AI-POWERED LEARNING AND TEACHING AUTOMATION SYSTEM

25-26J-162

Project Proposal Report

M Wanniarachchi W.A.P - IT22103154

BSc. (Hons) Degree in Information Technology, Specializing in Information Technology

Department of Information Technology

Sri Lanka Institute of Information Technology
Sri Lanka

August 2025

AI-POWERED LEARNING AND TEACHING AUTOMATION SYSTEM

25-26J-162

Project Proposal Report

Supervised By Prof. Samantha Rajapaksha
Co-Supervised by Mrs. Bhagyani Chathurika

BSc. (Hons) Degree in Information Technology, Specializing in Information Technology

Department of Information Technology

Sri Lanka Institute of Information Technology
Sri Lanka

August 2025

DECLARATION

I declare that this is our own work, and this proposal does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Name	Student ID	Signature
M Wanniarachchi W.A.P	IT22103154	

Supervisor's statement: The candidates mentioned above are conducting research for their undergraduate dissertations under my supervision.

Supervisor
(Prof. Samantha Rajapaksha)
Date: 26/08/2025

Co-supervisor
(Mrs. Bhagyani Chathurika)
Date: 26/08/2025

ABSTRACT

This research proposes the development of an AI-powered system that automates student grading and provides intelligent teacher reteaching guidance using custom-built Optical Character Recognition (OCR), Natural Language Processing (NLP), and Machine Learning (ML) models. Current educational assessment methods are often time-consuming, subjective, and inconsistent. Moreover, teachers lack structured guidance on which concepts to reteach and in what sequence after identifying student weaknesses. The proposed system tackles these problems by integrating a web and mobile application for document capture, automated evaluation, and AI-driven reteaching recommendations.

The solution starts with an OCR module that extracts text from scanned or photographed answer scripts and feedback forms. It supports both handwritten and printed content. The AI grading engine processes the extracted responses, applying NLP and ML techniques to assess answers based on marking schemes, semantic similarity, structure, and language quality. The system generates automated scores, question-wise explanations, and personalized improvement recommendations for students.

For teacher reteaching guidance, the platform utilizes a Knowledge Graph-based approach that maps student errors to syllabus concepts and prerequisite relationships. Using Graph Neural Networks (GNNs) and Bayesian Knowledge Tracing (BKT), the system propagates weaknesses across concept dependencies and predicts future learning risks. The AI-powered reteaching module generates targeted recommendations for teachers, identifying which prerequisite concepts need reinforcement and providing visual heatmaps overlaid on the Knowledge Graph to guide instructional decisions at both individual student and classroom levels.

The methodology includes developing a custom OCR pipeline, creating NLP-based grading models, constructing syllabus-based Knowledge Graphs, and implementing GNN-BKT hybrid models for weakness propagation analysis. The expected results include significant reductions in grading time, better feedback accuracy, and data-driven reteaching strategies that ensure students master foundational concepts before advancing. The system's offline capabilities and mobile support ensure it works well in low-connectivity environments. This research aims to deliver a scalable, objective, and efficient academic evaluation platform that benefits both students and educators through intelligent automation and structured pedagogical guidance.

Keywords: AI-powered grading, OCR, NLP, teacher reteaching automation, Knowledge Graph, GNN, BKT, educational analytics.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Abbreviation	Full Form
AI	Artificial Intelligence
API	Application Programming Interface
BERT	Bidirectional Encoder Representations from Transformers
BKT	Bayesian Knowledge Tracing
GPU	Graphics Processing Unit
GNN	Graph Neural Network
KG	Knowledge Graph
ML	Machine Learning
MSER	Maximally Stable Extremal Regions
NER	Named Entity Recognition
NLP	Natural Language Processing
OCR	Optical Character Recognition
SSL	Secure Sockets Layer
SUS	System Usability Scale
UI/UX	User Interface/User Experience
VPS	Virtual Private Server

Table 1 : List Of Abbreviation

1. INTRODUCTION

This research proposes the development of an AI-powered educational support system that integrates student grading and concept-level reteaching guidance into a single framework. Traditional assessment methods in education are slow, subjective, and often fail to identify deeper learning gaps. Teachers primarily rely on exam scores to judge performance, which does not capture the prerequisite knowledge students may be missing [1], [2]. As a result, reteaching strategies are often generic, repetitive, and disconnected from actual student weaknesses.

The proposed system addresses these limitations by combining multiple AI-driven components into one end-to-end solution. First, Optical Character Recognition (OCR) is applied to digitize handwritten and printed answer scripts, enabling automated grading using Natural Language Processing (NLP) and Machine Learning (ML) techniques [2], [3]. This ensures accurate and scalable evaluation of student responses with question-level analysis. Next, teacher performance evaluation is supported through feedback analysis and performance metrics, ensuring that instructional quality is continuously monitored.

The key novelty of this research lies in the introduction of an AI-powered Reteaching Guidance System. Unlike existing platforms that stop at grading or teacher evaluation, this component builds a Knowledge Graph (KG) of syllabus concepts and prerequisite relationships. Student errors identified during grading are mapped to corresponding nodes within the graph. Using Graph Neural Networks (GNNs) and Bayesian Knowledge Tracing (BKT), weaknesses are propagated across connected concepts, predicting future risks of failure. The system then generates actionable reteaching recommendations for teachers at both student and classroom levels, supported by interactive visualizations such as heatmaps overlaid on the Knowledge Graph.

This integrated approach delivers a two-fold benefit: automated grading that reduces time and bias in evaluation and targeted reteaching guidance that ensures students master prerequisite concepts before advancing to higher levels. By combining these elements, the system aims to provide a scalable, objective, and intelligent academic support platform that improves learning outcomes while supporting teachers with data-driven decision-making [4], [5].

1.1. Background & Literature Survey

Assessment of students and evaluation of their conceptual understanding are central to improving learning outcomes. However, conventional grading approaches end with assigning scores and do not provide teachers with actionable insights into prerequisite-based weaknesses [1], [2]. As a result, reteaching strategies are often generic and repetitive, failing to address the root causes of poor performance. This limitation becomes even more critical in large classrooms or remote learning settings, where teachers lack the time and resources to individually analyze student errors [1], [3].

The integration of Artificial Intelligence (AI) and Machine Learning (ML) into educational systems has opened new opportunities for fine-grained student modeling [4], [5]. One enabling technology is Optical Character Recognition (OCR), which digitizes handwritten and printed exam responses for automated processing [2]. When combined with Natural Language Processing (NLP), student answers can be evaluated not only for correctness but also for higher-order skills such as comprehension and application, often categorized under Bloom's Taxonomy [3]. While effective, most existing systems stop at providing overall marks or general feedback, leaving teachers without guidance on what to reteach or in which order [1], [3], [5].

Bayesian Knowledge Tracing (BKT) has been widely applied in intelligent tutoring systems to model a learner's mastery of skills over time based on sequences of correct and incorrect responses [4]. Similarly, Graph Neural Networks (GNNs) have recently shown promise in educational contexts by capturing dependencies among concepts and predicting performance across interconnected knowledge domains [6]. These methods highlight the potential for modeling knowledge as structured graphs, where weaknesses in foundational concepts can be propagated to reveal risks in advanced topics.

Knowledge Graphs (KGs) are particularly suited for representing educational domains, as they encode hierarchical structures of subjects, topics, and subtopics along with prerequisite relationships [6], [7]. When combined with student performance data, KGs can serve as powerful tools for visualizing strengths and weaknesses at a concept level. Yet, most prior works have focused either on predictive modeling (e.g., BKT) or observational metrics (e.g., classroom analysis tools like ACORN) without integrating graph-based visualizations that provide teachers with actionable reteaching strategies [7].

The research presented in this proposal seeks to address these gaps by introducing a self-contained AI-powered Reteaching Guidance System that integrates OCR-based grading, concept mapping, Knowledge Graph construction, and AI-driven analysis. The system will deliver concept-level heatmaps of student understanding and generate targeted reteaching recommendations, enabling teachers to focus on prerequisite concepts and ensure efficient, personalized instruction.

1.2. Research Gap

Analysis of existing educational assessment platforms reveals significant gaps in comprehensive teacher reteaching guidance automating system. The following comparative analysis demonstrates the novelty of the proposed teacher reteaching guide engine:

Aspect	**Previous Work 1 – ACORN (2021)**	**Previous Work 2 – Bayesian Knowledge Tracing (BKT, 2019)**	**Proposed Component – Knowledge Graph of Student Understanding**
Focus	Teacher observation via audio/video multimodal analysis to rate classroom climate.	Predict student mastery over time using statistical sequences of correct/incorrect answers.	Builds concept-level Knowledge Graph from OCR-extracted exam data to analyze understanding and reteaching needs.
Data Used	Classroom recordings (video + audio).	Student response sequences (online test logs).	Paper-based exam answers → OCR → mapped to syllabus concepts.
Output	Teacher performance scores (e.g., positive/negative climate).	Probability that student has mastered each skill.	Visual graph heatmap of concepts (weak/strong) + reteaching recommendations for teachers.
Methodology	Deep multimodal models (CNNs, temporal convs).	Hidden Markov Models / Bayesian updating.	Graph Neural Networks (GNNs) + Bayesian Knowledge Tracing integrated with OCR grading.

Limitations	Focused only on teacher side, ignores student concept mastery.	No representation of concept dependencies, only skill mastery probability.	Introduces structured graph of syllabus + concept dependency propagation → helps reteaching decisions.
Research Gap	No link between student answers and knowledge structures.	No visualization / reteaching guidance, only prediction values.	Fills both gaps: connects exam errors → concept nodes, shows weak areas, and gives reteaching guidance.
Novelty Claim	Teacher-centered analysis.	Predictive, not prescriptive.	First integration of OCR + Concept Mapping + GNN for student-centered reteaching guidance.

Table 2 : Research Gaps with Existing Studies

1.3. Research Problem

In educational environments, accurate diagnosis of student learning gaps and effective reteaching are critical processes that directly influence knowledge retention, progression, and overall academic performance. However, existing systems face several key limitations, particularly in identifying prerequisite-based weaknesses and providing structured reteaching guidance.

For student assessment, most current platforms are designed primarily to produce grades or overall performance scores [1], [2]. These systems rarely provide teachers with fine-grained insights into which specific concepts or prerequisite topics a student has failed to master. Without such mappings, reteaching strategies become repetitive and unfocused, leading to wasted instructional time and reduced learning efficiency [3].

For knowledge modeling, many existing solutions rely on sequential performance tracking (e.g., Bayesian Knowledge Tracing) or skill-level mastery predictions without incorporating explicit relationships between concepts [4]. While these models can forecast performance, they do not explain *why* students struggle or *which concepts* should be retaught first. Similarly, classroom observation systems such as ACORN emphasize teacher performance but fail to link student mistakes to structured knowledge representations [7].

The situation is further complicated by the absence of graph-based visualization and AI-driven analysis in existing educational systems. Most platforms lack Knowledge Graph (KG) representations that highlight dependencies among concepts, and very few incorporate advanced AI methods such as Graph Neural Networks (GNNs) for propagating weaknesses across prerequisite chains [6]. Teachers are therefore left without actionable, concept-level dashboards that can translate student performance data into effective reteaching strategies.

Given these challenges, there is a clear need for an AI-powered reteaching guidance system that can:

1. Construct a syllabus-based Knowledge Graph encoding topics, subtopics, and prerequisite relationships [6], [7].
2. Map student exam errors from OCR and NLP pipelines directly to concept nodes for precise weakness identification [2], [3].
3. Implement AI-driven analysis using Graph Neural Networks (GNNs) to propagate weaknesses and Bayesian Knowledge Tracing (BKT) to predict risks of future failure [4], [6].
4. Provide teachers with visual heatmaps and actionable recommendations for reteaching, ensuring that prerequisite concepts are mastered before advancing [5], [7].

Addressing this problem will result in a system that not only evaluates student performance but also generates structured, data-driven reteaching strategies. This ensures efficient use of classroom time, improved student mastery of foundational concepts, and enhanced overall learning outcomes.

2. OBJECTIVES

The primary objective of this research is to design and develop an AI-powered Reteaching Guidance System that integrates OCR-based student response analysis, Knowledge Graph (KG) construction, and AI-driven inference models to provide concept-level reteaching recommendations. The system will overcome the limitations of conventional grading platforms by identifying prerequisite-based weaknesses, propagating them across knowledge structures, and delivering actionable insights for teachers through visual dashboards.

2.1. Main Objectives

Develop a Knowledge Graph–driven AI-powered Reteaching Guidance System that accurately identifies student weaknesses at the concept level and provides targeted reteaching strategies using Graph Neural Networks (GNNs) and Bayesian Knowledge Tracing (BKT).

- **Specific**: Construct a syllabus-based ontology, integrate OCR and NLP modules for exam response extraction, build a Knowledge Graph with concept and prerequisite mappings, and implement AI models (GNNs + BKT) for weakness propagation and mastery prediction.

- **Measurable**: System performance will be evaluated by achieving at least 80% accuracy in mapping exam responses to correct syllabus concepts, producing reteaching recommendations with a minimum of 80% agreement compared to expert teacher judgments, and demonstrating prediction accuracy above 85% for future student performance on unseen concepts [2], [4], [6].

- **Achievable**: The objective is feasible with existing OCR/NLP frameworks (Tesseract, spaCy), graph modeling technologies (Neo4j, NetworkX), and AI frameworks (PyTorch Geometric, BKT algorithms), supported by well-documented methodologies in educational AI research [3], [4], [6], [7].

- **Realistic**: The development plan aligns with available expertise in AI/ML, educational data modeling, and software development, while the project scope ensures that implementation is achievable within the proposed timeframe and institutional support.

- **Time-bound**: Completion is planned within 5 months, with 2 months for OCR + NLP integration, 1 month for Knowledge Graph construction, 1.5 months for GNN + BKT implementation, and 0.5 months for system testing and optimization.

2.2. Specific Objective

Implement OCR and NLP-based Student Response Extraction

- Specific: Digitize handwritten and printed exam scripts using OCR, and classify extracted answers into correctness levels and Bloom's taxonomy categories.
- Measurable: Achieve at least 90% OCR accuracy on academic scripts and 80% accuracy in Bloom's taxonomy classification [2], [3].
- Achievable: Utilizes established OCR tools (Tesseract, OpenCV) and NLP models (BERT).
- Realistic: Feasible with available datasets of student exam scripts and training resources.
- Time-bound: Expected completion within 1.5 months.

Construct a Syllabus-based Knowledge Graph (KG)

- Specific: Build a graph structure representing syllabus topics, subtopics, and prerequisite dependencies, embedding student performance data into concept nodes.
- Measurable: Achieve at least 85% accuracy in mapping exam questions to correct graph nodes as validated by subject experts.
- Achievable: Leverages Neo4j graph database and ontology design methods.
- Realistic: Aligns with available syllabus documents and concept taxonomies.
- Time-bound: Expected completion within 1 month.

Implement Graph Neural Networks (GNNs) for Weakness Propagation

- Specific: Develop GNN models to propagate detected weaknesses across prerequisite nodes in the Knowledge Graph.
- Measurable: Achieve 80% correlation between GNN-predicted weaknesses and expert teacher evaluations.
- Achievable: Uses PyTorch Geometric and DGL libraries for model implementation.
- Realistic: Supported by recent research on GNNs in educational data mining [6].
- Time-bound: Expected completion within 1 month.

Integrate Bayesian Knowledge Tracing (BKT) for Mastery Prediction

- Specific: Apply BKT to predict future student performance and mastery probabilities across concept nodes.
- Measurable: Achieve predictive accuracy above 85% on test datasets compared to actual student outcomes.
- Achievable: Uses established probabilistic modeling frameworks with available educational datasets [4].
- Realistic: Aligns with the project's analytical scope and available AI expertise.
- Time-bound: Expected completion within 0.5 months.

Develop Visualization and Teacher Dashboard

- Specific: Create dashboards with heatmaps overlayed on the Knowledge Graph to highlight student strengths/weaknesses and generate reteaching recommendations.
- Measurable: Achieve at least 85% usability score (SUS) in teacher usability testing.
- Achievable: Uses React.js/React Native for the interface and D3.js/NetworkX for visualization.
- Realistic: Supported by available frontend development expertise.
- Time-bound: Expected completion within 1 month.

3. METHODOLOGY

The methodology of this research uses a structured, multi-phase approach to design and develop the AI-Powered Reteaching Guidance System. The project combines OCR-based student response extraction, syllabus-based Knowledge Graph construction, and AI-driven analysis into a single platform. The process includes data collection, knowledge modeling, system implementation, and evaluation to ensure a reliable and accurate solution.

3.1. Overall System Architecture

The AI-Powered Student Grading and Teacher Teaching Guidance Automation System will follow a five-layer architecture [6][7][8].

1. Input Layer: Upload answer sheets and feedback forms (mobile/web).

2. Preprocessing Layer: Image preprocessing, including noise reduction and skew correction, using OpenCV [2].
3. Processing Layer: OCR, NLP-based grading, and ML-driven analytics [1], [3].
4. Analysis Layer:
    - Automated student grading with explanations.
    - Teacher Reteaching Guidance and gives Student Questioners
5. Output Layer: Personalized improvement reports, dashboards, and performance analytics.

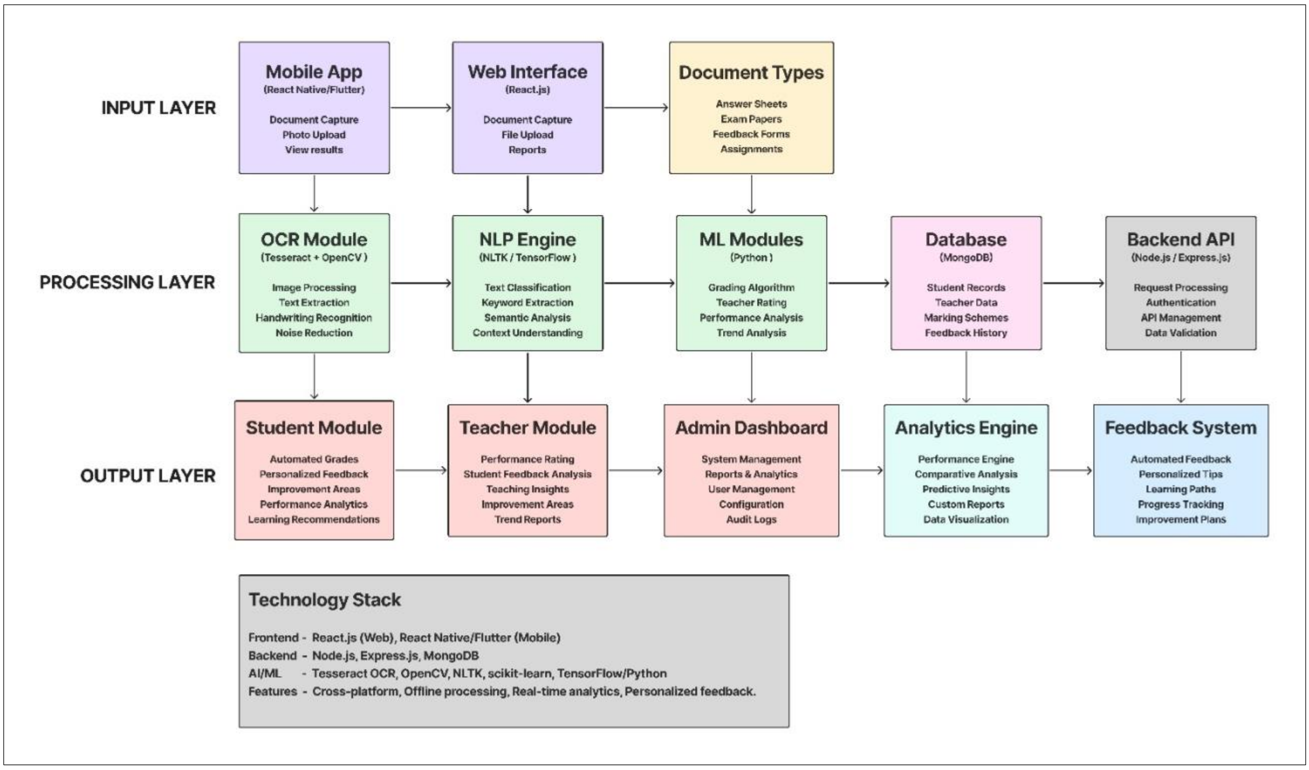

Figure 1: Overall System Architecture Diagram

3.2. Individual System Architecture

The Individual System Architecture illustrates the detailed flow of data and interactions within the AI-powered Reteaching Guidance System. It is structured into five key layers [2], [4], [6]:

1. Input & Capture Layer: Student exam scripts are captured via mobile or web applications and processed using OCR and NLP to extract responses and Bloom's levels.

2. Concept Mapping Layer: Each question is mapped to its respective syllabus concept using a predefined ontology. Errors are tagged with the corresponding nodes.

3. Knowledge Graph Layer: A Neo4j graph database stores concepts as nodes and prerequisite relationships as edges, embedding student performance (strength/weakness scores) directly into the graph.

4. AI Analysis Layer: GNN models propagate weaknesses across prerequisite chains, while BKT predicts the probability of future mastery or failure.

5. Visualization & Feedback Layer: A dashboard overlays performance heatmaps onto the Knowledge Graph and generates reteaching recommendations for teachers at both class and student levels.

This modular architecture ensures scalability, interpretability, and practical integration into real classroom workflows.

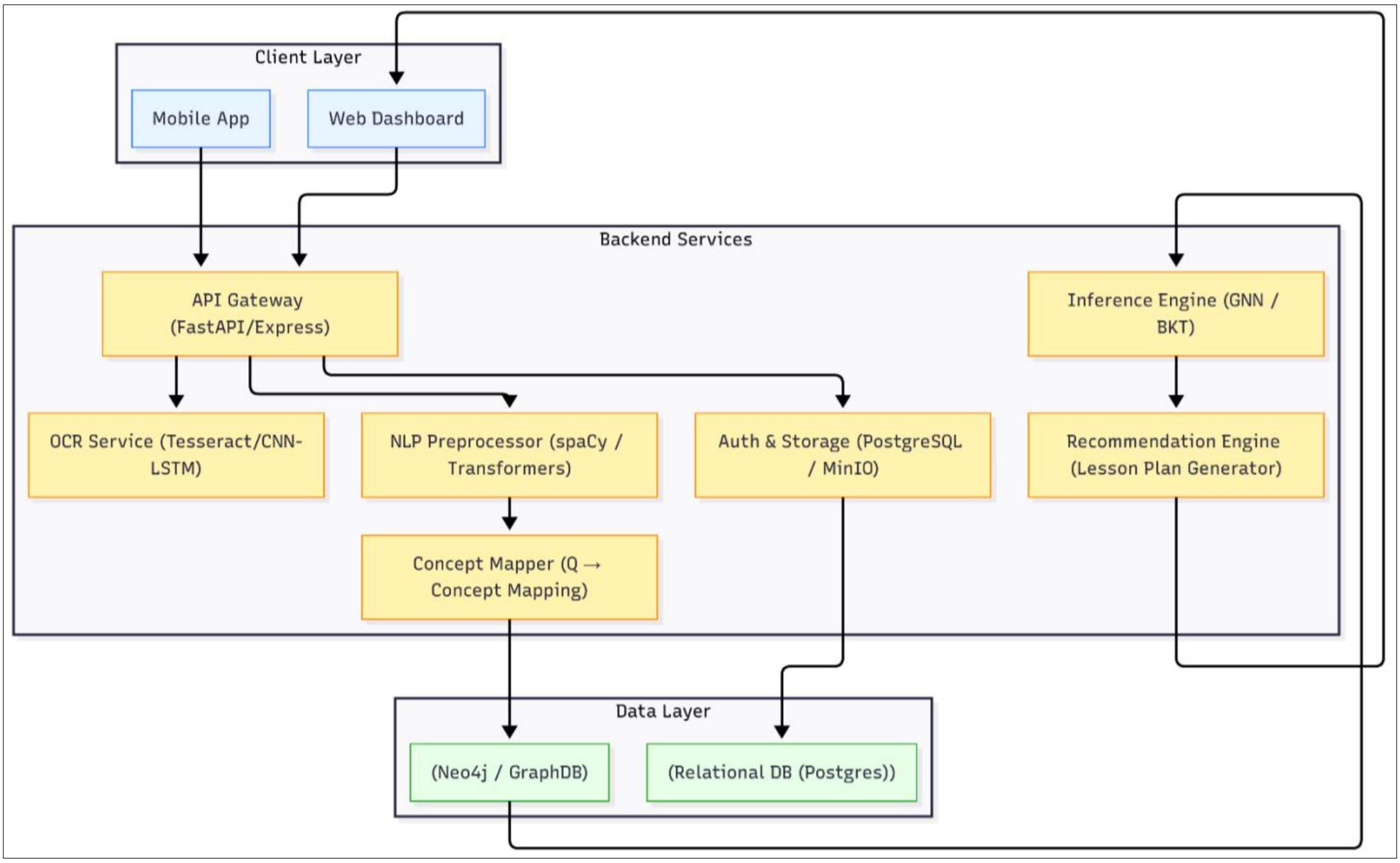

Figure 2 : Individual System Architecture Diagram

3.3. System Implementation

The implementation will follow a modular development approach. The OCR and NLP modules will be built using Tesseract, spaCy, and Bloom's taxonomy classification models. Concept mapping will be implemented with a custom syllabus ontology, stored in a Neo4j graph database. The AI engine will integrate PyTorch Geometric for GNNs and Bayesian models for Knowledge Tracing [4], [6].

The backend will be developed with Node.js and Express.js, serving RESTful APIs for communication between components. The frontend dashboard will be built with React.js for web and React Native for mobile, ensuring cross-platform accessibility. The visualization module will use NetworkX and D3.js for rendering heatmaps and graph overlays.

Offline-first functionality will be supported via local caching and synchronization with the central graph database once connectivity is available. This ensures the system can operate in low-resource educational environments.

4. GRANT CHART

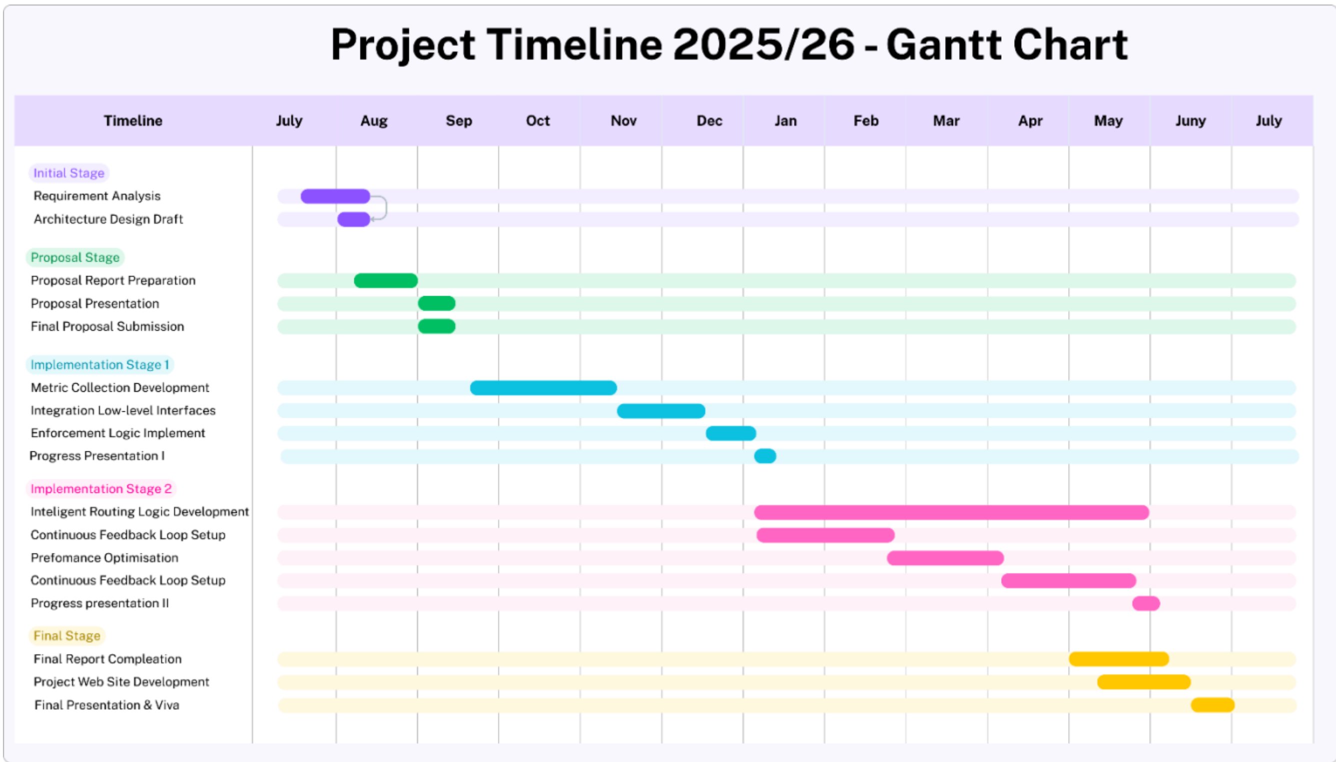

Figure 3 : Grant Chart

5. DESCRIPTION OF PERSONAL AND FACILITIES

5.1. Personnel

This project, AI-Powered Learning and Teaching Automation System, is carried out by a team of four undergraduate researchers specializing in Information Technology at SLIIT. Each member is responsible for a key part of the system:

- **Pathiraja P.U.M (IT22243362):** Development of the OCR module and student grading engine, focusing on text extraction from handwritten and printed academic scripts.
- **Wanniarachchi W.A.P.M (IT22103154):** Implementation of the Teacher Reteaching automation engine, integrating with AI-driven evaluation models like GKT,GNN with Knowledge Graps(KG).
- **Jayasooriya L.T (IT22095480):** Design of algorithms for student performance analysis, identifying weak areas and generating personalized improvement recommendations.
- **Hettiarachchi R.H (IT22120052):** Development of teacher improvement analytics using Bloom's Taxonomy, sentiment analysis, and performance monitoring.

The research team is supervised by Prof. Samantha Rajapaksha, who offers guidance on research methods, technical validation, and academic standards.

5.2. Facilities

The successful implementation of the AI-Powered Learning and Teaching Automation System needs a mix of software tools, hardware resources, and school infrastructure. The project will rely on modern software, using frameworks like Node.js, Express.js, React.js, and React Native/Flutter for backend and frontend development. For the artificial intelligence parts, TensorFlow, PyTorch, and Scikit-learn will design and train machine learning models. Tesseract and OpenCV will support the custom OCR module. A MongoDB database will act as the primary data storage solution. Collaboration platforms like GitHub, Slack, and Google Drive will be used for version control, communication, and document management.

For hardware resources, the project team will use high-performance computers with GPU acceleration to meet the computational needs of training and testing deep learning models. Mobile devices on both Android and iOS will be used for testing the cross-platform mobile application. For preparing datasets, scanners and digital cameras will capture and digitize handwritten answer sheets, creating a varied training and validation dataset.

The project will also gain strong support from the institution with access to laboratories, computing infrastructure, and technical resources available at SLIIT. These facilities will create a secure and collaborative environment for developing, testing, and integrating models. The institution will also offer expert help, especially for labeling datasets, validating grades, and evaluating teacher models. Additionally, secure storage will ensure that all student and teacher data stay anonymous and protected, following ethical and institutional data-handling standards.

6. BUDGET AND BUDGET JUSTIFICATION

Description	Cost	Occurring
Server Hosting (VPS/Cloud)	LKR 8,000	Monthly
App Store Hosting (Apple)	LKR 7,754	One-Time
Play Store Hosting (Google)	LKR 30,700	Annual
Cloud Database (Amazon S3)	USD 0.023 per GB	Monthly
GPU Cloud Service (Model Training)	LKR 15,000	Monthly (as required)
Domain Registration & SSL	LKR 5,000	Annual
Miscellaneous (Testing devices, Scanning tools, Internet, etc.)	LKR 5,000	One-Time

Table 3 : Budget For The project

6.1. Budget Justification

The proposed budget outlines the necessary costs for implementing and deploying the AI-Powered Learning and Teaching Automation System. A monthly server hosting fee of LKR 8,000 is needed to maintain the backend and ensure system reliability. The Apple App Store hosting fee of LKR 7,754 is required as a one-time expense, while the Google Play Store fee of LKR 30,700 is an annual cost. Both are crucial for distributing mobile applications to users. For secure and scalable data storage, Amazon S3 will be used, with costs estimated at USD 0.023 per GB per month, depending on the size of the dataset.

Since the project involves training machine learning models, we may need additional computing power from GPU-based cloud services, estimated at LKR 15,000 per month, based on usage. Domain registration and an SSL certificate cost LKR 5,000 annually and provide secure access to the system. Finally, we have included miscellaneous costs of LKR 12,000 as a one-time fee to cover expenses for testing devices, document scanning tools, and internet access during the development and validation phases.

This budget ensures that the system can be developed, deployed, and maintained effectively while meeting the project's technical and functional requirements.

7. REFERENCE

[1] K. Rani and R. Sujatha, “Automated Evaluation of Descriptive Answers using NLP Techniques,” *Int. J. of Advanced Computer Science and Applications*, vol. 11, no. 5, pp. 435–440, 2020.

[2] M. Jain and V. Yadav, “Optical Character Recognition Using Tesseract and OpenCV,” *Journal of Emerging Technologies and Innovative Research*, vol. 6, no. 2, pp. 112–116, 2019.

[3] R. S. Baker and P. S. Inventado, “Educational Data Mining and Learning Analytics,” in *Learning Analytics*, Springer, pp. 61–75, 2014.

[4] J. A. C. Conati and K. VanLehn, “Bayesian Student Modeling and Knowledge Tracing,” *User Modeling and User-Adapted Interaction*, vol. 12, no. 4, pp. 371–417, 2002.

[5] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.

[6] Z. Cui, X. Wang, and Y. Cai, “Graph Neural Networks for Student Performance Prediction,” *IEEE Transactions on Learning Technologies*, vol. 15, no. 3, pp. 313–326, 2022.

[7] S. S. Kidwai et al., “ACORN: AI-based Classroom Observation for Rating and Noticing,” in *Proc. Int. Conf. on Artificial Intelligence in Education (AIED)*, 2021.