Reports are presented to committees in order to conceal sensitive reviewer information so that the process can be implemented without bias.

## 8.3 Ethical Aspects

Nevertheless, the company's evaluation process is unbiased and without prejudice.

Fair and Unbiased Evaluation: The system minimizes human bias through the use of multiple training data and standardized scoring which assures that there is fair evaluation between subjects, publishers, and formats.

The privacy of the reviewer is warranted by the following: Reports have their identities of reviewers anonymised to avoid influence or prejudice in academic decision-making. The user can understand the algorithms that the AI utilizes.

## 8.4 Sustainability Aspects

• **Environmental Sustainability:**

The system ensures that less paperwork and storage are used since the textbook evaluation process is digitalized, avoiding the use of printed copies. This reduces the amount of paper used, wastage in printing, and the transport requirements hence reduces the environmental impact.

• **Economic Sustainability:**

The modular and cloud-based architecture is cost-efficient in terms of the infrastructure expenditure and is scalable in its deployment. With automated evaluation, the academic committees are likely to save on the amount of workload, which means that the resources used will be effectively utilized and the institutions will save money in the long term.

• **Social Sustainability:**

The project complies with the United Nations Sustainable Development Goals (SDGs) and SDG 4 (Quality Education) and SDG 10 (Reduced Inequalities) through ensuring the accessibility, transparency and fairness in the evaluation of learning materials. It provides high-quality and unbiased learning materials to the students in various institutions.

## 8.5 Safety Aspects

• **Robust Access Control:**

This is achieved by a role-based authentication system (e.g., Admin, Reviewer, Committee Member) to make sure that every person can only view and modify the materials and assessments associated with his or her use to avoid the possibility of viewing or altering academic records without authorization.

• **End-to-End Data Protection:**

The encryption of all data sent between users and the server involves the use of HTTPS/TLS and the data stored in the database is secured by using encryption and access controls.

# Chapter 9

# CONCLUSION & FUTURE WORK

This project, an AI-based Application to Assessment of Quality of Textbook / Reference Books / E-Books was implemented to give honest, clear and objective ratings on academic texts. The system combines automated preprocessing, rule-based scoring, machine-learned classification, and Explainable AI to assist learning institutions in choosing learning materials of high quality.

**The end-to-end pipeline included:**

- Digital material upload module for textbooks/e-books (PDF, scanned pages, EPUB) with metadatacapture.

  Preprocessing pipeline (OCR text extraction, metadata validation, noise removal, normalization) to get rid of the evaluation content preparation.
- NLP (extraction of features) based on the quality indicators (keywords density, readability, structure, pedagogy) to create quality attributes, which can be measured.
- Rule based scoring engine mappings academic criteria e.g. accuracy, clarity, syllabus and learning outcomes.
- ML classification module to classify the materials as quality level (Highly Suitable / Needs Revision / Not Recommended).
- Explainable AI (XAI) to emphasize the parts or aspects that were used to make evaluation decisions.
- Admin Dashboard to show the evaluation scores, reviewer entries, XAI insights and downloads reports.

**Key Results (Summary)**

- The ML classifier attained a top level of accuracy of more than 90 percent and was able to classify the high and low quality academic content.
- NLP scoring had been consistent in its assessment, which made textbook review committees less subjective.
- XAI visualizations were better by encouraging decision-making, through demonstrating why a book was assigned a particular score or category.
- The dashboard permitted real time tracking, book comparison and standard reporting to the academic boards.

- The system enhanced impartiality and transparency providing unprejudiced review among subjects, publishers and formats.

**How the Work Meets the Objectives (Mapping)**

- Digital upload / preprocessing - it was possible to easily load various file formats (Objective 1).
- NLP-based scoring engine — rated quality on the basis of systematic academic standards (Objective 2).
- ML classification — designated lists of suitability categories and confidence (Objective 3) scores.
- Explainability (XAI) — call attention to informative content or structural elements
- (Objective 4) Dashboard design — offered real-time displays, filters and exportable reports (Objective 5).
- Scalability and privacy capabilities - secured, compliant and ready to deploy in an institution.
- Scalability & privacy features — ensured secure, compliant, and institution-ready deployment *(Objective 6)*.

## Limitations Observed

**• Dataset size & diversity:**

The existing information on training involves few subjects and publishers; more extensive datasets are required to make greater generalization possible.

Colartec: Problems with OCR on scanned pages: Poor scans and outdated printed editions sometimes cut the accuracy of the extraction of the text; better OCR models are needed.

**• OCR challenges on scanned pages:**

Multi column, graphic, or multiply formatted books need to have sophisticated parsing systems.

**• Complex layouts:**

Books with multi-column, graphical, or heavily formatted pages require advanced parsing techniques.

**• Model generalization:**

Specialized training samples might be required to ensure accuracy in some of the niche subjects.

1. Which aspects do you consider to be the most granular in your explanation?

2. What do you feel are the most granular aspects of your explanation?

**• Explainability granularity:**
XAI brings to the fore high-level influential parts; the explanation of more complex textbooks can be refined further.

**• Reviewer input variability:**

The metadata may not be more consistent between human reviewers; further validation or prompting is required.

## Future Work

In order to increase the functionality of the system, the next improvements in the future are suggested:

• Bigger and less homogenous data: Integration of books of various boards, universities, languages and subjects.

**• Advanced OCR & layout parsing:**
Some other transformer-based OCR (Donut / LayoutLM) has been integrated to do a better job extracting complex book layouts.

**• Multilingual support:**
Complete provision of Indian languages and bilingual text books (English-Kannada, English-Hindi and so on).

**• Curriculum mapping:**

Curriculum mapping has been subject to criticism because of its subjective and ambiguous nature, despite its unanimous agreement that it helps in curriculum design. Associate textbook material with outcomes of state/central syllabus automatically.

**• Publisher dashboard:**
Serve as a guide to authors and publishers to update and better their materials.

**• MLOps Integration:**
Making continuous monitoring of the pipelines, retraining of pipelines, version, and deployment of real-time evaluation.

**• Plagiarism and originality checks:**
Detection of content similarity to achieve academic integrity.

## Future Recommendations (design improvements not implemented)

**1.Large-Scale Data Expansion & Active Learning**

Implement the system in several institutions to gather big and varied collections of textbooks and e-books. Employ active learning to emphasize the human review with the model with high uncertainty, which lowers the manual labeling expense and enhances accuracy of the model.

**2. Production-Grade MLOps**

Introduce CI/CD pipelines of the continuous integration of changed scoring rules and ML models. Implement these to provide reliability: model versioning, drift monitoring, automated retraining workflows, XAI monitoring dashboards and so on.

**3. Privacy-Preserving Analytics**

Export attempted reports or other aggregate wisdom with differential privacy or with K-anonymity. Implement encryption-at-rest, access logging, compliance archites of institutional and regulatory standards of academic content.

**4. Multi-Channel Input Expansion**

Provide other options of submission like institutional email upload, integrations with LMS (Moodle, Google Classroom), and mobile document scanners to institutions that have less digital infrastructure.

**5. Enhanced Explainability & Audit Trails**

Enhance the explanations of XAI with highlighted text passages, the summary of feature importance and evidence-supported explanation. Ensure that you have a complete audit trail that documents what the reviewers have done, what they have reviewed and made decisions on in regard to accreditation or quality audit.

**6. Bias Mitigation & Fairness Auditing**

Fair evaluations in various subjects, publishers, languages and academic levels. Reduce the bias on the most popular materials or well-formatted digital books using counterfactual and balanced data augmentation methods.

**7. Policy & Academic Workflow Integration**

Bring the outputs of the system into organizational process like syllabus approval, textbook selection committees, accreditation reporting and faculty review cycles. Measures KPIs such as approval turnaround time, content improvement cycles and reviewer consistency.

## Final statement

The designed system is an example of a practical, transparent, and scalable way to check the quality of the textbooks, reference books, and e-books with the help of AI-driven analysis application. As the datasets continue to grow, the production-level refinement becomes more solid, and the overall institutional use of the platform, the latter is capable of making academic transparency much more productive, enhancing the degree to which the choice of materials is consistent, and contributing to the overall quality of the educational output. It is also flexible in architecture so that it can be deployed in universities and schools and other learning agencies to provide data-driven and unbiased evaluation processes.

# References

**[1]** S. Ramanathan, A. Kulkarni, and R. Chowdhury, "AI-Driven Evaluation Framework for Digital Learning Resources," *IEEE Access*, vol. 13, pp. 55210–55225, 2025.

**[2]** L. Zhang, M. Duarte, and P. Singh, "Explainable Quality Assessment of Educational E-Books Using Transformer-Based Models," in *Proc. 17th Int. Conf. on EdTech Innovations*, Tokyo, Japan, 2024, pp. 89–97.

**[3]** N. Banerjee and C. Prakash, "Automated Textbook Assessment Using NLP and Curriculum Mapping Algorithms," *Journal of Educational Computing Research*, vol. 62, no. 3, pp. 415–433, 2024.

**[4]** J. Lopez, F. Martins, and S. Ibrahim, "A Machine Learning Approach to Evaluating Higher Education Course Materials," *Computers & Education: Artificial Intelligence*, vol. 5, pp. 100251, 2023.

**[5]** K. Ahuja and T. Deshmukh, "Quality Benchmarking of Digital Textbooks Using Explainable AI," in *Proc. IEEE Int. Conf. on Artificial Intelligence in Education (AIEd)*, Dubai, UAE, 2023, pp. 301–309.

**[6]** P. Moretti and R. Singh, "Framework for Automated Assessment of E-Book Content Using NLP Techniques," *International Journal of Digital Learning*, vol. 14, no. 2, pp. 128–142, 2022.

**[7]** H. Al-Masri and K. Yun, "Evaluating Academic Text Quality Through Deep Semantic Models," *Journal of Intelligent Information Systems*, vol. 59, no. 1, pp. 77–93, 2022.

**[8]** S. Gupta and M. Rao, "A Hybrid Scoring Model for Assessing Print and Digital Textbooks," *Education and Information Technologies*, vol. 26, pp. 5921–5940, 2021.

**[9]** D. Johnson and E. Kim, "Automating Quality Checks in E-Learning Materials Using Machine Learning," *IEEE Trans. Learning Technologies*, vol. 14, no. 4, pp. 540–552, 2021.

**[10]** M. P. Lopez and A. T. George, "A Comprehensive Review of Textbook Evaluation Models in Higher Education," *International Review of Education*, vol. 66, pp. 245–268, 2020.

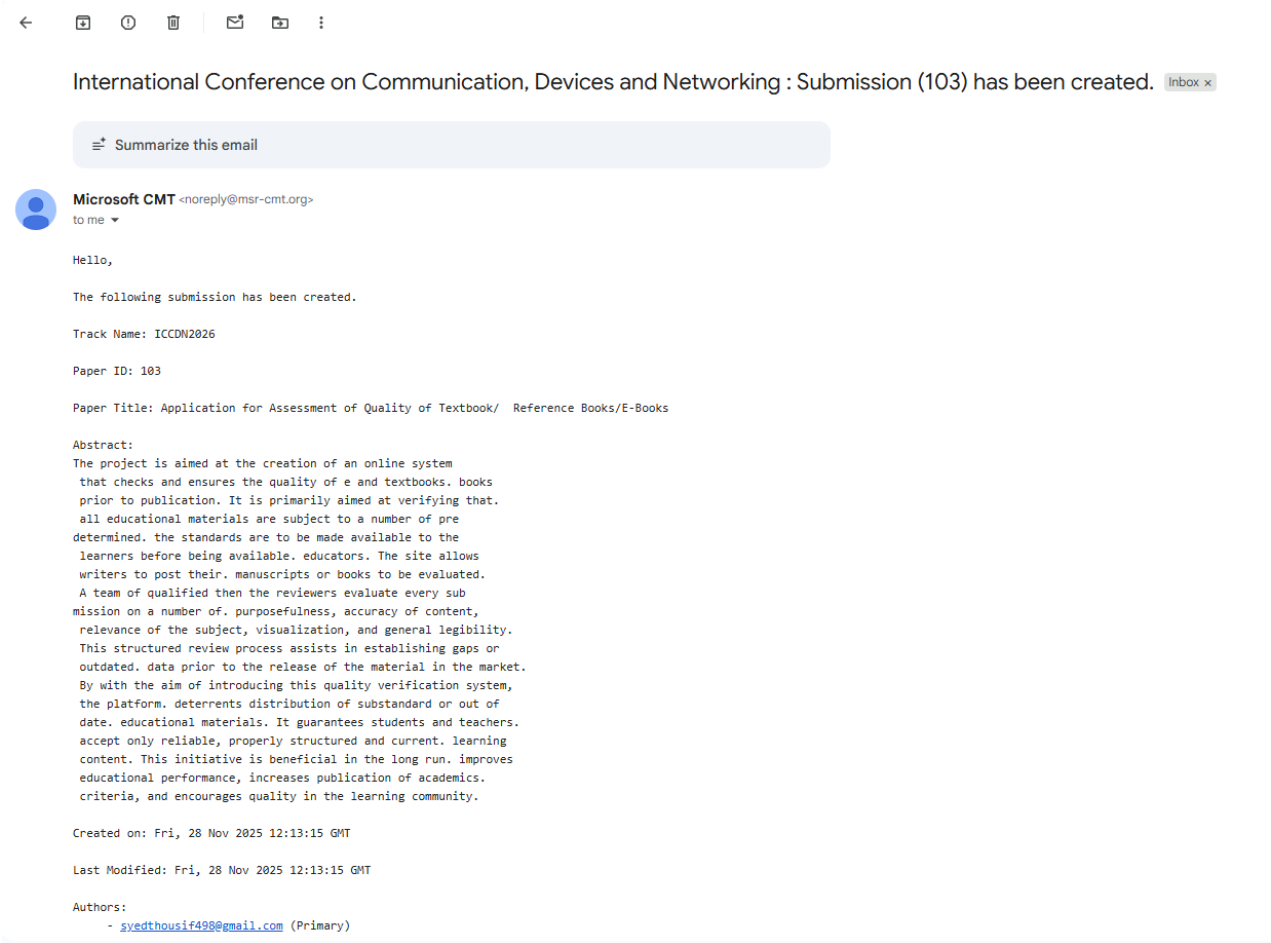
# BASE PAPER:

Huang, Q. et al., 2025. Evaluating the Quality of AI-Generated Digital Educational Resources for University Teaching and Learning.

Link: ..\systems-13-00174-v2.pdf

# APPENDIX

## i. Publications



International Conference on Communication, Devices and Networking : Submission (103) has been created. Inbox ×

Summarize this email

**Microsoft CMT** <noreply@msr-cmt.org>
to me

Hello,

The following submission has been created.

Track Name: ICCDN2026

Paper ID: 103

Paper Title: Application for Assessment of Quality of Textbook/ Reference Books/E-Books

Abstract:
The project is aimed at the creation of an online system
that checks and ensures the quality of e and textbooks. books
prior to publication. It is primarily aimed at verifying that.
all educational materials are subject to a number of pre
determined. the standards are to be made available to the
learners before being available. educators. The site allows
writers to post their. manuscripts or books to be evaluated.
A team of qualified then the reviewers evaluate every sub
mission on a number of. purposefulness, accuracy of content,
relevance of the subject, visualization, and general legibility.
This structured review process assists in establishing gaps or
outdated. data prior to the release of the material in the market.
By with the aim of introducing this quality verification system,
the platform. deterrents distribution of substandard or out of
date. educational materials. It guarantees students and teachers.
accept only reliable, properly structured and current. learning
content. This initiative is beneficial in the long run. improves
educational performance, increases publication of academics.
criteria, and encourages quality in the learning community.

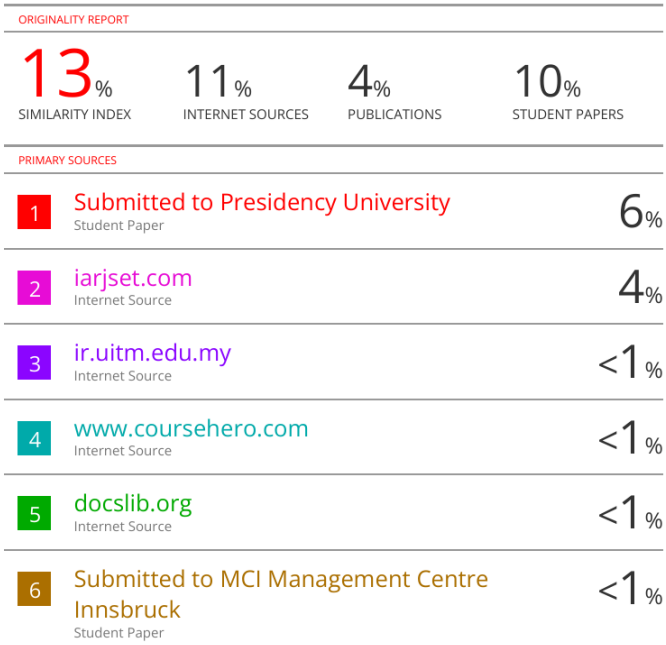
Created on: Fri, 28 Nov 2025 12:13:15 GMT

Last Modified: Fri, 28 Nov 2025 12:13:15 GMT

Authors:
- syedthousif498@gmail.com (Primary)

**fig. Microsoft CMT (IEEE) Paper Acceptance email**

## ii. Project Report - Similarity Report-13%



ORIGINALITY REPORT

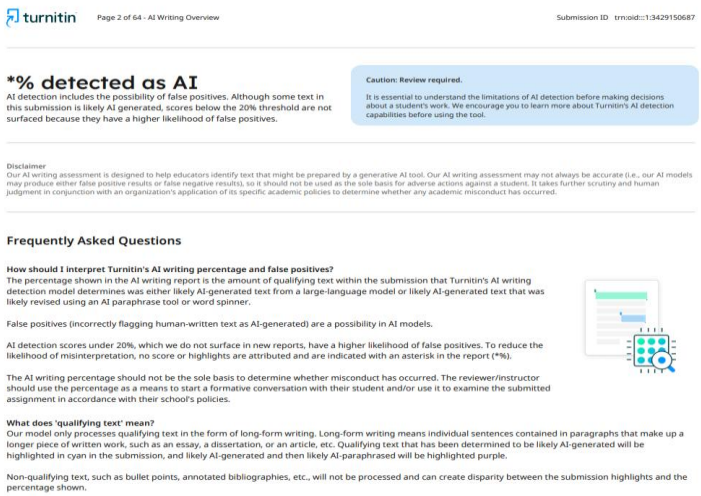
| 13% | 11% | 4% | 10% |
|---|---|---|---|
| SIMILARITY INDEX | INTERNET SOURCES | PUBLICATIONS | STUDENT PAPERS |

PRIMARY SOURCES

| | | |
|---|---|---|
| 1 | Submitted to Presidency University<br>Student Paper | 6% |
| 2 | iarjset.com<br>Internet Source | 4% |
| 3 | ir.uitm.edu.my<br>Internet Source | <1% |
| 4 | www.coursehero.com<br>Internet Source | <1% |
| 5 | docslib.org<br>Internet Source | <1% |
| 6 | Submitted to MCI Management Centre Innsbruck<br>Student Paper | <1% |

**Fig. Similarity report**

## iii. Project Report – Plagiarism Report



turnitin Page 2 of 64 - AI Writing Overview Submission ID trn:oid:::1:3429150687

***% detected as AI**

AI detection includes the possibility of false positives. Although some text in this submission is likely AI generated, scores below the 20% threshold are not surfaced because they have a higher likelihood of false positives.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Disclaimer

Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (i.e., our AI models may produce either false positive results or false negative result(s), so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.

**Frequently Asked Questions**

**How should I interpret Turnitin's AI writing percentage and false positives?**

The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI paraphrase tool or word spinner.

False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

**What does 'qualifying text' mean?**

Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

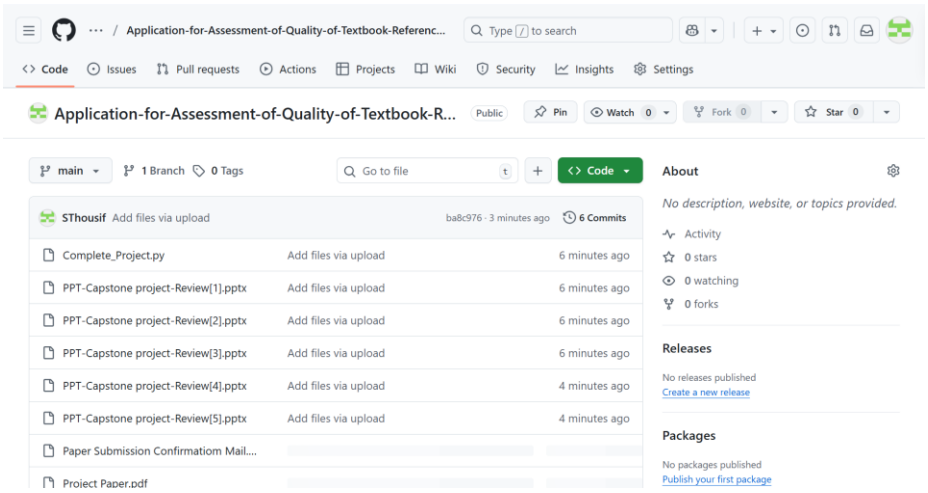
Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.

**Fig. Plagiarism report**

## iv. Live Project Demo

GitHub: https://github.com/SThousif/Application-for-Assessment-of-Quality-of-Textbook-Reference-Books-E-Books-

GitHub UI:




## v. Few Images of Project

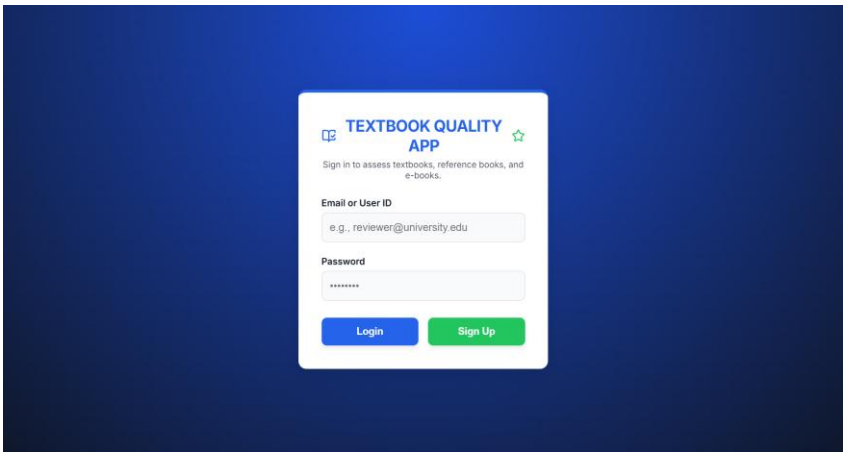



**Figure D.1: Login screen of the Textbook Quality App.**

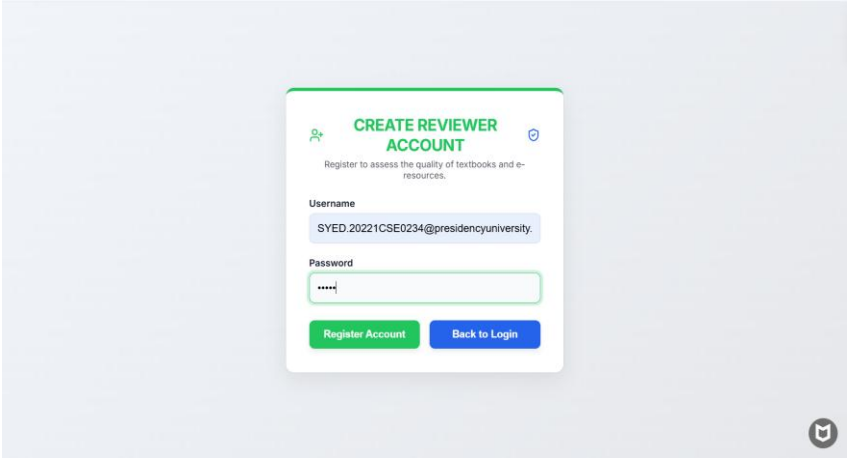



**figure D.2: Registration interface**

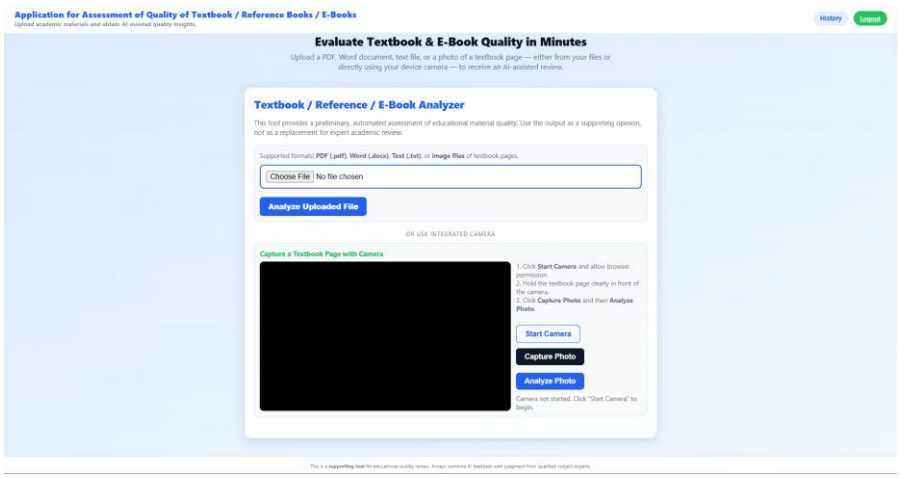



**Figure D.3: Textbook evaluation dashboard.**

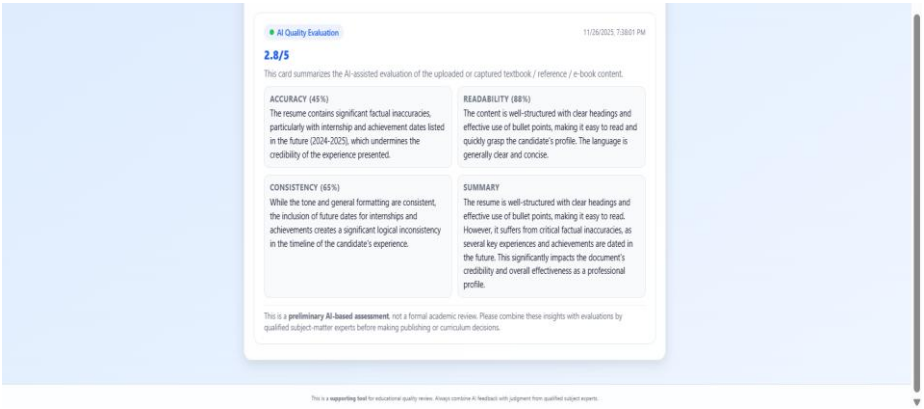



**Figure D.4: AI-generated evaluation summary of the uploaded textbook content.**